

AUTOMATIC
DYNAMIC
INCREMENTAL
NONLINEAR
ANALYSIS

Theory and Modeling Guide
Volume I: ADINA Solids & Structures

Report ARD 05-7

October 2005

ADINA R & D, Inc.

ADINA

Theory and Modeling Guide

Volume I: ADINA Solids & Structures

October 2005

ADINA R & D, Inc.
71 Elton Avenue
Watertown, MA 02472 USA

tel. (617) 926-5199
telefax (617) 926-0238
www.adina.com

Notices

ADINA R & D, Inc. owns both this software program system and its documentation. Both the program system and the documentation are copyrighted with all rights reserved by ADINA R & D, Inc.

The information contained in this document is subject to change without notice.

Trademarks

ADINA is a registered trademark of K.J. Bathe / ADINA R & D, Inc.

All other product names are trademarks or registered trademarks of their respective owners.

Copyright Notice

© ADINA R & D, Inc. 1987 - 2005

October 2005 Printing

Printed in the USA

Table of contents

1. Introduction	1
1.1 Objective of this manual.....	1
1.2 Supported computers and operating systems	2
1.3 Units	2
1.4 ADINA System documentation.....	4
2. Elements	7
2.1 Truss and cable elements.....	7
2.1.1 General considerations	7
2.1.2 Material models and formulations.....	9
2.1.3 Numerical integration.....	9
2.1.4 Mass matrices	10
2.1.5 Element output	11
2.1.6 Recommendations on use of elements.....	11
2.1.7 Rebar elements	12
2.2 Two-dimensional solid elements.....	15
2.2.1 General considerations	15
2.2.2 Material models and formulations.....	21
2.2.3 Numerical integration.....	22
2.2.4 Mass matrices	24
2.2.5 Element output	25
2.2.6 Recommendations on use of elements.....	30
2.3 Three-dimensional solid elements.....	34
2.3.1 General considerations	34
2.3.2 Material models and nonlinear formulations.....	42
2.3.3 Numerical integration.....	44
2.3.4 Mass matrices	47
2.3.5 Element output	47
2.3.6 Recommendations on use of elements.....	53
2.4 Beam elements	54
2.4.1 Linear beam element	58
2.4.2 Large displacement elastic beam element	64
2.4.3 Nonlinear elasto-plastic beam element.....	65
2.4.4 Moment-curvature beam element.....	70
2.4.5 Beam-tied element.....	80
2.4.6 Bolt element	82
2.5 Iso-beam elements; axisymmetric shell elements.....	84
2.5.1 General considerations	84

2.5.2 Numerical integration.....	89
2.5.3 Linear iso-beam elements.....	91
2.5.4 Nonlinear iso-beam elements	94
2.5.5 Axisymmetric shell element.....	95
2.5.6 Element mass matrices	96
2.5.7 Element output	96
2.6 Plate/shell elements	99
2.6.1 Formulation of the element	100
2.6.2 Elasto-plastic analysis	104
2.6.3 Element mass matrices	105
2.6.4 Element output	106
2.7 Shell elements.....	107
2.7.1 Basic assumptions in element formulation.....	107
2.7.2 Material models and formulations.....	115
2.7.3 Shell nodal point degrees of freedom.....	118
2.7.4 Transition elements	122
2.7.5 Numerical integration.....	124
2.7.6 Composite shell elements	127
2.7.7 Mass matrices	132
2.7.8 Anisotropic failure criteria	133
2.7.9 Element output	140
2.7.10 Selection of elements for analysis of thin and thick shells.....	147
2.8 Pipe elements.....	151
2.8.1 Material models and formulations.....	158
2.8.2 Pipe internal pressures.....	159
2.8.3 Numerical integration.....	162
2.8.4 Bolt element	165
2.8.5 Element mass matrices	166
2.8.6 Element output	167
2.8.7 Recommendations on use of elements.....	168
2.9 General and spring/damper/mass elements	169
2.9.1 General elements	169
2.9.2 Linear and nonlinear spring/damper/mass elements	171
2.9.3 Spring-tied element	176
2.9.4 User-supplied element	177
2.10 Displacement-based fluid elements	182
2.11 Potential-based fluid elements.....	184
2.11.1 Theory: Subsonic velocity formulation.....	187
2.11.2 Theory: Infinitesimal velocity formulation	194
2.11.3 Theory: Infinite fluid regions	195
2.11.4 Theory: Ground motion loadings	200

2.11.5 Theory: Static analysis	201
2.11.6 Theory: Frequency analysis	204
2.11.7 Theory: Mode superposition	205
2.11.8 Theory: Response spectrum, harmonic and random vibration analysis	207
2.11.9 Modeling: Formulation choice and potential master degree of freedom	209
2.11.10 Modeling: Potential degree of freedom fixities	210
2.11.11 Modeling: Elements	210
2.11.12 Modeling: Potential-interfaces	211
2.11.13 Modeling: Interface elements	213
2.11.14 Modeling: Loads	220
2.11.15 Modeling: Phi model completion	222
2.11.16 Modeling: Considerations for static analysis	237
2.11.17 Modeling: Considerations for dynamic analysis	238
2.11.18 Modeling: Considerations for frequency analysis and modal participation factor analysis.....	238
2.11.19 Modeling: Element output.....	239
3. Material models and formulations.....	241
3.1 Stress and strain measures in ADINA	241
3.1.1 Summary	241
3.1.2 Large strain thermo-plasticity and creep analysis	246
3.2 Linear elastic material models.....	251
3.2.1 Elastic-isotropic material model.....	253
3.2.2 Elastic-orthotropic material model.....	253
3.3 Nonlinear elastic material model (for truss elements).....	262
3.4 Isothermal plasticity material models.....	265
3.4.1 Plastic-bilinear and plastic-multilinear material models	266
3.4.2 Mroz-bilinear material model.....	272
3.4.3 Plastic-orthotropic material model	278
3.4.4 Ilyushin material model	282
3.4.5 Gurson material model	284
3.5 Thermo-elastic material models	286
3.6 Thermo-elasto-plasticity and creep material models	290
3.6.1 Evaluation of thermal strains.....	295
3.6.2 Evaluation of plastic strains	295
3.6.3 Evaluation of creep strains	299
3.6.4 Computational procedures.....	306
3.6.5 Shifting rule for the irradiation creep model	308
3.7 Concrete material model.....	309
3.7.1 General considerations	309

3.7.2 Formulation of the concrete material model in ADINA.....	310
3.8 Rubber and foam material models.....	328
3.8.1 Isotropic hyperelastic effects.....	329
3.8.2 Viscoelastic effects.....	348
3.8.3 Mullins effect	358
3.8.4 Orthotropic effect	362
3.8.5 Thermal strain effect	366
3.8.6 TRS material.....	368
3.8.7 Temperature-dependent material.....	368
3.9 Geotechnical material models	370
3.9.1 Curve description material model	370
3.9.2 Drucker-Prager material model	376
3.9.3 Cam-clay material model	380
3.9.4 Mohr-Coulomb material model.....	385
3.10 Fabric material model with wrinkling	389
3.11 Viscoelastic material model	390
3.12 Porous media formulation	394
3.13 Gasket material model.....	399
3.14 Shape Memory Alloy (SMA) material model	403
3.15 User-supplied material model.....	409
3.15.1 General considerations	409
3.15.2 2-D solid elements.....	410
3.15.3 3-D solid elements.....	421
3.15.4 User-supplied material models supplied by ADINA R&D, Inc.	422
4. Contact conditions.....	437
4.1 Overview	438
4.2 Contact algorithms for implicit analysis.....	446
4.2.1 Constraint-function method.....	447
4.2.2 Lagrange multiplier (segment) method	448
4.2.3 Rigid target method	449
4.2.4 Selection of contact algorithm.....	450
4.3 Contact algorithms for explicit analysis	450
4.3.1 Kinematic constraint method.....	451
4.3.2 Penalty method	452
4.3.3 Rigid target method	453
4.3.4 Selection of contact algorithm for explicit analysis	453
4.4 Contact group properties	454
4.5 Friction	462
4.5.1 Basic friction models.....	462
4.5.2 User-supplied friction models (not available in explicit analysis)	464

4.5.3 Frictional heat generation	467
4.6 Contact definition features	468
4.6.1 Analytical rigid targets	468
4.6.2 Contact body	468
4.6.3 Draw beads	469
4.7 Contact analysis features	470
4.7.1 Implicit dynamic contact/impact	470
4.7.2 Explicit dynamic contact /impact	471
4.7.3 Contact detection	472
4.7.4 Suppression of contact oscillations	473
4.7.5 Restart with contact	474
4.8 Modeling considerations	474
4.8.1 Contactor and target selection	474
4.8.2 General modeling hints	476
4.8.3 Modeling hints for implicit analysis	478
4.8.4 Modeling hints for explicit analysis	480
4.8.5 Convergence considerations (implicit analysis only)	482
5. Boundary conditions/applied loading/constraint	485
5.1 Introduction	485
5.2 Concentrated loads	490
5.3 Pressure and distributed loading	493
5.3.1 Two- and three-dimensional pressure loading	496
5.3.2 Hermitian (2-node) beam distributed loading	499
5.3.3 Iso-beam and pipe distributed loading	501
5.3.4 Plate/shell pressure loading	503
5.3.5 Shell loading	504
5.4 Centrifugal and mass proportional loading	504
5.5 Prescribed displacements	510
5.6 Prescribed temperatures and temperature gradients	512
5.7 Pipe internal pressure data	514
5.8 Electromagnetic loading	515
5.9 Poreflow loads	516
5.10 Phiflux loads	516
5.11 Contact slip loads	517
5.12 Surface tension boundary	518
5.13 Initial load calculations	519
5.14 User-supplied loads	519
5.14.1 General considerations	519
5.14.2 Usage of the user-supplied loads	519
5.14.3 Example: Hydrodynamic forces	522

5.15 Constraint equations	525
5.15.1 General considerations	525
5.15.2 Rigid links	526
5.16 Mesh Glueing	528
6. Eigenvalue problems	531
6.1 Linearized buckling analysis	531
6.1.1 General considerations	531
6.1.2 Introduction of geometric imperfections	534
6.2 Frequency analysis	534
6.2.1 Use of determinant search method	536
6.2.2 Use of subspace iteration method	536
6.2.3 Use of Lanczos iteration method	538
6.2.4 Modal stresses	539
7. Static and implicit dynamic analysis	541
7.1 Linear static analysis	541
7.1.1 Direct solver; sparse solver	542
7.1.2 Iterative solvers; multigrid solver.....	544
7.2 Nonlinear static analysis.....	546
7.2.1 Solution of incremental nonlinear static equations	547
7.2.2 Line search	550
7.2.3 Convergence criteria for equilibrium iterations	557
7.2.4 Selection of incremental solution method	561
7.2.5 Example.....	565
7.3 Linear dynamic analysis.....	570
7.3.1 Step-by-step implicit time integration	571
7.3.2 Time history by mode superposition	572
7.3.3 Lumped and consistent damping; Rayleigh damping	573
7.4 Nonlinear dynamic analysis	574
7.4.1 Step-by-step implicit time integration	575
7.4.2 Mode superposition	577
7.5 Choosing between implicit and explicit formulations	578
8. Explicit dynamic analysis	581
8.1 Formulation	583
8.1.1 Mass matrix	584
8.1.2 Damping	585
8.2 Stability	585
8.3 Time step management.....	588

9. Frequency domain analysis	591
9.1 Response spectrum analysis	591
9.2 Fourier analysis	602
9.3 Harmonic vibration analysis	605
9.4 Random vibration analysis	612
9.5 SDOF system response.....	617
10. Fracture mechanics	629
10.1 Overview	629
10.2 The line contour integration method	631
10.3 The virtual crack extension method	635
10.3.1 J-integral and energy release rate	635
10.3.2 Virtual material shifts.....	640
10.3.3 Results	642
10.4 Crack propagation	644
10.5 Recommendations on the use of elements and meshes	650
10.6 User-supplied fracture mechanics	653
10.6.1 General considerations	653
10.6.2 Usage of the user-supplied fracture mechanics subroutines.....	653
11. Additional capabilities	655
11.1 Substructuring	655
11.1.1 Substructuring in linear analysis	655
11.1.2 Substructuring in nonlinear analysis	657
11.2 Cyclic symmetry analysis.....	659
11.3 Reactions calculation.....	663
11.4 Element birth and death option	664
11.5 Element "death upon rupture"	671
11.6 Initial conditions.....	673
11.6.1 Initial displacements, velocities and accelerations	673
11.6.2 Initial temperatures and temperature gradients	673
11.6.3 Initial pipe internal pressures	674
11.6.4 Initial strains.....	674
11.6.5 Initial stresses	677
11.7 Strain energy densities.....	680
11.8 Element group inertial properties	681
11.9 Restart option	681
11.10 Parallel processing.....	684
11.11 Usage of memory and disk storage	685
11.12 Remeshing options	688
11.13 Analysis zooming	690

11.14 Stiffness stabilization	691
12. Postprocessing considerations	695
12.1 Calculation of results within the AUI.....	695
12.1.1 What results are available.....	701
12.1.2 Where results can be obtained.....	735
12.1.3 How results are evaluated (including smoothing)	745
12.1.4 When (for which load step, etc.) are results available.....	748
12.2 Evaluation of meshes used	752
Index	753

1. Introduction

1.1 Objective of this manual

The objective of this manual is to give a concise summary and guide for study of the theoretical basis of the finite element computer program ADINA Solids & Structures (ADINA in short). The program ADINA is used for displacement and stress analysis.

Since a large number of analysis options is available in this computer program, a user might well be initially overwhelmed with the different analysis choices and the theoretical bases of the computer program. A significant number of publications referred to in the text (books, papers and reports) describe in detail the finite element analysis procedures used in the program. However, this literature is very comprehensive and frequently provides more detail than the user needs to consult for the effective use of ADINA. Furthermore, it is important that a user can identify easily which publication should be studied if more information is desired on a specific analysis option.

The intent with this Theory and Modeling Guide is

- To provide a document that summarizes the methods and assumptions used in the computer program ADINA
- To provide specific references that describe the finite element procedures in more detail.

Hence, this manual has been compiled to provide a bridge between the actual practical use of the ADINA system and the theory documented in various publications. Much reference is made to the book *Finite Element Procedures* (ref. KJB) and to other publications but we endeavored to be specific when referencing these publications so as to help you to find the relevant information.

ref. K.J. Bathe, *Finite Element Procedures*, Prentice Hall, Englewood Cliffs, NJ, 1996.

Following this introductory chapter, Chapter 2 describes the

elements available in ADINA. The formulations used for these elements have been proven to be reliable and efficient in linear, large displacement, and large strain analyses. Chapter 3 describes the material models available in ADINA. Chapter 4 describes the different contact formulations and provides modeling tips for contact problems. Loads, boundary conditions and constraints are addressed in Chapter 5. Eigenvalue type analyses such as linearized buckling and frequency analyses are described in Chapter 6. Chapter 7 provides the formulations used for static and implicit dynamic analysis, while Chapter 8 deals with explicit dynamic analysis. Several frequency domain analysis tools are detailed in Chapter 9. Fracture mechanics features are described in Chapter 10. Additional capabilities such as substructures, cyclic symmetry, initial conditions, parallel processing and restarts are discussed in Chapter 11. Finally, some post-processing considerations are provided in Chapter 12.

We intend to update this report as we continue our work on the ADINA system. If you have any suggestions regarding the material discussed in this manual, we would be glad to hear from you.

1.2 Supported computers and operating systems

Table 1.2-1 shows all supported computers/operating systems.

1.3 Units

When using the ADINA system, it is important to enter all physical quantities (lengths, forces, masses, times, etc.) using a consistent set of units. For example, when working with the SI system of units, enter lengths in meters, forces in Newtons, masses in kg, times in seconds. When working with other systems of units, all mass and mass-related units must be consistent with the length, force and time units. For example, in the USCS system (USCS is an abbreviation for the U.S. Customary System), when the length unit is inches, the force unit is pound and the time unit is second, the mass unit is lb-sec²/in, not lb.

Table 1.3-1 gives some of the more commonly used units needed for ADINA input.

Table 1.2-1: Supported computers/operating systems

Platform/operating system	32-bit version	64-bit version	Parallelization options ¹
HP/HP-UX 11, PA-RISC computers	No	Yes, ADINA-M	Assembly, solver
HP/HP-UX 11.22, Itanium computers	No	Yes, ADINA-M	Assembly, solver
Linux kernel 2.4.0 and higher, x86 computers ^{2,3}	Yes, ADINA-M	No	Assembly, solver
Linux kernel 2.4.0 and higher, Itanium computers, (including SGI Altix) ⁴	The x86 version of the AUI is provided so that ADINA-M can be used.	Yes	Assembly, solver
Linux kernel 2.4.21 and higher, Intel x86_64 computers and AMD Opteron computers ⁵	No	Yes, ADINA-M	Assembly, solver
IBM/AIX 5.1	No	Yes, ADINA-M	Solver
SGI/IRIX 6.5.16m and higher	No	Yes, ADINA-M	Assembly, solver
Sun/SunOS 5.8 (Solaris 8)	No	Yes, ADINA-M	Solver
Windows 98, NT, 2000, XP ⁶	Yes, ADINA-M	No	Solver (not for Windows 98)

ADINA-M is supported only for the indicated program versions.

- 1) Only ADINA and ADINA-T have parallelized assembly.
- 2) Users of the user-supplied options for Linux must use the Intel compilers when using ADINA system 8.0 and higher.
- 3) The Linux version for x86 computers requires glibc 2.3.2 or higher (Red Hat 9.0 or higher, or equivalent).
- 4) The Linux version for Itanium computers requires glibc 2.2.4 or higher. SGI Altix computers require glibc-2.3.2 or higher (SGI ProPack 3).
- 5) The Linux version for x86_64 and AMD Opteron computers requires glibc 2.3.2 or higher.
- 6) The 3GB address space option is supported.

Table 1.3-1: Units

	SI	SI (mm)	USCS (inch)	USCS (kip)
length	meter (m)	millimeter (mm)	inch (in)	inch (in)
force	Newton (N)	Newton (N)	pound (lb)	kip (1000 lb)
time	second (s)	second (s)	second (sec)	second (sec)
mass	kilogram (kg) = N·s ² /m	N·s ² /mm	lb·sec ² /in (note 1)	kip·sec ² /in
pressure, stress, Young's modulus, etc.	Pascal (Pa) = N/m ²	N/mm ²	psi = lb/in ²	ksi = kip/in ²
density	kg/m ³	N·s ² /mm ⁴	lb·sec ² /in ⁴ (note 2)	kip·sec ² /in ⁴

- 1) A body that weighs 1 lb has a mass of $1/386.1 = 0.002590 \text{ lb}\cdot\text{sec}^2/\text{in}$.
 2) A body with weight density 1 lb/in³ has a density of $0.002590 \text{ lb}\cdot\text{sec}^2/\text{in}^4$.

1.4 ADINA System documentation

At the time of printing of this manual, the following documents are available with the ADINA System:

Installation Notes

Describes the installation of the ADINA System on your computer.

ADINA User Interface Command Reference Manual Volume I: ADINA Solids & Structures Model Definition, Report ARD 05-2, October 2005

Volume II: ADINA Heat Transfer Model Definition, Report ARD 05-3,

October 2005

Volume III: ADINA CFD Model Definition, Report ARD 05-4,

October 2005

Volume IV: Display Processing, Report ARD 05-5,

October 2005

These documents describe the AUI command language. You use the AUI command language to write batch files for the AUI.

ADINA User Interface Primer, Report ARD 05-6, October 2005

Tutorial for the ADINA User Interface, presenting a sequence of worked examples which progressively instruct you how to effectively use the AUI.

Theory and Modeling Guide

Volume I: ADINA Solids & Structures, Report ARD 05-7,

October 2005

Volume II: ADINA Heat Transfer, Report ARD 05-8,

October 2005

Volume III: ADINA CFD & FSI, Report ARD 05-9, October 2005

Provides a concise summary and guide for the theoretical basis of the analysis programs ADINA, ADINA-T, ADINA-F, ADINA-FSI and ADINA-TMC. The manuals also provide references to other publications which contain further information, but the detail contained in the manuals is usually sufficient for effective understanding and use of the programs.

ADINA Verification Manual, Report ARD 05-10, October 2005

Presents solutions to problems which verify and demonstrate the usage of the ADINA System. Input files for these problems are distributed along with the ADINA System programs.

TRANSOR for PATRAN Users Guide, Report ARD 05-14,

October 2005

Describes the interface between the ADINA System and MSC.Patran. The ADINA Preference, which allows you to

perform pre-/post-processing and analysis within the Patran environment, is described.

**TRANSOR for I-DEAS Users Guide, Report ARD 05-15,
October 2005**

Describes the interface between the ADINA System and UGS I-deas. The fully integrated TRANSOR graphical interface is described, including the input of additional data not fully described in the I-deas database.

ADINA System 8.3 Release Notes, October 2005

Provides update pages for all ADINA System documentation published before October 2005 and a description of the new and modified features of ADINA System 8.3.

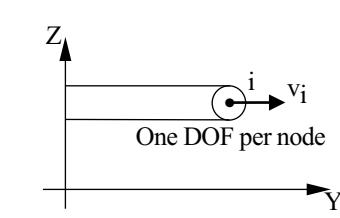
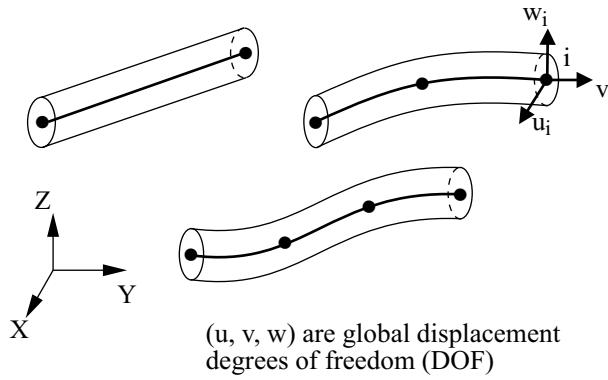
2. Elements

2.1 Truss and cable elements

- This chapter outlines the theory behind the different element classes, and also provides details on how to use the elements in modeling. This includes the materials that can be used with each element type, their applicability to large displacement and large strain problems, their numerical integration, etc.

2.1.1 General considerations

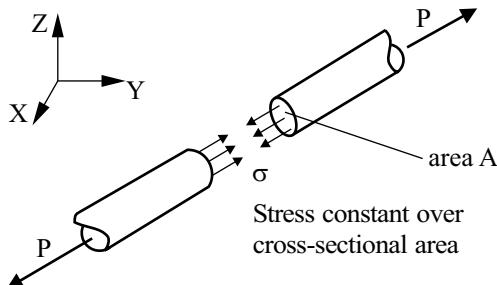
- The truss elements can be employed as 2-node, 3-node and 4-node elements, or as a 1-node ring element. Fig. 2.1-1 shows the elements available in ADINA.



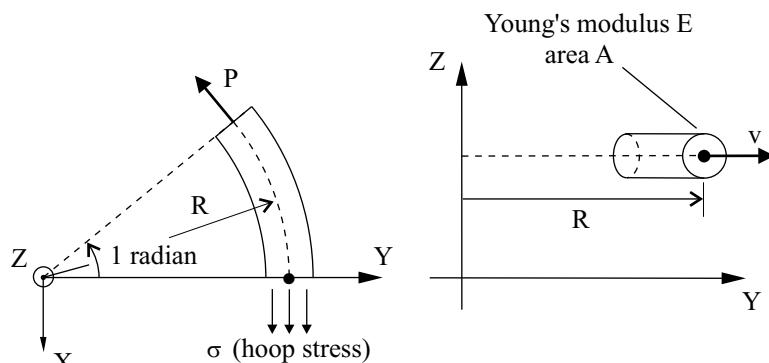
(b) 1-node ring element

Figure 2.1-1: Truss elements available in ADINA

- ref. KJB
Sections
5.3.1,
6.3.3*
- Note that the only force transmitted by the truss element is the longitudinal force as illustrated in Fig. 2.1-2. This force is constant in the 2-node truss and the ring element, but can vary in the 3- and 4-node truss (cable) elements.



(a) 2- to 4-node elements



$$\text{Hoop strain} = v/R \quad \text{Elastic stiffness} = EA/R \text{ (one radian)}$$

$$\text{Circumferential force } P = \sigma A$$

(b) Ring element

Figure 2.1-2: Stresses and forces in truss elements

- The ring element is formulated for one radian of the structure (as is the axisymmetric 2-D solid element, see Section 2.2.1), and this formulation is illustrated in Fig. 2.1-2.
- The local node numbering and natural coordinate system for the 2-node, 3-node and 4-node truss elements are shown in Fig. 2.1-3.

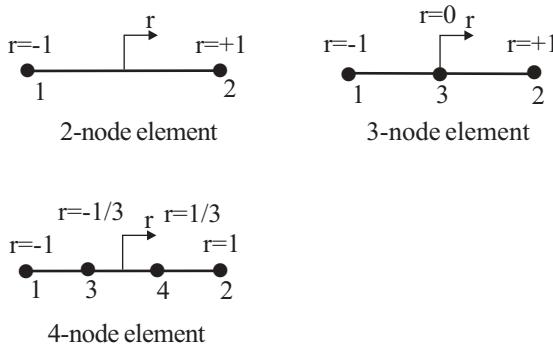


Figure 2.1-3: Local node numbering; natural coordinate system

2.1.2 Material models and formulations

- The truss element can be used with the following material models: **elastic-isotropic**, **nonlinear-elastic**, **plastic-bilinear**, **plastic-multilinear**, **thermo-isotropic**, **thermo-plastic**, **creep**, **plastic-creep**, **multilinear-plastic-creep**, **creep-variable**, **plastic-creep-variable**, **multilinear-plastic-creep-variable**, **viscoelastic**, **shape-memory alloy**.
- The truss element can be used with the **small or large displacement** formulations. In the small displacement formulation, the displacements and strains are assumed infinitesimally small. In the large displacement formulation, the displacements and rotations can be very large. In all cases, the cross-sectional area of the element is assumed to remain unchanged, and the strain is equal to the longitudinal displacement divided by the original length.

All of the material models in the above list can be used with either formulation. The use of a linear material with the small displacement formulation corresponds to a linear formulation, and the use of a nonlinear material with the small displacement formulation corresponds to a materially-nonlinear-only formulation.

2.1.3 Numerical integration

- The integration point labeling of the truss (cable) elements is given in Fig. 2.1-4.

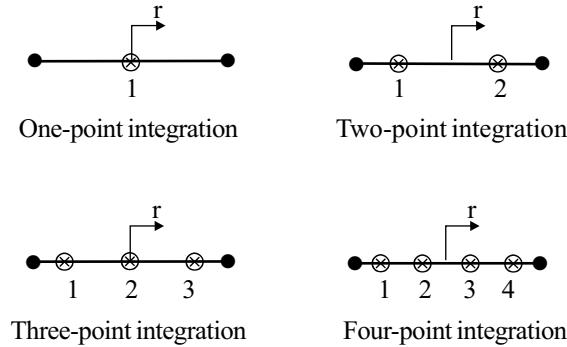


Figure 2.1-4: Integration point locations

- Note that when the material is temperature-independent, the 2-node truss and ring elements require only 1-point Gauss numerical integration for an exact evaluation of the stiffness matrix, because the force is constant in the element. However, 2-point Gauss numerical integration may be appropriate when a temperature-dependent material is used because, due to a varying temperature, the material properties can vary along the length of an element.

2.1.4 Mass matrices

- The consistent mass matrix is calculated using Eq. (4.25) in ref. KJB, p. 165, which accurately computes the consistent mass distribution.

The lumped mass matrix is formed by dividing the element's mass among its nodes. The mass assigned to each node is $M \cdot \left(\frac{\ell_i}{L} \right)$,

where M = total mass, L = total element length, ℓ_i = fraction of the total element length associated with element node i (i.e., for the 2-node truss element, $\ell_1 = \frac{L}{2}$ and $\ell_2 = \frac{L}{2}$, which for the 4-node truss $\ell_1 = \ell_2 = \frac{L}{6}$ and $\ell_3 = \ell_4 = \frac{L}{3}$). The element has no rotational mass.

2.1.5 Element output

- Each element outputs, at its integration points, the following information to the porthole file, based on the material model. This information is accessible in the AUI using the given variable names.

Elastic-isotropic, nonlinear-elastic: FORCE-R, STRESS-RR,
STRAIN-RR

Elastic-isotropic with thermal effects: FORCE-R, STRESS-RR,
STRAIN-RR, THERMAL_STRAIN,
ELEMENT_TEMPERATURE

Thermo-isotropic: FORCE-R, STRESS-RR, STRAIN-RR,
ELEMENT_TEMPERATURE

Plastic-bilinear, plastic-multilinear: PLASTIC_FLAG,
FORCE-R, STRESS-RR, STRAIN-RR,
PLASTIC_STRAIN-RR

Thermo-plastic, creep, plastic-creep, multilinear-plastic-creep:
PLASTIC_FLAG, NUMBER_OF_SUBINCREMENTS,
FORCE-R, STRESS-RR, STRAIN-RR,
ACCUMULATED_EFFECTIVE_PLASTIC_STRAIN,
PLASTIC_STRAIN-RR, CREEP_STRAIN-RR,
THERMAL_STRAIN-RR, ELEMENT_TEMPERATURE,
FE_EFFECTIVE_STRESS, YIELD_STRESS,
EFFECTIVE_CREEP_STRAIN

Viscoelastic: PLASTIC_FLAG, STRESS-RR, STRAIN-RR,
THERMAL_STRAIN-RR, ELEMENT_TEMPERATURE

See Section 12.1.1 for the definitions of those variables that are not self-explanatory.

2.1.6 Recommendations on use of elements

- Of the 2-, 3- and 4-node truss elements, the 2-node element is usually most effective and can be used in modeling truss structures,

EA/L where L is the length of the element) and nonlinear elastic gap elements (see Section 3.3).

- The 3- and 4-node elements are employed to model cables and steel reinforcement in reinforced concrete structures that are modeled with higher-order continuum or shell elements. In this case, the 3- and 4-node truss elements are compatible with the continuum or shell elements.
- The internal nodes for the 3-node and 4-node truss elements are usually best placed at the mid- and third-points, respectively, unless a specific predictive capability of the elements is required (see ref. KJB, Examples 5.2 and 5.17, pp. 348 and 370).

2.1.7 Rebar elements

- The AUI can generate truss elements that are connected to the 2D or 3D solid elements in which the truss elements lie. There are several separate cases, corresponding to the type of the truss element group and the type of the solid element group in which the truss elements lie.

Axisymmetric truss elements, axisymmetric solid elements: The AUI can connect axisymmetric truss elements that exist in the model prior to data file generation to the axisymmetric 2D elements in which the truss elements lie. The AUI does this connection during data file generation as follows. For each axisymmetric truss element that lies in an axisymmetric 2D element, constraint equations are defined between the axisymmetric truss element node and the three closest corner nodes of the 2D element.

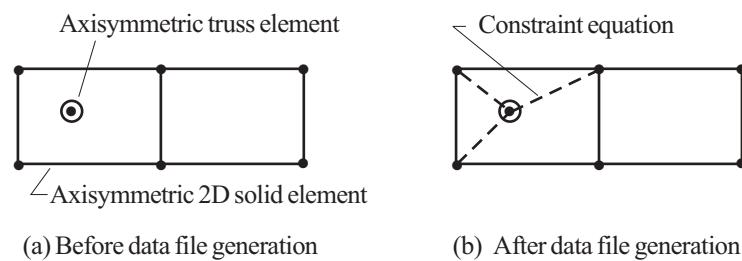


Figure 2.1-5: Rebar in axisymmetric truss, 2D solid elements

3D truss elements, planar 2D solid elements: The AUI can generate 3D truss elements and then connect the truss elements to the planar 2D elements in which the truss elements lie. The AUI does this during data file generation as follows. For each rebar line, the AUI finds the intersections of the rebar line and the sides of the 2D elements. The AUI then generates nodes at these intersections and generates truss elements that connect the successive nodes. The AUI also defines constraint equations between the generated nodes and the corner nodes of the 2D elements.

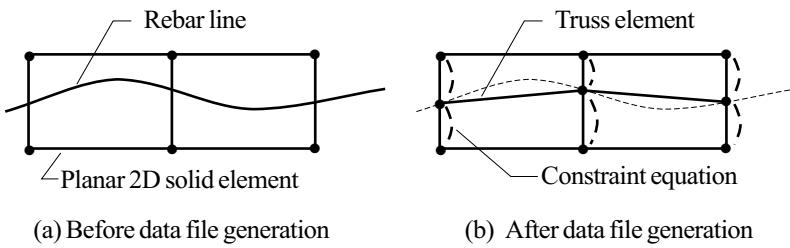


Figure 2.1-6: Rebar in 3D truss, planar 2D solid elements

3D truss elements, 3D solid elements: The AUI can generate 3D truss elements and then connect the truss elements to the 3D elements in which the truss elements lie. The AUI does this during data file generation as follows. For each rebar line, the AUI finds the intersections of the rebar line and the faces of the 3D elements. The AUI then generates nodes at these intersections and generates truss elements that connect the successive nodes. The AUI also defines constraint equations between the generated nodes and three corner nodes of the 3D element faces.

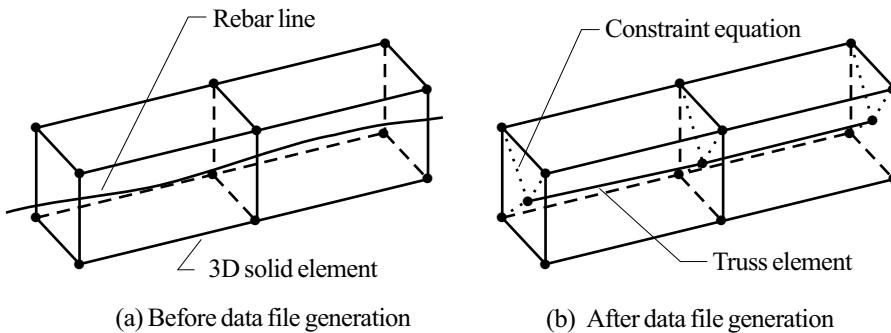


Figure 2.1-7: Rebar in 3D truss, 3D solid elements

- The rebar option is intended for use with lower-order solid elements (that is, the solid elements should not have mid-side nodes).
 - For 3D truss element generation, if the rebar lines are curved, you should specify enough subdivisions on the rebar lines so that the distance between subdivisions is less than the solid element size. The more subdivisions that are on the rebar lines, the more accurately the AUI will compute the intersections of the rebar lines and the solid element sides or faces.
 - To request the connection:

AUI: Click the Define Element Group icon to open the Define Element Group window. In this window, check that the Element Type has been set to “Truss”. Click on the Advanced tab. Select the “Use as Rebars” option from the Element Option drop-down, and then fill in the Rebar-Line Label box below the Element Option drop-down box.

Command Line: Set OPTION=REBAR in the EGROUP TRUSS command. For 3D truss elements, use the REBAR-LINE command and RB-LINE parameter of the EGROUP TRUSS command to define the rebar lines.

2.2 Two-dimensional solid elements

2.2.1 General considerations

- The following kinematic assumptions are available for two-dimensional elements in ADINA: **plane stress**, **plane strain**, **axisymmetric**, **generalized plane strain** and **3-D plane stress** (membrane). Figures 2.2-1 and 2.2-2 show some typical 2-D elements and the assumptions used in the formulations.

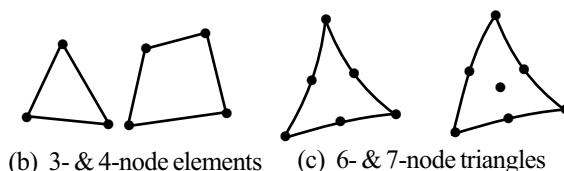
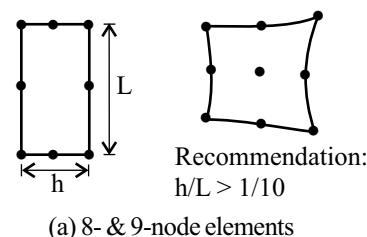


Figure 2.2-1: 2-D solid elements

- The 3-D plane stress element can lie in general three-dimensional space.
- The plane stress, plane strain, generalized plane strain, and axisymmetric elements must be defined in the YZ plane. The axisymmetric element must, in addition, lie in the +Y half plane. However, these elements can be combined with any other elements available in ADINA.
- The axisymmetric element provides for the stiffness of one radian of the structure. Hence, when this element is combined with other elements, or when concentrated loads are defined, these must also refer to one radian, see ref. KJB, Examples 5.9 and 5.10, p. 356.

ref. KJB
Sections 5.3.1
and 5.3.2

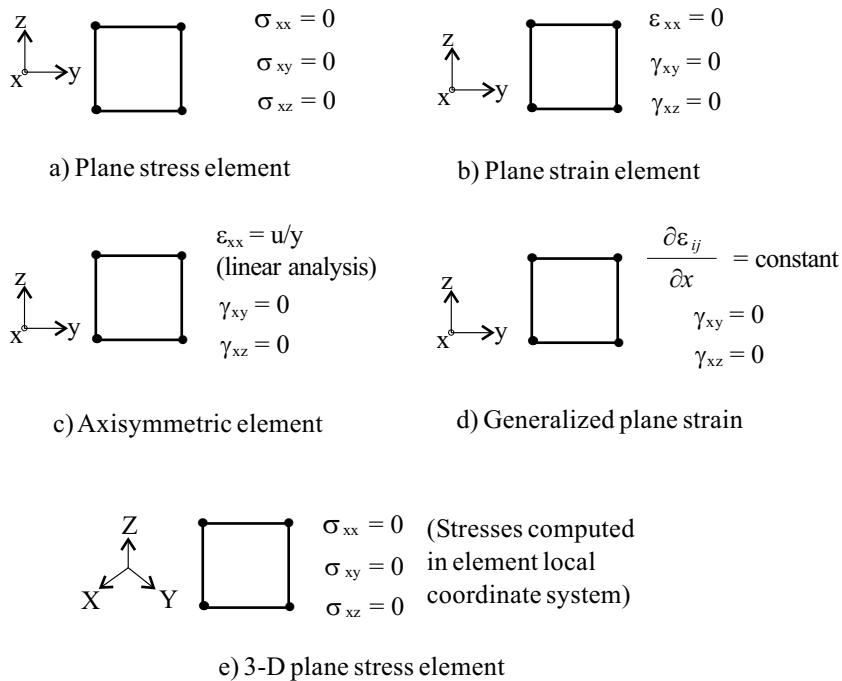
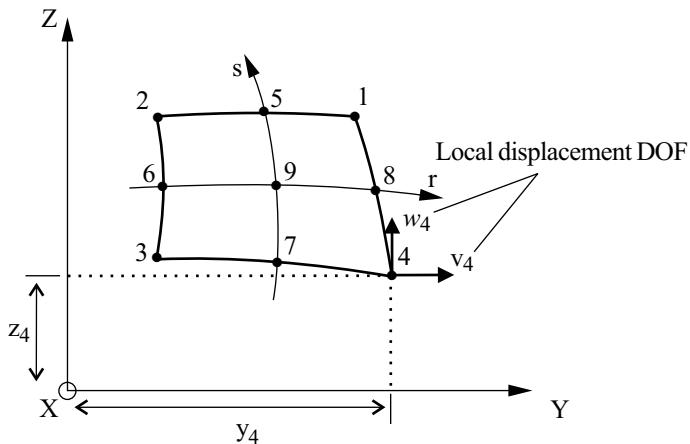


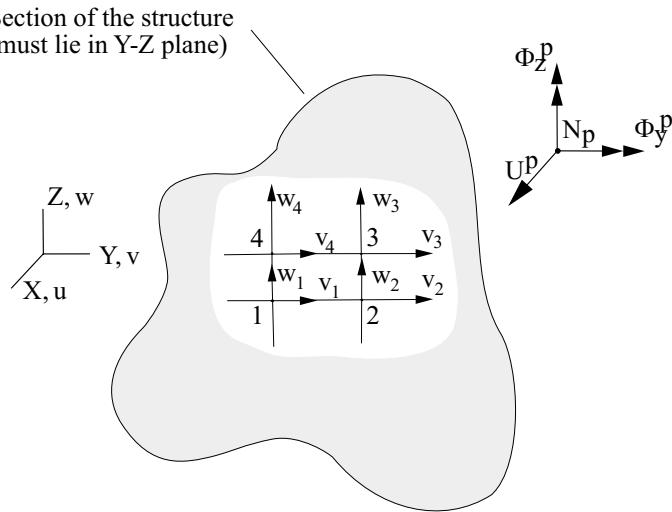
Figure 2.2-2: Basic assumptions in 2-D analysis

- The plane strain element provides for the stiffness of unit thickness of the structure.
- The elements usually used in ADINA are isoparametric displacement-based finite elements, and their formulation is described in detail in ref. KJB, Section 5.3.
- The basic finite element assumptions are, see Fig. 2.2-3, for the coordinates

$$y = \sum_{i=1}^q h_i y_i ; \quad z = \sum_{i=1}^q h_i z_i$$



a) Plane strain, plane stress, and axisymmetric elements



The coordinates of the auxiliary node N_p are (x_p, y_p, z_p) .

The element has unit length in the X direction.

b) Generalized plane strain element

Figure 2.2-3: Conventions used for the nodal coordinates and displacements of the 2-D solid element

for the displacements in plane stress, plane strain, and axisymmetric elements

$$v = \sum_{i=1}^q h_i v_i ; \quad w = \sum_{i=1}^q h_i w_i$$

for the displacements in generalized plane strain elements

$$u = x U^p - x(y - y_p^p) \Phi_z^p + x(z - z_p) \Phi_y^p$$

$$v = \sum_{i=1}^q h_i v_i + \frac{1}{2} x^2 \Phi_z^p$$

$$w = \sum_{i=1}^q h_i w_i - \frac{1}{2} x^2 \Phi_z^p$$

where

$h_i(r,s)$ = interpolation function corresponding to node i

(r,s) = isoparametric coordinates

q = number of element nodes, $4 \leq q \leq 9$, excluding auxiliary node in generalized plane strain

y_i, z_i = nodal point coordinates

v_i, w_i = nodal point displacements

U^p, Φ_y^p, Φ_z^p = degrees of freedom of the auxiliary node

x_p, y_p, z_p = coordinates of the auxiliary node

- In addition to the displacement-based elements, special mixed-interpolated elements are also available, in which the displacements and pressure are interpolated separately. These elements are effective and should be preferred in the analysis of incompressible media and inelastic materials (specifically for materials in which Poisson's ratio is close to 0.5, for rubber-like materials and for elasto-plastic materials). The 4-node element (1 pressure degree of freedom) and 9-node element (3 pressure degrees of freedom) are recommended for such analyses. Note that these mixed-interpolated elements are useful (and available in ADINA) only for plane strain, generalized plane strain, or axisymmetric analysis.

ref. T. Sussman and K.J. Bathe, "A Finite Element Formulation for Nonlinear Incompressible Elastic and Inelastic Analysis," *J. Computers and Structures*, Vol. 26, No. 1/2, pp. 357-409, 1987.

- In addition to the displacement-based and mixed-interpolated elements, ADINA also includes the possibility of including incompatible modes in the formulation of the 4-node element. Within this element, additional displacement degrees of freedom are introduced. These additional displacement degrees of freedom are not associated with nodes; therefore the condition of displacement compatibility between adjacent elements is not satisfied in general. However the additional displacement degrees of freedom increase the flexibility of the element, especially in bending situations.

Figure 2.2-4 shows a situation in which the incompatible modes elements give an improved solution.

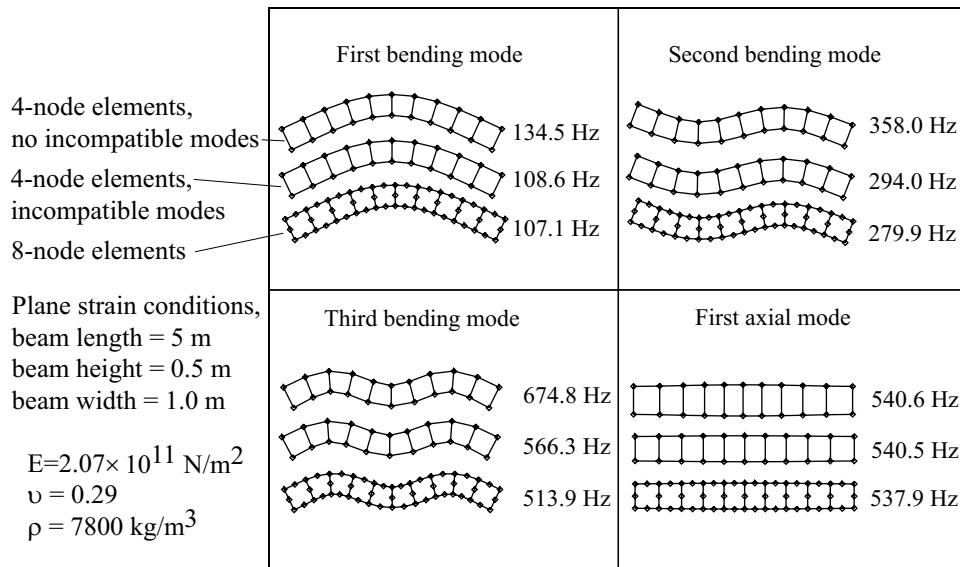


Figure 2.2-4: Frequency analysis of a free-free beam with three types of finite elements

For theoretical considerations, see reference KJB, Section 4.4.1. Note that these elements are formulated to pass the patch test. Also note that element distortions deteriorate the element performance.

The incompatible modes feature cannot be used in conjunction with the mixed-interpolation formulation.

The incompatible modes feature is only available for the 4-node element; in particular note that the incompatible modes feature is not available for the 3-node triangular element.

- The elements can be used with 3 to 9 nodes. The interpolation functions are defined in ref. KJB, Fig. 5.4, p. 344.

*ref. KJB
Section 5.3.2*

- Triangular elements are formed in ADINA by assigning the same global node number to all the nodes located on one side. It is recommended that this "collapsed" side be the one with local end nodes 1 and 4.

The triangular elements can be of different types:

- ▶ A collapsed 6-node triangular element with the strain singularity $\frac{1}{\sqrt{r}}$ (see Chapter 9)
- ▶ A collapsed 8-node element with the strain singularity $\frac{1}{r}$
(see Chapter 9)
- ▶ A collapsed 6-node spatially isotropic triangular element
- ▶ A collapsed 7-node spatially isotropic triangular element

These element types are described on pp. 369-370 of ref. KJB. The strain singularities are obtained by collapsing one side of an 8-node rectangular element and not correcting the interpolation functions.

The 6-node spatially isotropic triangle is obtained by correcting the interpolation functions of the collapsed 8-node element. It then uses the same interpolation functions for each of the 3 corner nodes and for each of the midside nodes. Note that when the corrections to the interpolation functions are employed, only the element side with end nodes 1 and 4 can be collapsed.

The 3-node triangular element is obtained by collapsing one

side of the 4-node element. This element exhibits the constant strain conditions (except that the hoop strain in axisymmetric analysis varies over the element) but is usually not effective.

- Linear and nonlinear fracture mechanics analysis of stationary or propagating cracks can be performed with two-dimensional elements (see Chapter 9).

2.2.2 Material models and formulations

- The 2-D elements can be used with the following material models: **elastic-isotropic**, **elastic-orthotropic** (with or without wrinkling), **plastic-bilinear**, **plastic-multilinear**, **Drucker-Prager**, **Mroz-bilinear**, **plastic-orthotropic**, **thermo-isotropic**, **thermo-orthotropic**, **thermo-plastic**, **creep**, **plastic-creep**, **multilinear-plastic-creep**, **concrete**, **curve-description**, **Ogden**, **Mooney-Rivlin**, **Arruda-Boyce**, **hyper-foam**, **creep-variable**, **plastic-creep-variable**, **multilinear-plastic-creep-variable**, **Gurson-plastic**, **Cam-clay**, **Mohr-Coulomb**, **viscoelastic**, **creep irradiation**, **gasket**, **shape-memory**, **alloy user-supplied**.

- In plane strain, generalized plane strain and axisymmetric analysis, you should use the mixed interpolation formulation whenever using the plastic-bilinear, plastic-multilinear, Mroz-bilinear, plastic-orthotropic, thermo-plastic, creep, plastic-creep, multilinear-plastic-creep, Ogden, Mooney-Rivlin, Arruda-Boyce, creep-variable, plastic-creep-variable, multilinear-plastic-creep-variable or viscoelastic material models; or when using the elastic-isotropic material and the Poisson's ratio is close to 0.5.

The AUI automatically chooses the mixed interpolation formulation whenever the Ogden, Mooney-Rivlin or Arruda-Boyce material models are used. For the other material types listed in the previous paragraph, you should explicitly select the mixed interpolation formulation.

- The two-dimensional elements can be used with a **small displacement/small strain**, **large displacement/small strain** or a **large displacement/large strain** formulation.

The small displacement/small strain and large displacement/small strain formulations can be used with any material model, except for the Ogden, Mooney-Rivlin, Arruda-

Boyce and hyper-foam material models. The use of a linear material with the small displacement/small strain formulation corresponds to a linear formulation, and the use of a nonlinear material with the small displacement/small strain formulation corresponds to a materially-nonlinear-only formulation. The program uses the TL (total Lagrangian) formulation when you choose a large displacement/small strain formulation.

The large displacement/large strain formulations can be used with the plastic-bilinear, plastic-multilinear, Mroz-bilinear, thermo-plastic, creep, plastic-creep, multilinear-plastic-creep, creep-variable, plastic-creep-variable, multilinear-plastic-creep-variable, viscoelastic and user-supplied material models. The program uses a ULH formulation when you choose a large displacement/large strain formulation with these material models.

A large displacement/large strain formulation is used with the Ogden, Mooney-Rivlin, Arruda-Boyce and hyper-foam material models. The program uses a TL (total Lagrangian) formulation in this case.

*ref. KJB
Sections 6.2 and
6.3.4*

- The basic continuum mechanics formulations are described in ref. KJB, pp. 497-537, and the finite element discretization is given in ref. KJB pp. 538-542, 549-555.

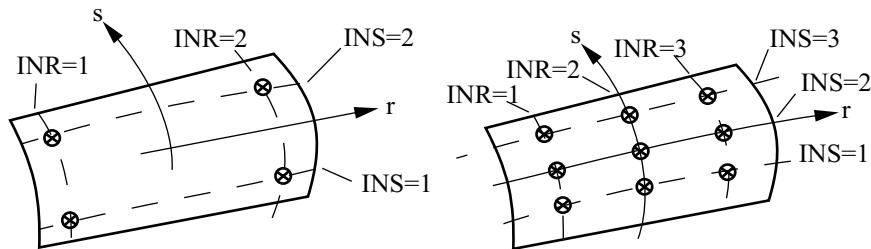
*ref. KJB
Section 6.8.1*

- Note that all these formulations can be mixed in one same finite element mesh. If the elements are initially compatible, then they will remain compatible throughout the complete analysis.

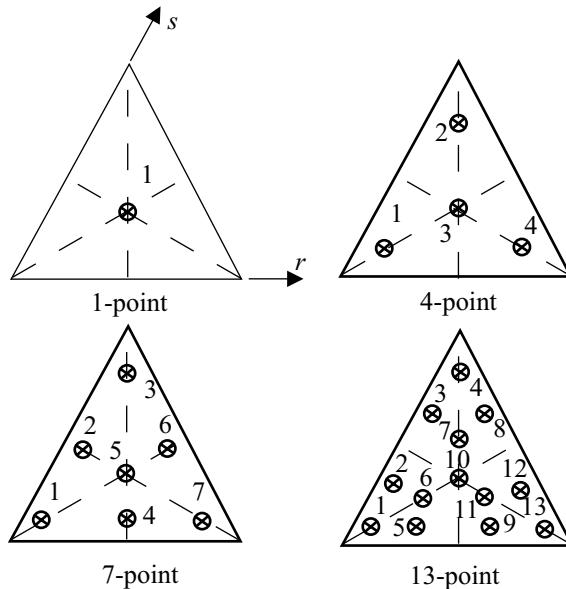
2.2.3 Numerical integration

*ref. KJB
Sections 5.5.3,
5.5.4 and 5.5.5*

- For the calculation of all element matrices and vectors, numerical Gauss integration is used. You can use from 2×2 to 6×6 Gauss integration. The numbering convention and the location of the integration points are given in Fig. 2.2-5 for 2×2 and 3×3 Gauss integration. The convention used for higher orders is analogous. The default Gauss integration orders for rectangular elements are 2×2 for 4-node elements and 3×3 otherwise.



a) Rectangular elements



b) Triangular elements

Figure 2.2-5: Integration point positions for 2-D solid elements

- For the 8-node rectangular element, the use of 2×2 Gauss integration corresponds to a slight under-integration and one spurious zero energy mode is present. In practice, this particular kinematic mode usually does not present any problem in linear analysis (except in the analysis of one-element cases, see ref. KJB, Fig. 5.40, p. 472) and thus 2×2 Gauss integration can be employed with caution for the 8-node elements.

- The 3-node, 6-node and 7-node triangular elements are spatially isotropic with respect to integration point locations and interpolation functions (see Figure 2.2-5).

- The integration order used for triangular elements depends upon the integration order used for rectangular elements as follows.

When rectangular elements use 2×2 Gauss integration, triangular elements use 4-point Gauss integration; when rectangular elements use 3×3 Gauss integration, triangular elements use 7-point Gauss integration; when rectangular elements use 4×4 Gauss integration or higher, triangular elements use 13-point Gauss integration.

However, for 3-node triangular elements, when the axisymmetric subtype is not used and when incompatible modes are not used, then 1-point Gauss integration is always used.

- Note that in geometrically nonlinear analysis, the spatial positions of the Gauss integration points change continuously as the element undergoes deformations, but throughout the response the same material particles are at the integration points.

ref. KJB
Section 6.8.4

- The order of numerical integration in the large displacement elastic analysis is usually best chosen to be equal to the order used in linear elastic analysis. Hence, the program default values should usually be employed. However, in inelastic analysis, a higher order integration should be used when the elements are used to model thin structures (see Fig. 6.25, p. 638, ref. KJB), or large deformations (large strains) are anticipated.

2.2.4 Mass matrices

- The consistent mass matrix is always calculated using either 3×3 Gauss integration for rectangular elements or 7-point Gauss integration for triangular elements.
- The lumped mass matrix of an element is formed by dividing the element's mass M equally among its n nodes. Hence, the mass assigned to each node is M/n . No special distributory concepts are employed to distinguish between corner and midside nodes, or to account for element distortion.

- Note that n is the number of distinct non-repeated nodes in the element. Hence, when a quad element side is collapsed to a single node, the total mass of the element is equally distributed among the final nodes of the triangular configuration.

2.2.5 Element output

You can request that ADINA either print or save stresses or forces.

Stresses: Each element outputs, at its integration points, the following information to the porthole file, based on the material model. This information is accessible in the AUI using the given variable names.

Notice that for the orthotropic material models, you can request that the stresses and strains be saved in the material coordinate system.

For the 3-D plane stress elements, results with indices YY, ZZ, etc are saved in the element local coordinate system (see Figure 2.2-6).

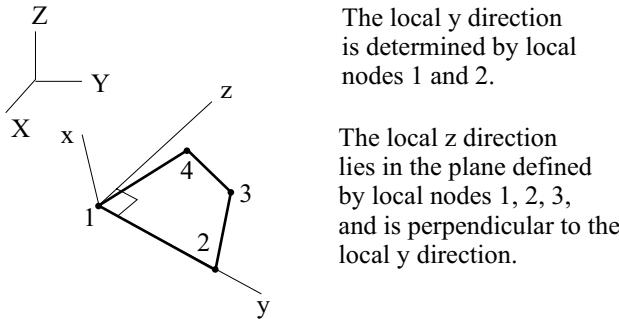


Figure 2.2-6: Local coordinate system for 3-D plane stress element

Elastic-isotropic, elastic-orthotropic (no wrinkling, results saved in global system): STRESS (XYZ) , STRAIN (XYZ)

Elastic-isotropic with thermal effects: STRESS (XYZ) , STRAIN (XYZ) , THERMAL_STRAIN, ELEMENT_TEMPERATURE

Elastic-orthotropic (no wrinkling, results saved in material system):
STRESS (ABC) , STRAIN (ABC)

Thermo-isotropic, thermo-orthotropic (results saved in global system): STRESS (XYZ) , STRAIN (XYZ) , THERMAL_STRAIN (XYZ) , ELEMENT_TEMPERATURE

Curve description: CRACK_FLAG, STRESS (XYZ) , STRAIN (XYZ) , FE_SIGMA-P1, FE_SIGMA-P2, FE_SIGMA-P1_ANGLE, GRAVITY_IN-SITU_PRESSURE, VOLUMETRIC_STRAIN

Concrete: CRACK_FLAG, STRESS (XYZ) , STRAIN (XYZ) , FE_SIGMA-P1, FE_SIGMA-P2, FE_SIGMA-P1_ANGLE, GRAVITY_IN-SITU_PRESSURE, ELEMENT_TEMPERATURE, THERMAL_STRAIN (XYZ)

Small strains: Plastic-bilinear, plastic-multilinear, Mroz-bilinear:
PLASTIC_FLAG, STRESS (XYZ) , STRAIN (XYZ) , PLASTIC_STRAIN (XYZ) , FE_EFFECTIVE_STRESS, YIELD_STRESS, ACCUMULATED_EFFECTIVE_PLASTIC_STRAIN, THERMAL_STRAIN (XYZ) , ELEMENT_TEMPERATURE

Large strains: Plastic-bilinear, plastic-multilinear, Mroz-bilinear:
PLASTIC_FLAG, STRESS (XYZ) , DEFORMATION_GRADIENT (XYZ) , FE_EFFECTIVE_STRESS, YIELD_STRESS, ACCUMULATED_EFFECTIVE_PLASTIC_STRAIN, THERMAL_STRAIN (XYZ) , ELEMENT_TEMPERATURE

Plastic-orthotropic (results saved in global system):
PLASTIC_FLAG, STRESS (XYZ) , STRAIN (XYZ) , PLASTIC_STRAIN (XYZ) , HILL_EFFECTIVE_STRESS, YIELD_STRESS, ACCUMULATED_EFFECTIVE_PLASTIC_STRAIN, THERMAL_STRAIN (XYZ) , ELEMENT_TEMPERATURE

Plastic-orthotropic (results saved in material system):
PLASTIC_FLAG, STRESS (ABC) , STRAIN (ABC) , PLASTIC_STRAIN (ABC) , HILL_EFFECTIVE_STRESS, YIELD_STRESS, ACCUMULATED_EFFECTIVE_PLASTIC_STRAIN, THERMAL_STRAIN (ABC) , ELEMENT_TEMPERATURE

Small strains: Drucker-Prager: PLASTIC_FLAG,
PLASTIC_FLAG-2, STRESS(XYZ), STRAIN(XYZ),
PLASTIC_STRAIN(XYZ), CAP_LOCATION,
YIELD_FUNCTION

Large strains: Drucker-Prager: PLASTIC_FLAG,
PLASTIC_FLAG-2, STRESS(XYZ),
DEFORMATION_GRADIENT(XYZ), CAP_LOCATION,
YIELD_FUNCTION

Small strains: Thermo-plastic, creep, plastic-creep, multilinear-plastic-creep, creep-variable, plastic-creep-variable, multilinear-plastic-creep-variable: PLASTIC_FLAG,
NUMBER_OF_SUBINCREMENTS, STRESS(XYZ),
STRAIN(XYZ), PLASTIC_STRAIN(XYZ),
CREEP_STRAIN(XYZ), THERMAL_STRAIN(XYZ),
ELEMENT_TEMPERATURE,
ACCUMULATED_EFFECTIVE_PLASTIC_STRAIN,
FE_EFFECTIVE_STRESS, YIELD_STRESS,
EFFECTIVE_CREEP_STRAIN

Large strains: Thermo-plastic, creep, plastic-creep, multilinear-plastic-creep, creep-variable, plastic-creep-variable, multilinear-plastic-creep-variable: PLASTIC_FLAG,
NUMBER_OF_SUBINCREMENTS, STRESS(XYZ),
DEFORMATION_GRADIENT(XYZ),
FE_EFFECTIVE_STRESS, YIELD_STRESS,
ACCUMULATED_EFFECTIVE_PLASTIC_STRAIN,
THERMAL_STRAIN(XYZ), ELEMENT_TEMPERATURE,
EFFECTIVE_CREEP_STRAIN

Mooney-Rivlin , Ogden, Arruda-Boyce, hyper-foam (strains saved): STRESS(XYZ), STRAIN(XYZ)

Mooney-Rivlin, Ogden, Arruda-Boyce, hyper-foam (deformation gradients saved): STRESS(XYZ),
DEFORMATION_GRADIENT(XYZ) .

Small strains: User-supplied: STRESS(XYZ), STRAIN(XYZ),
USER_VARIABLE_I

Large strains: User-supplied: STRESS (XYZ),
DEFORMATION_GRADIENT (XYZ), USER_VARIABLE_I

Elastic-orthotropic (wrinkling, results saved in global system):
WRINKLE_FLAG, STRESS (XYZ), STRAIN (XYZ)

Elastic-orthotropic (wrinkling, results saved in material system):
WRINKLE_FLAG, STRESS (ABC), STRAIN (ABC)

Thermo-orthotropic (results saved in material system):
STRESS (ABC), STRAIN (ABC),
THERMAL_STRAIN (ABC), ELEMENT_TEMPERATURE

Small strains: Gurson plastic: PLASTIC_FLAG,
STRESS (XYZ), STRAIN (XYZ),
PLASTIC_STRAIN (XYZ), FE_EFFECTIVE_STRESS,
YIELD_STRESS,
ACCUMULATED_EFFECTIVE_PLASTIC_STRAIN,
THERMAL_STRAIN (XYZ), ELEMENT_TEMPERATURE,
VOID_VOLUME_FRACTION

Large strains: Gurson plastic: PLASTIC_FLAG,
STRESS (XYZ), DEFORMATION_GRADIENT (XYZ),
FE_EFFECTIVE_STRESS, YIELD_STRESS,
ACCUMULATED_EFFECTIVE_PLASTIC_STRAIN,
THERMAL_STRAIN (XYZ), ELEMENT_TEMPERATURE,
VOID_VOLUME_FRACTION

Small strains, Cam-clay: PLASTIC_FLAG, STRESS (XYZ),
STRAIN (XYZ), PLASTIC_STRAIN (XYZ),
YIELD_SURFACE_DIAMETER_P, YIELD_FUNCTION,
MEAN_STRESS, DISTORTIONAL_STRESS,
VOLUMETRIC_STRAIN, VOID_RATIO,
EFFECTIVE_STRESS_RATIO, SPECIFIC_VOLUME

Large strains, Cam-clay: PLASTIC_FLAG, STRESS (XYZ),
DEFORMATION_GRADIENT (XYZ),
YIELD_SURFACE_DIAMETER_P, YIELD_FUNCTION,
MEAN_STRESS, DISTORTIONAL_STRESS,
VOLUMETRIC_STRAIN, VOID_RATIO,
EFFECTIVE_STRESS_RATIO, SPECIFIC_VOLUME

Mohr-Coulomb: PLASTIC_FLAG, STRESS(XYZ),
STRAIN(XYZ), PLASTIC_STRAIN(XYZ),
YIELD_FUNCTION

Small strains, viscoelastic: STRESS(XYZ), STRAIN(XYZ),
THERMAL_STRAIN(XYZ), ELEMENT_TEMPERATURE

Large strains, viscoelastic: STRESS(XYZ),
DEFORMATION_GRADIENT(XYZ),
THERMAL_STRAIN(XYZ), ELEMENT_TEMPERATURE

Small strains: creep-irradiation: STRESS(XYZ),
STRAIN(XYZ), PLASTIC_STRAIN(XYZ),
CREEP_STRAIN(XYZ), THERMAL_STRAIN(XYZ),
ELEMENT_TEMPERATURE,
ACCUMULATED_EFFECTIVE_PLASTIC_STRAIN,
EFFECTIVE_STRESS, YIELD_STRESS,
EFFECTIVE_CREEP_STRAIN,
IRRADIATION_STRAIN(XYZ).

Large strains: creep-irradiation: STRESS(XYZ),
DEFORMATION_GRADIENT(XYZ),
THERMAL_STRAIN(XYZ), ELEMENT_TEMPERATURE,
ACCUMULATED_EFFECTIVE_PLASTIC_STRAIN,
EFFECTIVE_STRESS, YIELD_STRESS,
EFFECTIVE_CREEP_STRAIN,
IRRADIATION_STRAIN(XYZ).

In the above lists,

STRESS(XYZ) = STRESS-YY, STRESS-ZZ, STRESS-YZ, STRESS-XX

STRESS(ABC) = STRESS-AA, STRESS-BB, STRESS-AB, STRESS-CC

with similar definitions for the other abbreviations used above. But the variable DEFORMATION_GRADIENT(XYZ) is interpreted as follows:

DEFORMATION_GRADIENT(XYZ) =
DEFORMATION_GRADIENT-YY,
DEFORMATION_GRADIENT-ZZ,

```
DEFORMATION_GRADIENT-XX,  
DEFORMATION_GRADIENT-YZ,  
DEFORMATION_GRADIENT-ZY
```

Also note that you can request stretches instead of deformation gradients, and in this case STRETCH (XYZ) replaces DEFORMATION_GRADIENT (XYZ) in the above lists.

See Section 12.1.1 for the definitions of those variables that are not self-explanatory.

- ref. KJB
Section 4.2.1,
6.6.3*
- In ADINA, the stresses are calculated using the strains at the point of interest. Hence, they are not spatially extrapolated or smoothed. However, the AUI can be employed to calculate smoothed stresses (see Section 12.1.3).
 - You can also request that strain energy densities be output along with the stresses.

Nodal forces: The nodal forces which correspond to the element stresses can also be requested in ADINA. This force vector is calculated in a linear analysis using

$$\mathbf{F} = \int_V \mathbf{B}^T \boldsymbol{\tau} dV$$

where \mathbf{B} is the element strain displacement matrix, and $\boldsymbol{\tau}$ is the stress vector. The integration is performed over the volume of the element; see Example 5.11 in ref. KJB, pp. 358-359.

The nodal forces are accessible in the AUI using the variable names NODAL_FORCE-X, NODAL_FORCE-Y, NODAL_FORCE-Z.

- The same relation is used for the element force calculation in a nonlinear analysis, except that updated quantities are used in the integration.

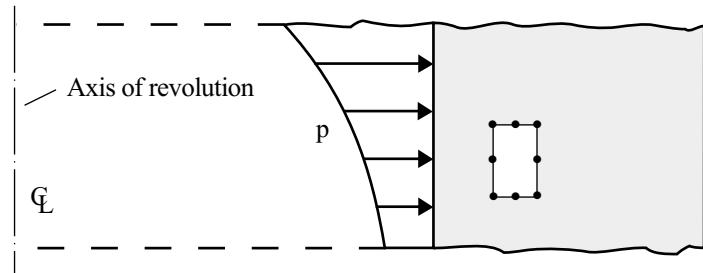
2.2.6 Recommendations on use of elements

- The 9-node element is usually most effective (except in explicit dynamic analysis).

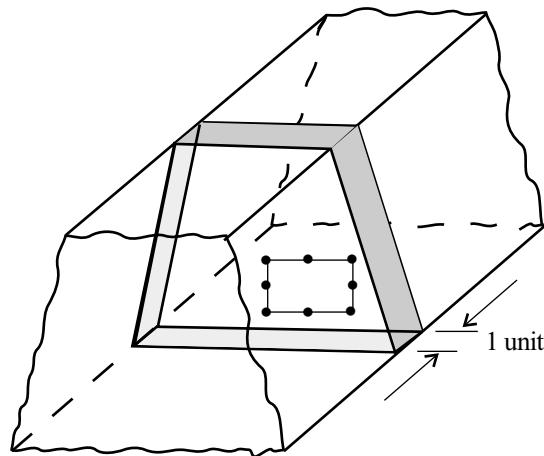
- The 8-node and 9-node elements can be employed for analysis of thick structures and solids, such as (see Fig. 2.2-7):
 - ▶ Thick cylinders (axisymmetric idealization)
 - ▶ Dams (plane strain idealization)
 - ▶ Membrane sheets (plane stress idealization)
 - ▶ Turbine blades (generalized plane strain idealization)

and for thin structures, such as (see Fig. 2.2-8):

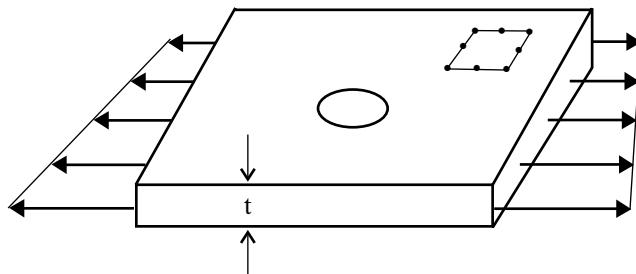
- ▶ Plates and shells (axisymmetric idealization)
- ▶ Long plates (plane strain idealization)
- ▶ Beams (plane stress idealization)
- The 8- and 9-node elements are usually most effective if the element is rectangular (non-distorted).
- The 4-node rectangular and 3-node triangular elements should only be used in analyses when bending effects are not significant. If the 4-node rectangular elements must be used when bending effects are significant, consider the use of the incompatible modes element (see above).
- When the structure to be modeled is axisymmetric and has a dimension which is very small compared with the others, e.g., thin plates and shells, the use of axisymmetric shell elements is more effective (see Section 2.5).



a) Thick cylinder (axisymmetric idealization)

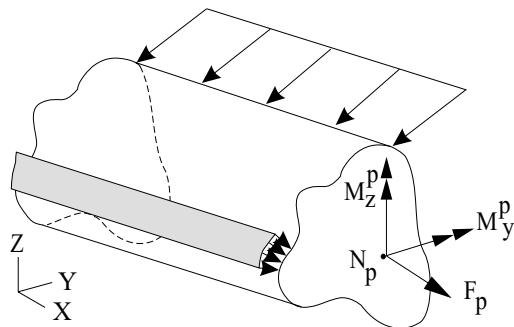


b) Long dam (plane strain idealization)



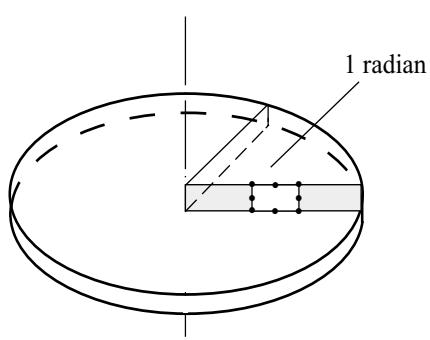
c) Sheet in membrane action (plane stress idealization)

Figure 2.2-7: Use of 2-D solid element for thick structures and solids

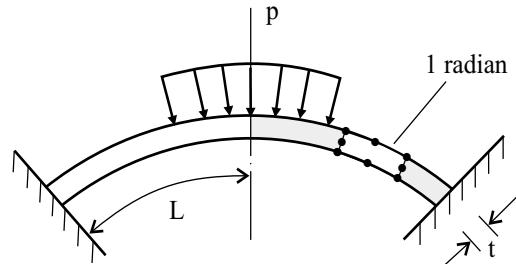


d) Turbine blade (generalized plane strain idealization)

Figure 2.2-7: (continued)

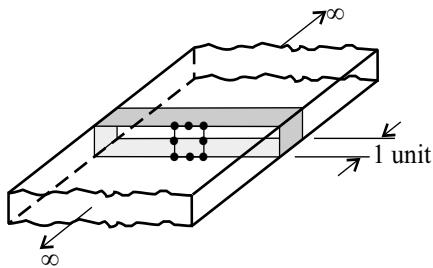


Circular plate

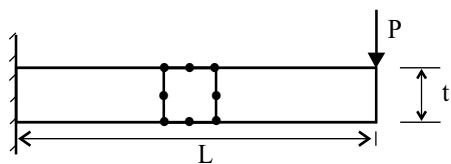


Spherical shell

a) Axisymmetric idealizations



b) Plane strain idealization (long plate)



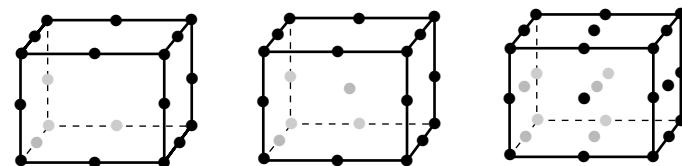
c) Plane stress idealization (cantilever)

Figure 2.2-8: Use of 2-D solid element for thin structures

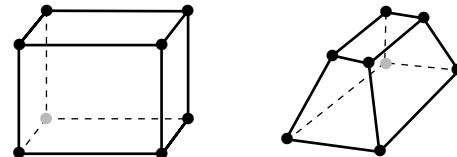
2.3 Three-dimensional solid elements

2.3.1 General considerations

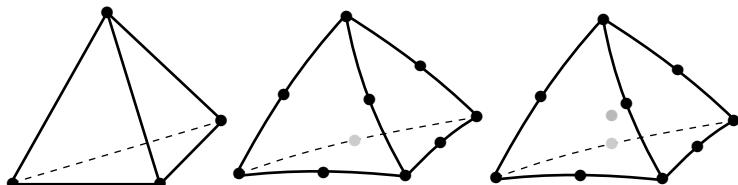
- The three-dimensional (3-D) solid element is a variable 4- to 20-node or a 21- or 27-node isoparametric element applicable to general 3-D analysis. Some typical 3-D solid elements are shown in Fig. 2.3-1.



(a) 20-, 21- & 27-node elements



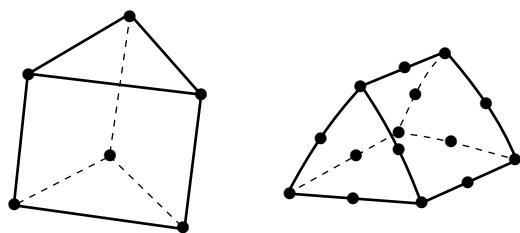
(b) 8-node elements



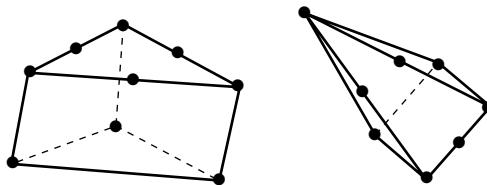
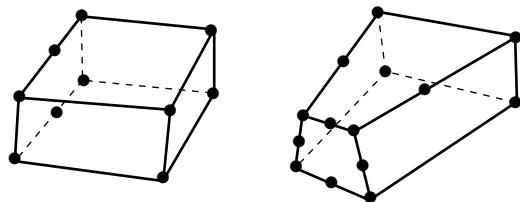
(c) 4-, 10- and 11-node tetrahedral elements

Figure 2.3-1: Some 3-D solid elements

- The 3-D solid element should be employed in analyses in which the three-dimensional state of stress (or strain) is required or in which special stress/strain conditions, such as those given in Section 2.2, do not exist (see Figs. 2.3-2 and 2.3-3).

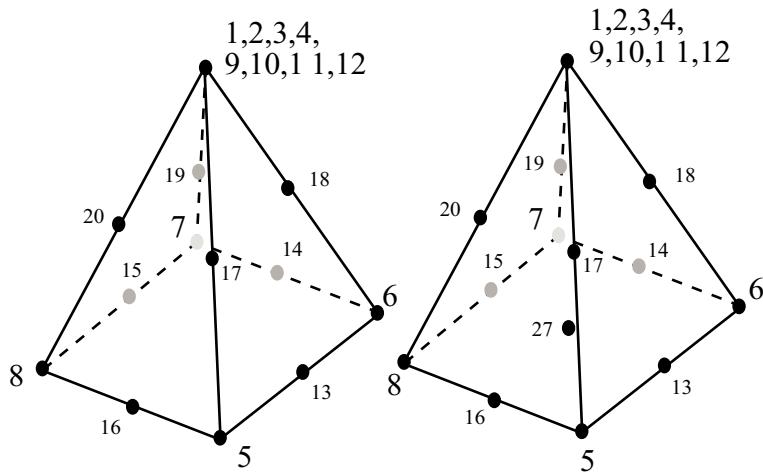


(d) 6- and 15- node prisms



(e) Elements for transition zones

Figure 2.3-1: (continued)



(f) 13-node and 14-node pyramid elements

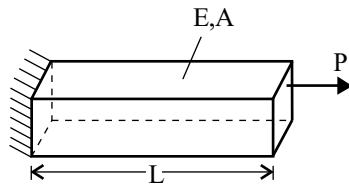
Figure 2.3-1 (continued)

*ref. KJB
Section 5.3* • The elements usually used are isoparametric displacement-based finite elements, and the formulation of the elements used in ADINA is described in ref. KJB, Ch. 5.

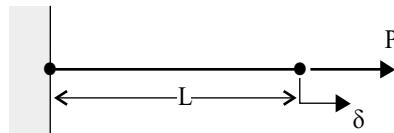
- The basic finite element assumptions are (see Fig. 2.3-4):

for the coordinates

$$x = \sum_{i=1}^q h_i x_i \quad y = \sum_{i=1}^q h_i y_i \quad z = \sum_{i=1}^q h_i z_i$$

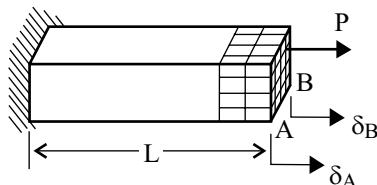


(a) Physical problem considered



(b) 1-D model

Plane sections remain plane. δ is constant for the entire section.



(c) 3-D model

Plane sections do not remain plane.

$$\delta_A \neq \delta_B$$

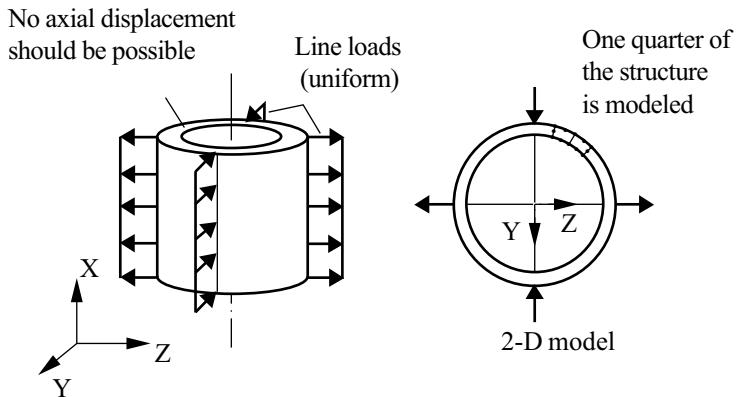
Figure 2.3-2: Illustration of use of 3-D elements

and for the displacements

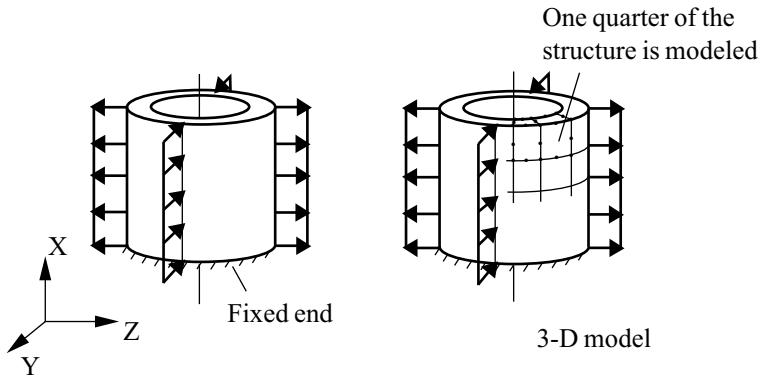
$$u = \sum_{i=1}^q h_i u_i \quad v = \sum_{i=1}^q h_i v_i \quad w = \sum_{i=1}^q h_i w_i$$

where

- $h_i(r, s, t)$ = interpolation function corresponding to node i
- r, s, t = isoparametric coordinates
- q = number of element nodes, $8 \leq q \leq 27$
- x_i, y_i, z_i = nodal point coordinates
- u_i, v_i, w_i = nodal point displacements



a) Plane strain 2-D analysis applicable



b) Three-dimensional model is used

Figure 2.3-3: Examples of 3-D solid modeling versus 2-D solid modeling

- In addition to the displacement-based elements, special mixed-interpolated elements are also available, in which the displacements and pressure are interpolated. These elements are effective and should be preferred in the analysis of incompressible media and inelastic materials (specifically for materials in which Poisson's ratio is close to 0.5, for rubber-like materials and for elasto-plastic materials). The use of the 8-node (one pressure variable) element or 27-node (4 pressure variables) element is recommended in this case.

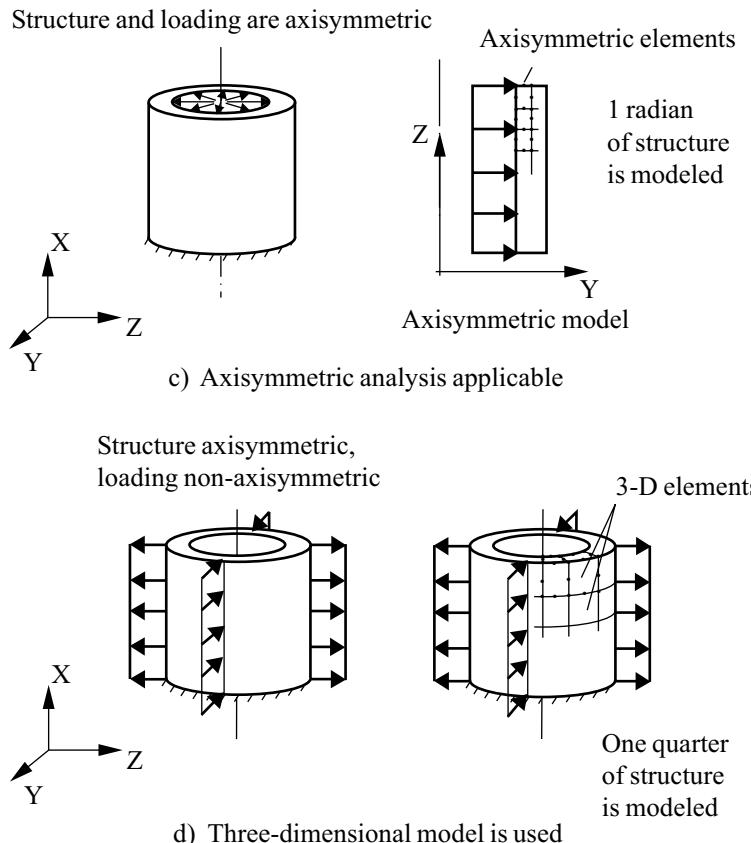


Figure 2.3-3: (continued)

- In addition to the displacement-based and mixed-interpolated elements, ADINA also includes the possibility of including incompatible modes in the formulation of the 8-node element. The addition of the incompatible modes increases the flexibility of the element, especially in bending situations. This element is analogous to the 4-node incompatible modes 2-D solid element discussed in Section 2.2.1, see the comments in that section for theoretical considerations.

The incompatible modes feature cannot be used in conjunction with the mixed-interpolation formulation.

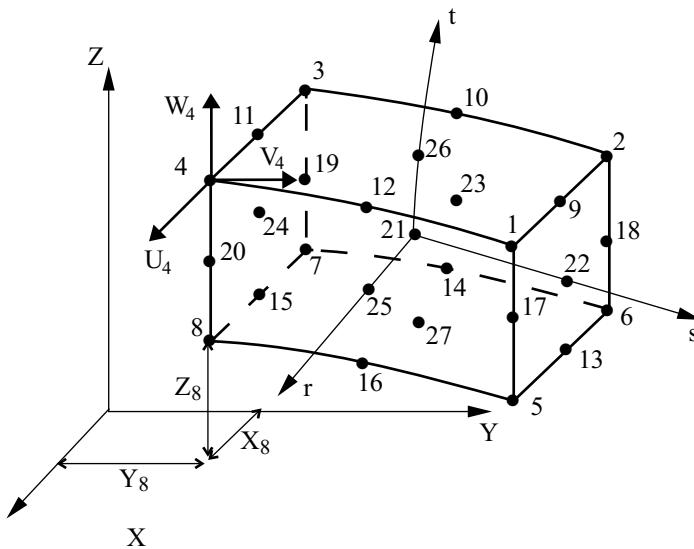


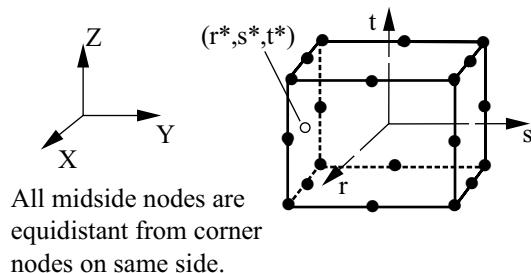
Figure 2.3-4: Conventions used for the nodal coordinates and displacements of the 3-D solid element

The incompatible modes feature is only available for the 8-node element; in particular note that the incompatible modes feature is not available for the degenerated 8-node elements or the 4-node tetrahedral element.

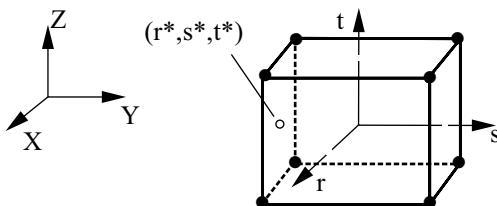
- It may sometimes be effective to interpolate the coordinates using a set of lower order interpolation functions, resulting in a sub-parametric element. For example, the 20-node and 8-node rectangular parallelepipeds in Fig. 2.3-5 have identical geometrical interpolation, i.e.,

$$\mathbf{x}_p(r^*, s^*, t^*) = \sum_{i=1}^{20} h_i(r^*, s^*, t^*) \mathbf{x}^i = \sum_{i=1}^8 h'_i(r^*, s^*, t^*) \mathbf{x}^i$$

- The elements can be used with 4 to 20 or with 21 or 27 nodes (tetrahedra, pyramids or prisms are derived from the degeneration of the 4 to 20-node rectangular elements, see Example 5.16 on pp. 366-367, ref. KJB). The interpolation functions for $q \leq 20$ are shown in Fig. 5.5, ref. KJB, p. 345.



a) 20-node element



b) 8-node element

Figure 2.3-5: Same geometry interpolation using 8 and 20 nodes

- Degenerated elements such as prisms, pyramids or tetrahedra are formed in ADINA by assigning the same global node to the local element nodes located along the same side or on the same face.

The triangular prism as a degenerated 20-node element (see Fig. 2.3-1(d)) can be used in ADINA in three different ways:

- ▶ As a 15-node triangular prism with the strain singularity $\frac{1}{\sqrt{r}}$. This is an extension of the techniques described on pp. 368-376 of ref. KJB (see also Chapter 9);
- ▶ As a 20-node triangular prism with the strain singularity $\frac{1}{r}$ (see also Chapter 9);

- ▶ As a spatially isotropic triangular prism where corrections are applied to the interpolation functions of the collapsed 20-node element.

The 10-node tetrahedron (see Fig. 2.3-1(c)) is obtained by collapsing nodes and sides of rectangular elements. A spatially isotropic 10-node tetrahedron is available in ADINA. Similarly, an 11-node spatially isotropic tetrahedron is available in ADINA.

The 13-node pyramid and the 14-node pyramid (see Fig. 2.3-1(f)) are degenerate solid elements from 20-node and 21-node bricks, respectively. They can be used as spatially isotropic elements.

The 4-node tetrahedron (see Fig. 2.3-1(c)) is obtained by collapsing nodes and sides of the 8-node rectangular element. This element exhibits constant strain conditions.

The elements in Fig. 2.3-1(e) are not spatially isotropic and should usually only be employed in transition regions.

- Linear and nonlinear fracture mechanics analysis of stationary cracks can be performed with three-dimensional elements (see Chapter 9).

2.3.2 Material models and nonlinear formulations

- The 3-D elements can be used with the following material models: **elastic-isotropic, elastic-orthotropic, plastic-bilinear, plastic-multilinear, Drucker-Prager, Mroz-bilinear, plastic-orthotropic, thermo-isotropic, thermo-orthotropic, thermo-plastic, creep, plastic-creep, multilinear-plastic-creep, concrete, curve-description, Ogden, Mooney-Rivlin, Arruda-Boyce, hyper-foam, creep-variable, plastic-creep-variable, multilinear-plastic-creep-variable, Gurson-plastic, Cam-clay, Mohr-Coulomb, viscoelastic, irradiation creep, gasket, shape-memory alloy, user-supplied.**
- You should use the mixed interpolation formulation whenever using the plastic-bilinear, plastic-multilinear, Mroz-bilinear, plastic-orthotropic, thermo-plastic, creep, plastic-creep, multilinear-plastic-creep, Ogden, Mooney-Rivlin, Arruda-Boyce, creep-variable, plastic-creep-variable, multilinear-plastic-creep-variable or viscoelastic material models; or when using the elastic-isotropic

material and the Poisson's ratio is close to 0.5.

The AUI automatically chooses the mixed interpolation formulation whenever the Ogden, Mooney-Rivlin or Arruda-Boyce material models are used. For the other material types listed in the previous paragraph, you should explicitly select the mixed interpolation formulation.

- The 3-D elements can be used with a **small displacement/small strain, large displacement/small strain** or a **large displacement/large strain** formulation.

The small displacement/small strain and large displacement/small strain formulations can be used with any material model, except for the Ogden, Mooney-Rivlin, Arruda-Boyce and hyper-foam material models. The use of a linear material with the small displacement/small strain formulation corresponds to a linear formulation, and the use of a nonlinear material with the small displacement/small strain formulation corresponds to a materially-nonlinear-only formulation. The program uses the TL (total Lagrangian) formulation when you choose a large displacement/small strain formulation.

The large displacement/large strain formulations can be used with the plastic-bilinear, plastic-multilinear, Mroz-bilinear, thermo-plastic, creep, plastic-creep, multilinear-plastic-creep, creep-variable, plastic-creep-variable, multilinear-plastic-creep-variable, viscoelastic and user-supplied material models. The program uses a ULH formulation when you choose a large displacement/large strain formulation with these material models.

A large displacement/large strain formulation is used with the Ogden, Mooney-Rivlin, Arruda-Boyce and hyper-foam material models. The program uses a TL (total Lagrangian) formulation in this case.

*ref. KJB
Sections 6.2
and 6.3.5*

- The basic continuum mechanics formulations are described in ref. KJB, pp. 497-568. The finite element discretization is summarized in Table 6.6, p. 555, ref. KJB.

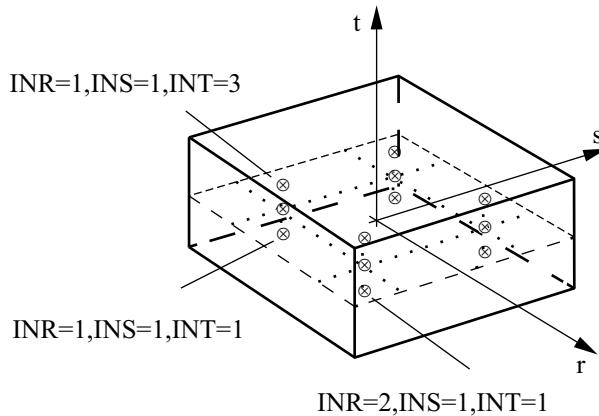
*ref. KJB
Section 6.8.1*

- Note that all these formulations can be used in one finite element mesh. If the elements are initially compatible, then they will remain compatible throughout the analysis.

2.3.3 Numerical integration

*ref. KJB
Sections 5.5.3,
5.5.4 and 5.5.5*

- For the calculation of element matrices, Gauss numerical integration is used. The same integration order (NINT) is always assigned to the r- and s-directions, and can be from 2 to 6. The integration order in the t-direction (NINTT) can, however, be assigned independently, and can also be from 2 to 6. The default Gauss integration orders are $2 \times 2 \times 2$ for the 8-node (cube or prism) elements and $3 \times 3 \times 3$ otherwise, except for tetrahedra.
- Except for the 4-node, 10-node and 11-node tetrahedra, the convention for the integration point numbering used in the stress output is as follows: The first integration point is the point with the most negative location in r, s and t. The next integration points are located by increasing t (and label INT) successively up to its maximum positive value, then increasing s (and label INS) one position towards positive and varying t again from its maximum negative to its maximum positive values, and so on (Fig. 2.3-6).

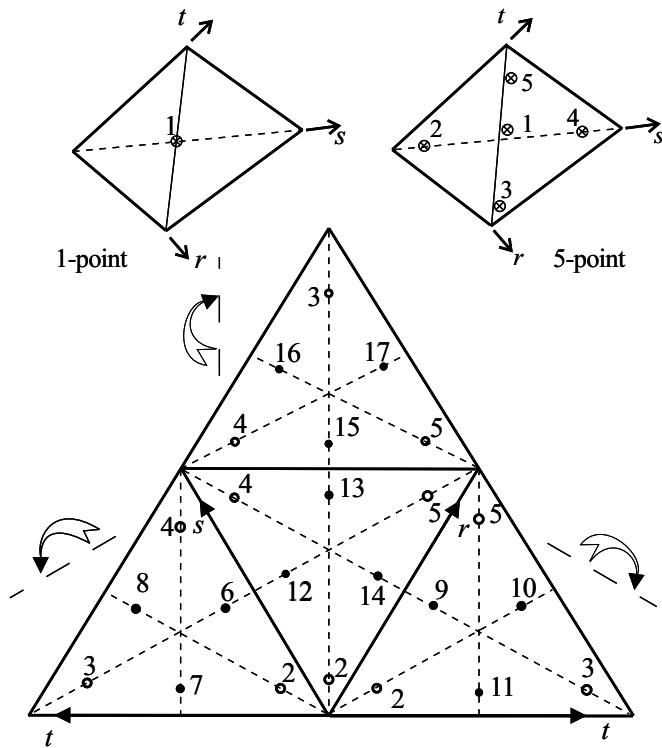


- a) All elements except tetrahedra

Figure 2.3-6: Example of integration point labeling for 3-D solid elements

- Tetrahedral elements are spatially isotropic with respect to integration point locations and interpolation functions (see Figure 2.3-6). For the 4-node tetrahedral element, 1-point Gauss integration is used. For the 10-node and 11-node tetrahedral elements, 17-point Gauss integration is used.
- The integration order used for tetrahedral elements depends upon the integration order used for hexahedral (brick) elements as follows. Given NINT, NINTT, the program computes the total number of integration points INTRST=NINT*NINT*NINTT. If INTRST is equal to 1, then tetrahedral elements use 1 point Gauss integration; otherwise, if INTRST is less than or equal to 8, then tetrahedral elements use 5 point Gauss integration; otherwise tetrahedral elements use 17 point Gauss integration.

However, for 4-node tetrahedral elements, 1-point Gauss integration is always used.



17-point (unfolded) (Point 1 at centroid not shown)

b) Tetrahedral elements

Figure 2.3-6: (continued)

- Note that in geometrically nonlinear analysis, the spatial positions of the Gauss integration points change continuously as the element undergoes deformations, but throughout the response the same material particles are at the integration points.
- The order of numerical integration in large displacement elastic analysis is usually best chosen to be equal to the order appropriate in linear elastic analysis. Hence the default values are usually appropriate. However, in certain inelastic analyses, a higher order should be used (see Fig. 6.25, p. 638, ref. KJB).

ref. KJB
Section 6.8.4

2.3.4 Mass matrices

- The consistent mass matrix is always calculated using $3 \times 3 \times 3$ Gauss integration except for the tetrahedral 4-node, 10-node and 11-node elements which use a 17-point Gauss integration.
- The lumped mass matrix of an element is formed by dividing the element's mass M equally among each of its n nodal points. Hence the mass assigned to each node is M / n . No special distributory concepts are employed to distinguish between corner and midside nodes, or to account for element distortion.
- Note that n is the number of distinct non-repeated nodes in the element. Hence, when an element side or face is collapsed to a single node, the total mass of the element is divided among the unique nodes in the element.

2.3.5 Element output

- You can request that ADINA either print or save stresses or forces.

Stresses: Each element outputs, at its integration points, the following information to the porthole file, based on the material model. This information is accessible in the AUI using the given variable names.

Notice that for the orthotropic material models, you can request that the stresses and strains be saved in the material coordinate system.

Elastic-isotropic, elastic-orthotropic (results saved in global system): STRESS (XYZ) , STRAIN (XYZ)

Elastic-isotropic with thermal effects: STRESS (XYZ) , STRAIN (XYZ) , THERMAL _ STRAIN , ELEMENT _ TEMPERATURE

Orthotropic (results saved in material system): STRESS (ABC) , STRAIN (ABC)

Thermo-isotropic, thermo-orthotropic (results saved in global system): STRESS(XYZ), STRAIN(XYZ), THERMAL_STRAIN(XYZ), ELEMENT_TEMPERATURE

Curve description: CRACK_FLAG, STRESS(XYZ), STRAIN(XYZ), FE_SIGMA-P1, FE_SIGMA-P2, FE_SIGMA-P3, GRAVITY_IN-SITU_PRESSURE, VOLUMETRIC_STRAIN, FE_SIGMA-P1_DIRECTION-X, FE_SIGMA-P1_DIRECTION-Y, FE_SIGMA-P1_DIRECTION-Z, FE_SIGMA-P2_DIRECTION-X, FE_SIGMA-P2_DIRECTION-Y, FE_SIGMA-P2_DIRECTION-Z, FE_SIGMA-P3_DIRECTION-X, FE_SIGMA-P3_DIRECTION-Y, FE_SIGMA-P3_DIRECTION-Z

Concrete: CRACK_FLAG, STRESS(XYZ), STRAIN(XYZ), FE_SIGMA-P1, FE_SIGMA-P2, FE_SIGMA-P3, GRAVITY_IN-SITU_PRESSURE, ELEMENT_TEMPERATURE, FE_SIGMA-P1_DIRECTION-X, FE_SIGMA-P1_DIRECTION-Y, FE_SIGMA-P1_DIRECTION-Z, FE_SIGMA-P2_DIRECTION-X, FE_SIGMA-P2_DIRECTION-Y, FE_SIGMA-P2_DIRECTION-Z, FE_SIGMA-P3_DIRECTION-X, FE_SIGMA-P3_DIRECTION-Y, FE_SIGMA-P3_DIRECTION-Z, THERMAL_STRAIN(XYZ)

Small strains: Plastic-bilinear, plastic-multilinear, Mroz-bilinear:
PLASTIC_FLAG, STRESS(XYZ), STRAIN(XYZ), PLASTIC_STRAIN(XYZ), FE_EFFECTIVE_STRESS, YIELD_STRESS, ACCUMULATED_EFFECTIVE_PLASTIC_STRAIN, THERMAL_STRAIN(XYZ), ELEMENT_TEMPERATURE

Large strains: Plastic-bilinear, plastic-multilinear, Mroz-bilinear:
PLASTIC_FLAG, STRESS(XYZ), DEFORMATION_GRADIENT(XYZ), FE_EFFECTIVE_STRESS, YIELD_STRESS, ACCUMULATED_EFFECTIVE_PLASTIC_STRAIN, THERMAL_STRAIN(XYZ), ELEMENT_TEMPERATURE

Plastic-orthotropic (results saved in global system):

PLASTIC_FLAG, STRESS(XYZ), STRAIN(XYZ),
PLASTIC_STRAIN(XYZ), HILL_EFFECTIVE_STRESS,
YIELD_STRESS,
ACCUMULATED_EFFECTIVE_PLASTIC_STRAIN,
THERMAL_STRAIN(XYZ), ELEMENT_TEMPERATURE

Plastic-orthotropic (results saved in material system):

PLASTIC_FLAG, STRESS(ABC), STRAIN(ABC),
PLASTIC_STRAIN(ABC), HILL_EFFECTIVE_STRESS,
YIELD_STRESS,
ACCUMULATED_EFFECTIVE_PLASTIC_STRAIN,
THERMAL_STRAIN(ABC), ELEMENT_TEMPERATURE

Small strains: Drucker-Prager: PLASTIC_FLAG,
PLASTIC_FLAG-2, STRESS(XYZ), STRAIN(XYZ),
PLASTIC_STRAIN(XYZ), CAP_LOCATION,
YIELD_FUNCTION

Large strains: Drucker-Prager: PLASTIC_FLAG,
PLASTIC_FLAG-2, STRESS(XYZ),
DEFORMATION_GRADIENT(XYZ), CAP_LOCATION,
YIELD_FUNCTION

Small strains: Thermo-plastic, creep, plastic-creep, multilinear-plastic-creep, creep-variable, plastic-creep-variable, multilinear-plastic-creep-variable: PLASTIC_FLAG,
NUMBER_OF_SUBINCREMENTS, STRESS(XYZ),
STRAIN(XYZ), PLASTIC_STRAIN(XYZ),
CREEP_STRAIN(XYZ), THERMAL_STRAIN(XYZ),
ELEMENT_TEMPERATURE,
ACCUMULATED_EFFECTIVE_PLASTIC_STRAIN,
FE_EFFECTIVE_STRESS, YIELD_STRESS,
EFFECTIVE_CREEP_STRAIN

Large strains: Thermo-plastic, creep, plastic-creep, multilinear-plastic-creep, creep-variable, plastic-creep-variable, multilinear-plastic-creep-variable: PLASTIC_FLAG,
NUMBER_OF_SUBINCREMENTS, STRESS(XYZ),
DEFORMATION_GRADIENT(XYZ),
THERMAL_STRAIN(XYZ), ELEMENT_TEMPERATURE,
ACCUMULATED_EFFECTIVE_PLASTIC_STRAIN,

FE_EFFECTIVE_STRESS, YIELD_STRESS,
EFFECTIVE_CREEP_STRAIN

Mooney-Rivlin , Ogden, Arruda-Boyce, hyper-foam (strains saved): STRESS(XYZ), STRAIN(XYZ)

Mooney-Rivlin , Ogden, Arruda-Boyce, hyper-foam (deformation gradients saved): STRESS(XYZ),
DEFORMATION_GRADIENT(XYZ)

Small strains: User-supplied: STRESS(XYZ), STRAIN(XYZ),
USER_VARIABLE_I

Large strains: User-supplied: STRESS(XYZ),
DEFORMATION_GRADIENT(XYZ), USER_VARIABLE_I

Thermo-orthotropic (results saved in material system):

STRESS(ABC), STRAIN(ABC),
THERMAL_STRAIN(ABC), FE_EFFECTIVE_STRESS,
ELEMENT_TEMPERATURE

Small strains: Gurson plastic: PLASTIC_FLAG,
STRESS(XYZ), STRAIN(XYZ),
PLASTIC_STRAIN(XYZ), FE_EFFECTIVE_STRESS,
YIELD_STRESS,
ACCUMULATED_EFFECTIVE_PLASTIC_STRAIN,
THERMAL_STRAIN(XYZ), ELEMENT_TEMPERATURE,
VOID_VOLUME_FRACTION

Large strains: Gurson plastic: PLASTIC_FLAG,
STRESS(XYZ), DEFORMATION_GRADIENT(XYZ),
FE_EFFECTIVE_STRESS, YIELD_STRESS,
ACCUMULATED_EFFECTIVE_PLASTIC_STRAIN,
THERMAL_STRAIN(XYZ), ELEMENT_TEMPERATURE,
VOID_VOLUME_FRACTION

Small strains: Cam-clay: PLASTIC_FLAG, STRESS(XYZ),
STRAIN(XYZ), PLASTIC_STRAIN(XYZ),
YIELD_SURFACE_DIAMETER_P, YIELD_FUNCTION,
MEAN_STRESS, DISTORTIONAL_STRESS,
VOLUMETRIC_STRAIN, VOID_RATIO,
EFFECTIVE_STRESS_RATIO, SPECIFIC_VOLUME

Large strains: Cam-clay: PLASTIC_FLAG, STRESS(XYZ),
DEFORMATION_GRADIENT(XYZ),
YIELD_SURFACE_DIAMETER_P, YIELD_FUNCTION,
MEAN_STRESS, DISTORTIONAL_STRESS,
VOLUMETRIC_STRAIN, VOID_RATIO,
EFFECTIVE_STRESS_RATIO, SPECIFIC_VOLUME

Mohr-Coulomb: PLASTIC_FLAG, STRESS(XYZ),
STRAIN(XYZ), PLASTIC_STRAIN(XYZ),
YIELD_FUNCTION

Small strains: viscoelastic: STRESS(XYZ), STRAIN(XYZ),
THERMAL_STRAIN(XYZ), ELEMENT_TEMPERATURE

Large strains: viscoelastic: STRESS(XYZ),
DEFORMATION_GRADIENT(XYZ),
THERMAL_STRAIN(XYZ), ELEMENT_TEMPERATURE

Small strains: creep-irradiation: STRESS(XYZ),
STRAIN(XYZ), PLASTIC_STRAIN(XYZ),
CREEP_STRAIN(XYZ), THERMAL_STRAIN(XYZ),
ELEMENT_TEMPERATURE,
ACCUMULATED_EFFECTIVE_PLASTIC_STRAIN,
EFFECTIVE_STRESS, YIELD_STRESS,
EFFECTIVE_CREEP_STRAIN,
IRRADIATION_STRAIN(XYZ).

Large strains: creep-irradiation: STRESS(XYZ),
DEFORMATION_GRADIENT(XYZ),
THERMAL_STRAIN(XYZ), ELEMENT_TEMPERATURE,
ACCUMULATED_EFFECTIVE_PLASTIC_STRAIN,
EFFECTIVE_STRESS, YIELD_STRESS,
EFFECTIVE_CREEP_STRAIN,
IRRADIATION_STRAIN(XYZ).

In the above lists,

STRESS(XYZ) = STRESS-XX, STRESS-YY, STRESS-ZZ, STRESS-XY, STRESS-XZ, STRESS-YZ

STRESS(ABC) = STRESS-AA, STRESS-BB, STRESS-CC, STRESS-AB, STRESS-AC, STRESS-BC

with similar definitions for the other abbreviations used above. But the variable DEFORMATION_GRADIENT(XYZ) is interpreted as follows:

```
DEFORMATION_GRADIENT(XYZ) =  
DEFORMATION_GRADIENT-XX,  
DEFORMATION_GRADIENT-XY,  
DEFORMATION_GRADIENT-XZ,  
DEFORMATION_GRADIENT-YX,  
DEFORMATION_GRADIENT-YY,  
DEFORMATION_GRADIENT-YZ,  
DEFORMATION_GRADIENT-ZX,  
DEFORMATION_GRADIENT-ZY,  
DEFORMATION_GRADIENT-ZZ,
```

Also note that you can request stretches instead of deformation gradients, and in this case STRETCH(XYZ) replaces DEFORMATION_GRADIENT(XYZ) in the above lists.

See Section 12.1.1 for the definitions of those variables that are not self-explanatory.

ref. KJB
Section 4.2.1

- In ADINA, the stresses are calculated using the strains at the point of interest. Hence, they are not spatially extrapolated or smoothed. The AUI can be employed to calculate smoothed stresses from the results output by ADINA.
- You can also request that strain energy densities be output along with the stresses.

Nodal forces: The nodal forces which correspond to the element stresses can also be requested in ADINA. This force vector is calculated in a linear analysis using

$$\mathbf{F} = \int_V \mathbf{B}^T \boldsymbol{\tau} dV$$

where \mathbf{B} is the element strain-displacement matrix and $\boldsymbol{\tau}$ is the stress vector. The integration is performed over the volume of the element; see Example 5.11 in ref. KJB, pp. 358-359.

The nodal forces are accessible in the AUI using the variable names NODAL_FORCE-X, NODAL_FORCE-Y, NODAL_FORCE-Z.

- The same relation is used for the element force calculation in a nonlinear analysis, except that updated quantities are used in the integration.

2.3.6 Recommendations on use of elements

- The 27-node element is the most accurate among all available elements. However, the use of this element can be costly.
- The 20-node element is usually the most effective (except in wave propagation analysis using the central difference method and a lumped mass idealization, see Chapter 7).
- The 20-node element can be employed for analysis of thick structures and solids; some examples are given in Fig. 2.3-7.
- The 20-node element is usually most effective if the element is rectangular (undistorted).
- It is not recommended to use the 20-node element for contact analysis because, in this case, the corresponding segment tractions will be very approximate.
- The 11-node tetrahedral element should be used when tetrahedral meshing is used and a mixed interpolated formulation is required (as in incompressible analysis).

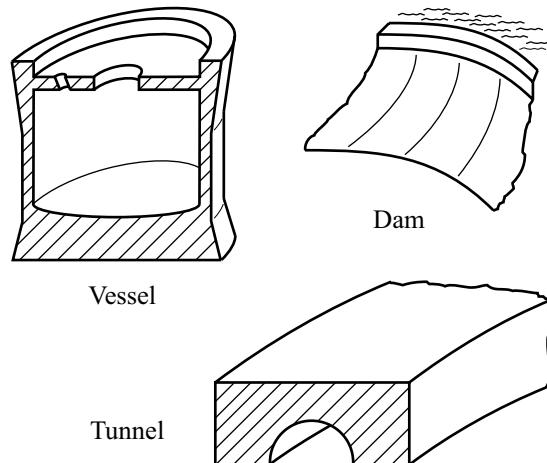


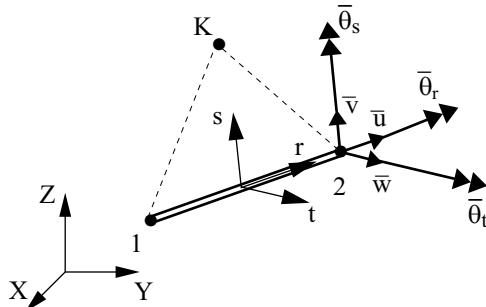
Figure 2.3-7: Some structures for which 3-D solid elements can be used

- The 8-node brick element and 4-node tetrahedral element should only be used in analyses when bending effects are not significant. If the 8-node brick element must be used when bending effects are significant, consider the use of the incompatible modes element.
- When the structure to be modeled has a dimension which is extremely small compared with the others, e.g., thin plates and shells, the use of the 3-D solid element usually results in too stiff a model and a poor conditioning of the stiffness matrix. In this case, the use of the shell or the plate/shell elements (see Sections 2.7 and 2.6) is more effective.

2.4 Beam elements

- The beam element is a 2-node Hermitian beam with a constant cross-section and 6 degrees of freedom at each node. The displacements modeled by the beam element are (see Fig. 2.4-1):
 - ▶ Cubic transverse displacements \bar{v} and \bar{w} (s- and t-direction displacements)

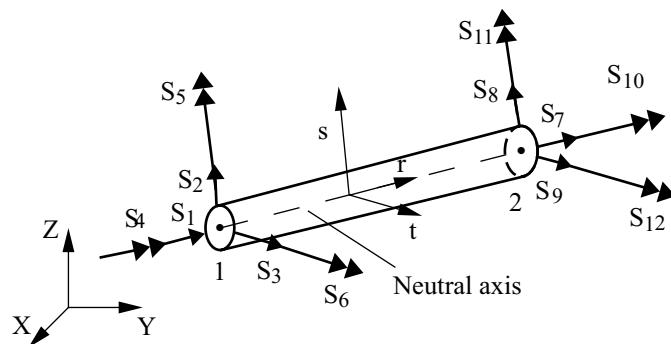
- ▶ Linear longitudinal displacements \bar{u} (r-direction displacement)
- ▶ Linear torsional displacements $\bar{\theta}_r$ and warping displacements
- The element is formulated based on the Bernoulli-Euler beam theory, corrected for shear deformation effects if requested.
- The behavior of the beam can be described using either a cross-section shape and a material model, or a moment-curvature input.
- The principal planes of inertia of the element coincide with the element local coordinate system (r,s,t) . The (r,s,t) coordinate system is defined using the element end nodes (local nodes 1 and 2) and the auxiliary node K (see Figure 2.4-1). You must position node K appropriately with respect to the actual orientation of the principal planes of inertia of the element, especially in the case of composite cross-sections.



The element s-direction is defined as follows.
The s-direction lies in the plane defined by nodes 1, 2, K, and the s-direction is perpendicular to the r-direction.

(a) Geometry definition

Figure 2.4-1: Conventions used for 2-node
Hermitian beam element



- S_1 = r-direction force at node 1 (axial force, positive in compression)
- S_2 = s-direction force at node 1 (shear force)
- S_3 = t-direction force at node 1 (shear force)
- S_4 = r-direction moment at node 1 (torsion)
- S_5 = s-direction moment at node 1 (bending moment)
- S_6 = t-direction moment at node 1 (bending moment)

- S_7 = r-direction force at node 2 (axial force, positive in tension)
- S_8 = s-direction force at node 2 (shear force)
- S_9 = t-direction force at node 2 (shear force)
- S_{10} = r-direction moment at node 2 (torsion)
- S_{11} = s-direction moment at node 2 (bending moment)
- S_{12} = t-direction moment at node 2 (bending moment)

(b) Element end forces/moment

Figure 2.4-1: Conventions used for 2-node
Hermitian beam element

- You can create off-centered beam elements using the rigid link (see Fig. 2.4-2 and Section 5.11-2).
- To model beam internal hinges, a moment and force release option can be used (see Fig. 2.4-3). Twisting moments and axial forces can also be released. End releases are applied to the element local nodes (not to the global nodes). Therefore, to model the hinge shown in Fig. 2.4-3(a), a moment end release can be applied to local node 2 of element 1, or to local node 1 of element 2 (but not to both local nodes).

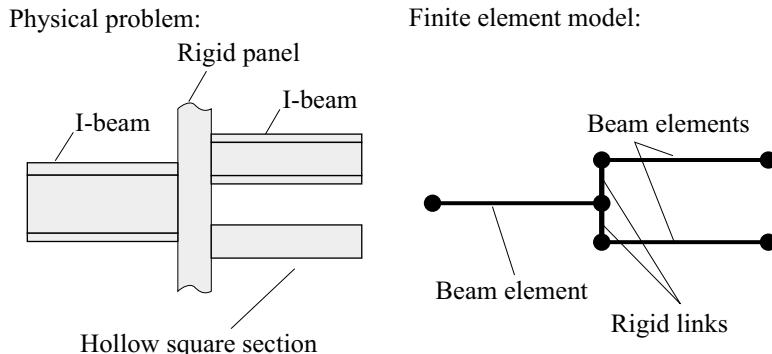


Figure 2.4-2: Use of rigid link for modeling of off-centered beams

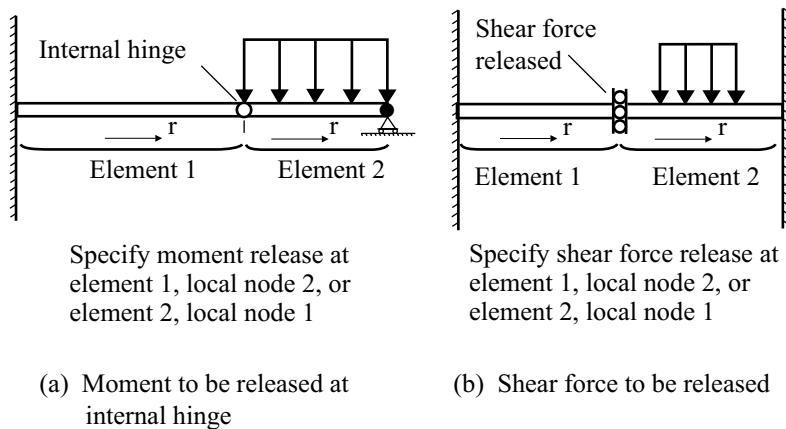


Figure 2.4-3: Use of moment and shear force release options

Rigid-end capability: For the modeling of assemblages of braces and struts, it is sometimes of interest that the joints of the structure be considered as infinitely rigid (see Figure 2.4-4). Hence, a rigid-end capability exists for the beam element: with this capability, a fraction at each end of the element can be considered as rigid.

You can select either perfectly rigid ends (with infinite rigidity) or very stiff ends (with rigidity calculated by multiplying the rigidity of the element by a large number, such as 10^6).

The lengths of the beam rigid ends are specified in the units of the model.

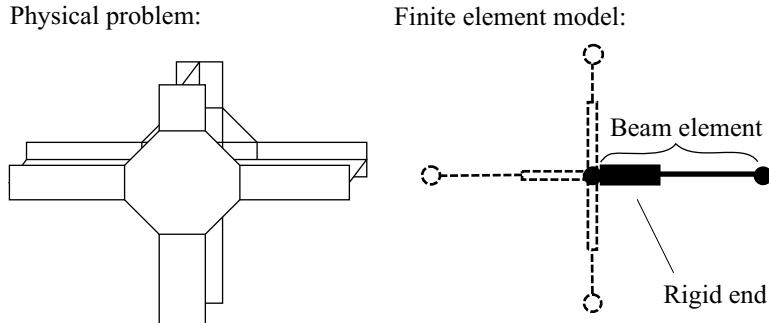


Figure 2.4-4: Modeling of a beam intersection using elements with rigid ends

- The assumptions of the rigid-end option are as follows:
 - ▶ The element, including its rigid ends, is formulated as one single element. Hence, as for all beam elements, the internal deformations and element internal forces are all referred to a line connecting the end-nodes, and using the assumptions of small internal relative deformations.
 - ▶ The rigid ends will never undergo plasticity. Hence, although the stress in the rigid ends may be very high (they are not printed by the program), plasticity is not reached. This may be quite unrealistic in a model using the rigid-end capability.
 - ▶ The mass of the rigid ends is taken into account in the mass matrix calculations.

2.4.1 Linear beam element

- It is assumed that the displacements, rotations, and strains are infinitesimally small, and the elastic-isotropic material is used.
- The beam element stiffness matrix is evaluated in closed form. The stiffness matrix used is discussed in detail in the following reference:

ref. J.S. Przemieniecki, *Theory of Matrix Structural Analysis*, McGraw-Hill Book Co., 1968.

Note that the element stiffness matrix is defined by the following quantities:

- E = Young's modulus
- ν = Poisson's ratio
- L = length of the beam
- I_r, I_s, I_t = moments of inertia about the local principal axes r , s , and t
- A = cross-sectional area
- A_s^{sh}, A_t^{sh} = effective shear areas in s and t directions

This stiffness matrix is transformed from the local degrees of freedom (in the r , s , t axes) to the global degrees of freedom (in the X , Y , Z axes) and is then assembled into the stiffness matrix of the complete structure.

- You can enter either the section areas and moments of inertia, or the dimensions for various sections (see Fig. 2.4-5 and Table 2.4-1).
- The x' and y' axes are the principal axes of the cross-section, and have their origin at the centroid of the cross-section.

Note that the x' and y' axes are oriented relative to the element local s and t coordinate axes according to Fig. 2.4-6, and that the element s and t coordinate axes directions are controlled by the location of the auxiliary node K (Fig. 2.4-1). The x' and y' axes can be offset from the element s and t coordinate axes using parameters SOFFSET, TOFFSET.

- When offsets are used they affect the moments of inertia I_s and I_t of the cross-section as follows

$$\begin{aligned}I_s &= I_{x'x'} + A \times \text{SOFFSET}^2 \\I_t &= I_{y'y'} + A \times \text{TOFFSET}^2\end{aligned}$$

Note that axial forces applied at Nodes 1 or 2 are assumed to be acting along the beam's centroid and hence cause no bending. Also shear forces applied at Nodes 1 or 2 are assumed to be acting through the beam's shear center and hence cause no twisting.

In order to model the bending due to an off-centroidal axial force or a shear force applied away from the shear center, you can either apply the resulting moments directly or apply the forces at an offset location using rigid links (section 5.15-2).

Beam offsets can be used to model multiple off-centered beam elements such as shown in Figure 2.4-2 without using rigid links (see Figure 2.4-7). In this case, there is only one node at the cross-section which should be located at the centroid of the combined cross-section.

Table 2.4-1: Formulas used for torsional moment of inertia in various cross sections (see Figure 2.4-5)

Cross-section	Torsional moment of inertia I_r
Rectangular	$\frac{1}{3} \left(1 - 0.63 \frac{H1}{H2} \left(1 - \frac{H1^4}{12H2^4} \right) \right) H1^3 H2, \quad H1 \leq H2$
Pipe	$\frac{\pi}{32} (H1^4 - H2^4)$
Box	$\frac{2 H3 H4 (H1 - H3)^2 (H2 - H4)^2}{H1 H3 + H2 H4 - H3^2 - H4^2}$
I	$\frac{1}{3} (H4^3 H1 + H6^3 H3 + H5^3 (H2 - H4 - H6))$
U	$\frac{1}{3} (2H4^3 H1 + H3^3 (H2 - 2H4))$
L	$\frac{1}{3} (H3^3 H2 + H4^3 (H1 - H3))$

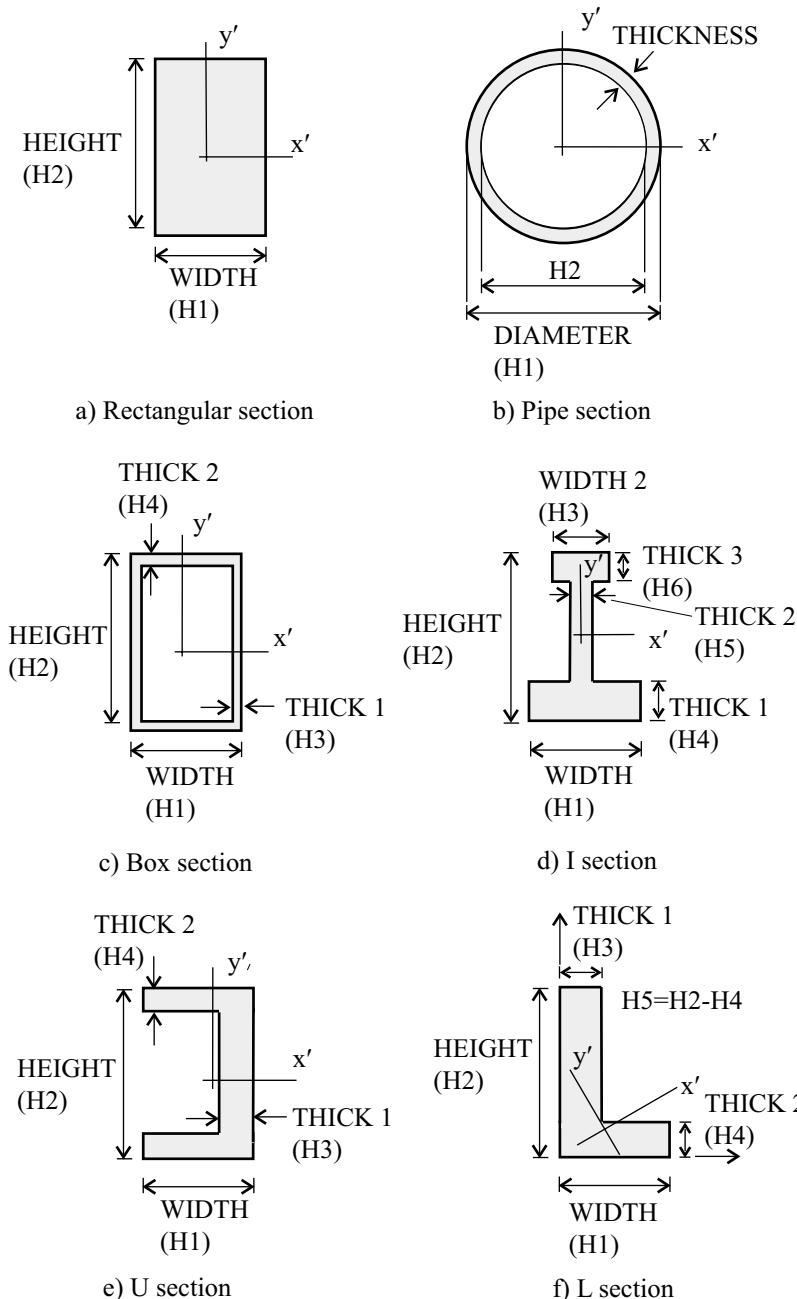


Figure 2.4-5: Possible cross-section shapes for the linear elastic 2-node Hermitian beam element

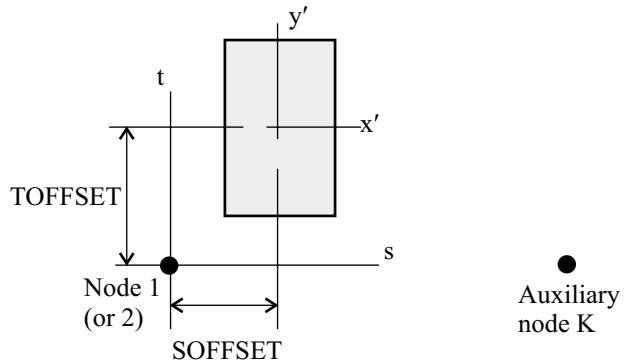


Figure 2.4-6: Relationship between principal cross-section axes and element local coordinate system

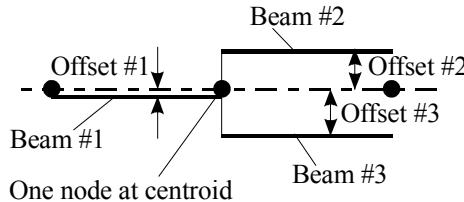


Figure 2.4-7: Using offsets to model off-centered beams

- Note that warping effects are only taken into account by a modification of the torsional rigidity, this modification to be specified in the input.
- In the case of the L cross-section, the maximum and minimum bending moments of inertia (about the x' and y' axes) are obtained as follows:

$$I_{x'x'} = \frac{I_{xx} + I_{yy}}{2} + \sqrt{\left(\frac{I_{xx} - I_{yy}}{2}\right)^2 + I_{xy}^2}$$

$$I_{y'y'} = \frac{I_{xx} + I_{yy}}{2} - \sqrt{\left(\frac{I_{xx} - I_{yy}}{2}\right)^2 + I_{xy}^2}$$

where the moments and products of inertia with respect to the axes x and y are:

$$I_{xx} = \frac{H_4 H_1^3}{12} + H_1 H_4 \left(\frac{H_1}{2} - C_x \right)^2 + \frac{H_5 H_3^3}{12} + H_5 H_3 \left(C_x - \frac{H_3}{2} \right)^2$$

$$I_{yy} = \frac{H_1 H_4^3}{12} + H_1 H_4 \left(C_y - \frac{H_4}{2} \right)^2 + \frac{H_3 H_5^3}{12} + H_5 H_3 \left(\frac{H_5}{2} + H_4 - C_y \right)^2$$

$$I_{xy} = H_5 H_3 \left(\frac{H_3}{2} - C_x \right) \left(\frac{H_5}{2} + H_4 - C_y \right)$$

$$+ H_1 H_4 \left(-C_y + \frac{H_4}{2} \right) \left(\frac{H_1}{2} - C_x \right)$$

and where (C_x, C_y) are the centroid coordinates.

In this case, the angles between the principal axes x' and y' and the horizontal (according to Figure 2.4-5) are

$$\theta_{x'}, \theta_{y'} = \frac{1}{2} \arctan \left(\frac{I_{y'y'} - I_{x'x'}}{2I_{x'y'}} \right)$$

These two angles are always 90° apart. For this L-shaped geometry, the angle defining the major principal axis $\theta_{x'}$ will always be the angle between 0° and 90° .

- You can specify the coefficient of thermal expansion as a material property. The coefficient of thermal expansion is constant (independent of the temperature).
- For the linear beam element are calculated either
 - element end forces/moments or
 - section forces/moments at discrete locations along the axial direction of the element.

The element end forces/moments are accessible in the AUI using the variable names NODAL_FORCE-R, NODAL_FORCE-S,

NODAL_FORCE-T, NODAL_MOMENT-R, NODAL_MOMENT-S and NODAL_MOMENT-T. The end forces/moment are computed at the element local nodes. In the AUI, element local nodes are defined as element points of type label. For example, to access the result computed at element 5, local node 2, define an element point of type label with element number 5, label number 2.

The section forces/moment along the element are accessible in the AUI using the variable names AXIAL_FORCE, BENDING_MOMENT-S, BENDING_MOMENT-T, SHEAR_FORCE-S, SHEAR_FORCE-T and TORSIONAL_MOMENT. In the AUI, the section forces/moment along the axial direction of the element are reported starting at local node 1.

2.4.2 Large displacement elastic beam element

- The element is used with the elastic-isotropic material model.
- It is assumed that large displacements/rotations can occur, but only small strains.
- The large displacement beam element assuming elastic conditions is used in the same way as the linear elastic beam element (see Section 2.4.1). The stiffness matrix and internal force vector are evaluated in closed form.
- The large displacements and rotations are taken into account through a co-rotational framework, in which the element rigid body motion (translations and rotations) is separated from the deformational part of the motion. For more information, see the following reference:

ref: B. Nour-Omid and C.C. Rankin, *Finite rotation analysis and consistent linearization using projectors*, Comput. Meth. Appl. Mech. Engng. (93) 353-384, 1991.

- As for the linear elastic beam elements, the section flexural, shear and torsional properties can be directly input or the dimensions of the cross-section can be defined.

- This beam element is primarily suited for elastic stability analysis, and in this case the option of a general beam section is important.
- Note that warping effects are only included very approximately by a modification of the torsional rigidity, as input.
- The calculated quantities for this beam element are only the element end forces/moments, and the output is as for the linear elastic beam element.

2.4.3 Nonlinear elasto-plastic beam element

- The element is used with the plastic-bilinear material model with isotropic hardening.
- The element can be used either with the small displacement formulation (in which case the formulation is materially-nonlinear-only) or with the large displacement formulation (in which case the displacements/rotations can be large and a UL formulation is employed). In all cases, the strains are assumed to be small.
- The nonlinear elasto-plastic beam element can only be employed for circular and rectangular cross-sections, see Fig. 2.4-8.

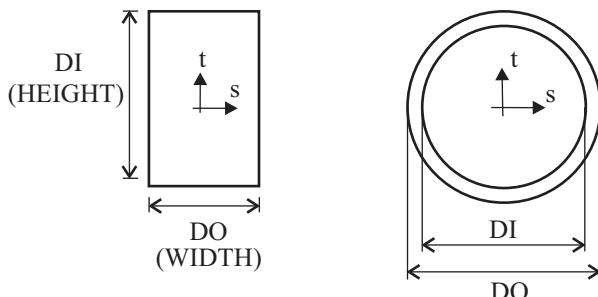


Figure 2.4-8: Possible cross-sections for nonlinear analysis

- The element can either be used using a 2-D action option (in which case the element must be defined in the global X-Y, X-Z or Y-Z planes) or a 3-D action option (in which case the element can have an arbitrary orientation).

- The beam element matrices are formulated using the Hermitian displacement functions, which give the displacement interpolation matrix summarized in Table 2.4-2.

Table 2.4-2: Beam interpolation functions not including shear effects

$$\begin{bmatrix} \mathbf{h}_{u_r} \\ \mathbf{h}_{u_s} \\ \mathbf{h}_{u_t} \end{bmatrix} = \begin{bmatrix} 1 - \frac{r}{L} & 6\Psi_1 \frac{s}{L} & 6\Psi_1 \frac{t}{L} & 0 & \Psi_2 t & -\Psi_2 s & \frac{r}{L} & -6\Psi_1 \frac{s}{L} \\ 0 & \Psi_4 & 0 & -\left(1 - \frac{r}{L}\right)t & 0 & \Psi_5 L & 0 & \Psi_6 \\ 0 & 0 & \Psi_4 & \left(1 - \frac{r}{L}\right)s & -\Psi_5 L & 0 & 0 & 0 \\ -6\Psi_1 \frac{t}{L} & 0 & -\Psi_3 t & \Psi_3 s & -(1 - 6\Psi_1)t & (1 - 6\Psi_1)s \\ 0 & -\frac{r}{L}t & 0 & -\Psi_1 r & 0 & (r - \Psi_7)L \\ \Psi_6 & \frac{r}{L}s & \Psi_1 r & 0 & (r - \Psi_7)L & 0 \end{bmatrix}$$

in which

$$\Psi_1 = \frac{r}{L} - \left(\frac{r}{L}\right)^2; \quad \Psi_2 = 1 - 4\frac{r}{L} + 3\left(\frac{r}{L}\right)^2; \quad \Psi_3 = 2\frac{r}{L} - 3\left(\frac{r}{L}\right)^2$$

$$\Psi_4 = 1 - 3\left(\frac{r}{L}\right)^2 + 2\left(\frac{r}{L}\right)^3; \quad \Psi_5 = \frac{r}{L} - 2\left(\frac{r}{L}\right)^2 + \left(\frac{r}{L}\right)^3$$

$$\Psi_6 = 3\left(\frac{r}{L}\right)^2 - 2\left(\frac{r}{L}\right)^3; \quad \Psi_7 = \frac{r}{L} - 3\left(\frac{r}{L}\right)^2 + 2\left(\frac{r}{L}\right)^3$$

Note: displacements correspond to the forces/moment S_i in Fig. 2.4-1 after condensation of the last two columns containing the shear effects in the s- and t-directions.

- In the formulation, shear deformations can be included in beams with a rectangular section in an approximate way; namely, if requested, constant shear distortions γ_{rs} and γ_{rt} along the length of the beam are assumed, as depicted in Fig. 2.4-9. In this case the displacement interpolation matrix of Table 2.4-2 is modified for the additional displacements corresponding to these shear deformations.

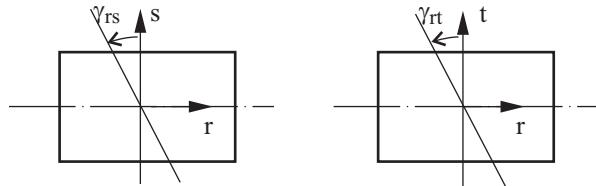


Figure 2.4-9: Assumptions of shear deformations through element thickness for nonlinear elasto-plastic beam element

- The interpolation functions in Table 2.4-2 do not account for warping in torsional deformations (absent if two-dimensional action is specified). The circular section does not warp, but for the rectangular section the displacement function for longitudinal displacements is corrected for warping as described in the following reference:

ref. K.J. Bathe and A. Chaudhary, "On the Displacement Formulation of Torsion of Shafts with Rectangular Cross-Sections", *Int. Num. Meth. in Eng.*, Vol. 18, pp. 1565-1568, 1982.

- Note that when using the small displacement formulation, the stiffness matrix is identical to the one used in linear elastic analysis if
 - ▶ The cross-section of the beam is circular (or rectangular) with $b = a$ or $b \gg a$, (because the exact warping functions are employed for $b = a$ and $b \gg a$, and the appropriate torsional rigidity is used).
 - ▶ The numerical integration is of high enough order

- ▶ The material is (still) elastic
- The derivation of the beam element matrices employed in the large displacement formulation is given in detail in the following paper:

ref. K.J. Bathe and S. Bolourchi, "Large Displacement Analysis of Three-Dimensional Beam Structures," *Int. J. Num. Meth. in Eng.*, Vol. 14, pp. 961-986, 1979.

The derivations in the above reference demonstrated that the updated Lagrangian formulation is more effective than the total Lagrangian formulation, and hence the updated Lagrangian formulation is employed in ADINA.

*ref. KJB
Section 6.6.3*

- All element matrices in elasto-plastic analysis are calculated using numerical integration. The default integration orders are given in Table 2.4-3. The locations and the labeling of the integration points are given in Fig. 2.4-10.
- The elasto-plastic stress-strain relation is based on the classical flow theory with the von Mises yield condition and is derived from the three-dimensional stress-strain law using the following assumptions:
 - the stresses τ_{ss} and τ_{tt} are zero
 - the strain γ_{st} is zero

Hence, the elastic-plastic stress-strain matrix for the normal stress τ_{rr} and the two shear stresses τ_{rs} and τ_{rt} is obtained using static condensation.

- In elasto-plastic analysis, either end forces or element stresses can be requested. Note that if end forces are requested in an elastic-plastic analysis, the state of stress in the element (whether elastic or plastic) is not indicated.

Table 2.4-3: Recommended integration orders in elasto-plastic beam analysis

Coordinate	Section	Analysis	Integration scheme	Integration orders	
				Default	Maximum
r	Any	Any	N.C.(**)	5	7
	Rect.	2-D action in r-s plane		3	7
s		3-D action	N.C.	7 ^(*)	7
Pipe	Any			3	7
	Rect.	2-D action in r-s plane		1	7
t or θ		3-D	N.C.	7 ^(*)	7
Pipe	r-s plane	Composite trapezoidal rule	5	5	
	3-D		8	8	

(*) Currently, seven point integration is used in the s- and t-directions for the rectangular section in 3-D action, regardless of user input.

(**) N.C.: Newton-Cotes

Stresses: Each element outputs, at its integration points, the following information to the porthole file. This information is accessible in the AUI using the variable names PLASTIC-FLAG, STRESS-RR, STRESS-RS, STRESS-RT, STRAIN-RR, STRAIN-RS, STRAIN-RT, PLASTIC_STRAIN-RR, PLASTIC_STRAIN-RS, PLASTIC_STRAIN-RT, YIELD_STRESS, ACCUM_EFF_PLASTIC_STRAIN

See Section 12.1.1 for the definitions of those variables that are not self-explanatory.

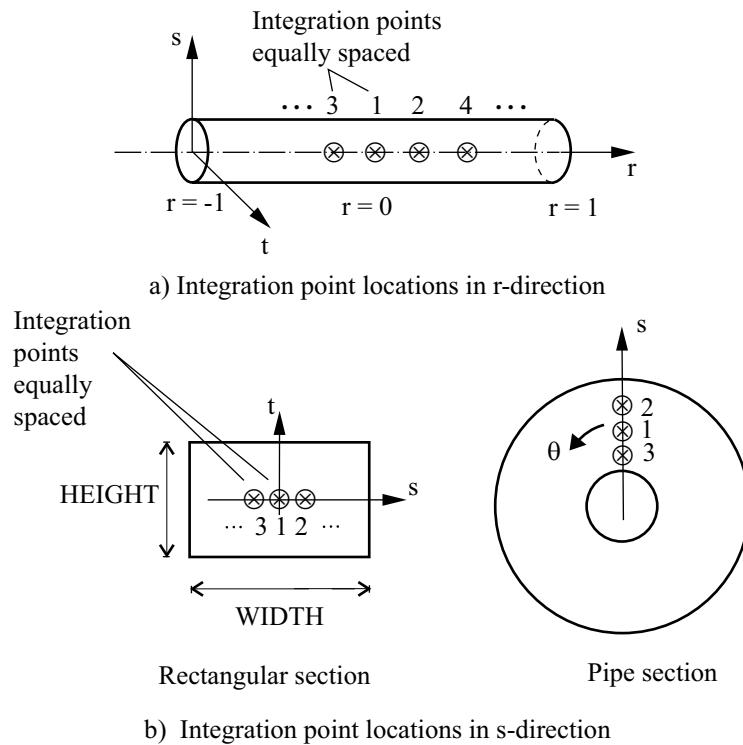


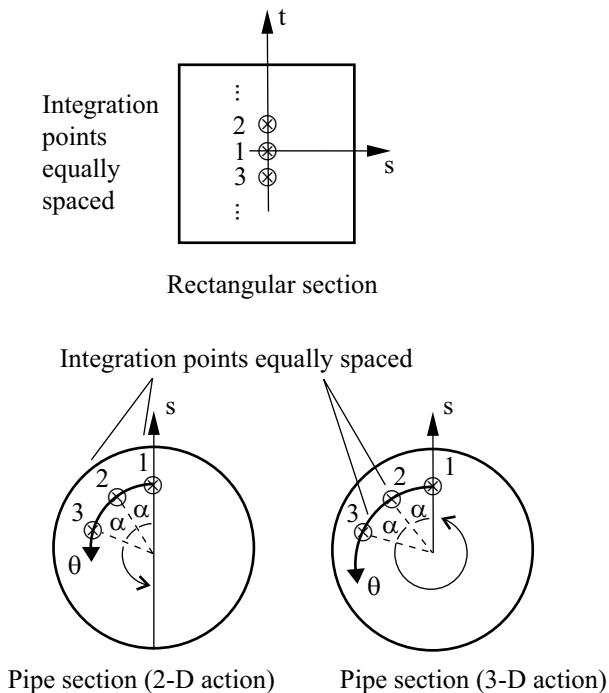
Figure 2.4-10: Integration point locations in elasto-plastic beam analysis

Nodal forces: Same as for the elastic beam element.

2.4.4 Moment-curvature beam element

- In practical engineering analysis, the data available for the description of the behavior of beam members may be given only in the form of relationships between bending moment and curvature, and between torsional moment and angle of twist. ADINA offers the capability of directly using these data without having to define an "equivalent" stress-strain law and the exact beam cross-sectional shape.

This element is suitable for modeling nonlinear elastic and elasto-plastic beam problems involving arbitrary cross-sections, especially cross-sections that are neither rectangular nor circular.

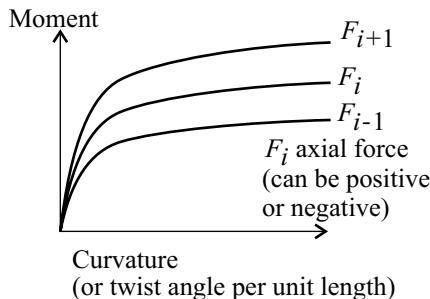


c) Integration point locations in t-direction

Figure 2.4-10: (continued)

- In the moment-curvature input, it is assumed that both the centroid and the shear center of the beam cross-section lie on the r-axis of the element: transverse forces applied to the element cannot generate twisting (as with a shear center offset) and axial forces applied to the element cannot generate bending (as with a centroid offset). However, you can create off-centered beam elements by using the rigid link (see Figure 2.4-2 and Section 5.11.2).
- The flexural and the torsional behavior of the beam element are respectively described by "bending moment vs. curvature" and "torsional moment vs. angle of twist" relationships. These relationships are input in ADINA in the form of multilinear functions (see Figure 2.4-11).

Typical beam data set:



ADINA input:

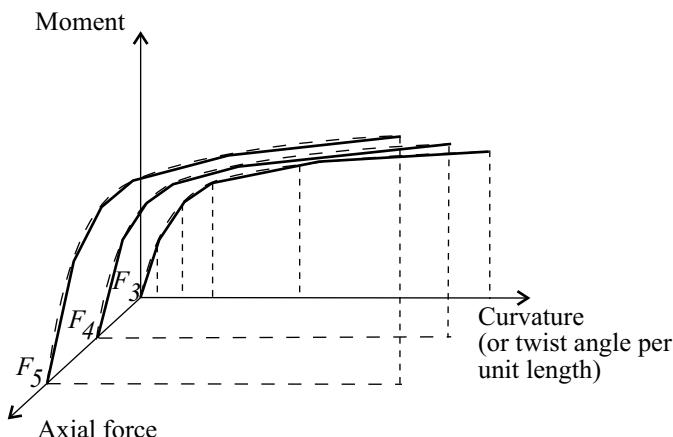


Figure 2.4-11: Input curves for the moment-curvature models
(curves are shown only for positive forces, moments
and curvatures/ twists)

- The "bending moment vs. curvature" and "torsional moment vs. angle of twist" relationships are functions of the axial force. Note that the axial force is positive when the element is in tension and is negative when the element is in compression. The sign conventions used for moments and torsion is that used for local node 2 in Figure 2.4-1(b).
- The flexural behavior of the beam element is defined by two "bending vs. curvature" relationships, one for each principal plane of inertia. Curvature is defined as $\chi = \frac{1}{\rho}$ where ρ is the radius

of curvature. For a linear element (linear elastic material, small displacements/small strains), the relationship between bending

$$\text{moment and curvature is } \frac{M}{\chi} = EI.$$

- The input for the torsional behavior of the element is similar to that for bending: A multilinear function is used to express the torsional moment in terms of the angle of twist per unit length.
- Note that the beam element does not include any shear deformation when the moment-curvature description is used.
- Note also that warping effects must be taken into account in the definition of the "torsional moment vs. twist" input curve if necessary.
- The rigid-end and end-release capabilities described in Section 2.4 can be used with this element.
- For computation of the beam element stiffness matrix and internal force vector, numerical integration (Newton-Cotes) is only used for integration along the length of the element (r-axis) and no integration is needed over the cross-section (s-axis and t-axis).
- Thermal effects can be included in moment-curvature relations.

Nonlinear elastic model: A nonlinear elastic model can be used (see Figure 2.4-12). In this case, the behavior for negative curvatures (resp. twist angles) may be different than the behavior for positive curvatures (resp. twist angles). The axial force versus axial deformation relationship is always linear elastic.

Note that the last segments of the moment-curvature curve are extrapolated if necessary, in order to calculate the moment when the curvature or twist angle per unit length is out of the input curve range. The end points of the curve do not represent rupture points.

ADINA input for a given axial force:

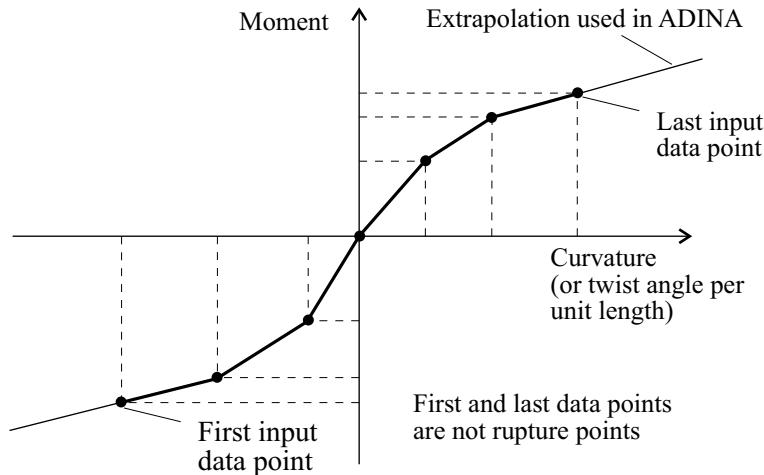


Figure 2.4-12: Nonlinear elastic moment-curvature beam input

Elastic-plastic model: For the analysis of beam members undergoing plastic deformations, ADINA offers bilinear and multilinear plasticity. Hardening can be linear isotropic, linear kinematic or linear mixed, as shown in Figure 2.4-13.

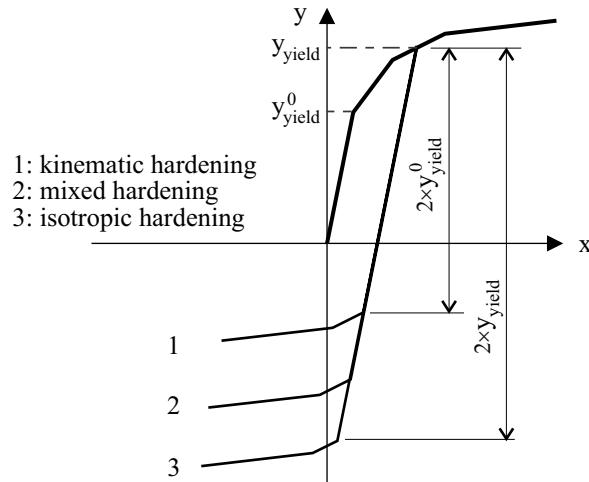


Figure 2.4-13: Hardening models for moment-curvature beams

The moment-curvature plasticity model consists of uniaxial plasticity laws respectively applied to the axial strain, each bending curvature, and the twist angle per unit length:

Axial force/axial strain relationship: The relationship can either be symmetric or non-symmetric with respect to the sign of the axial strain.

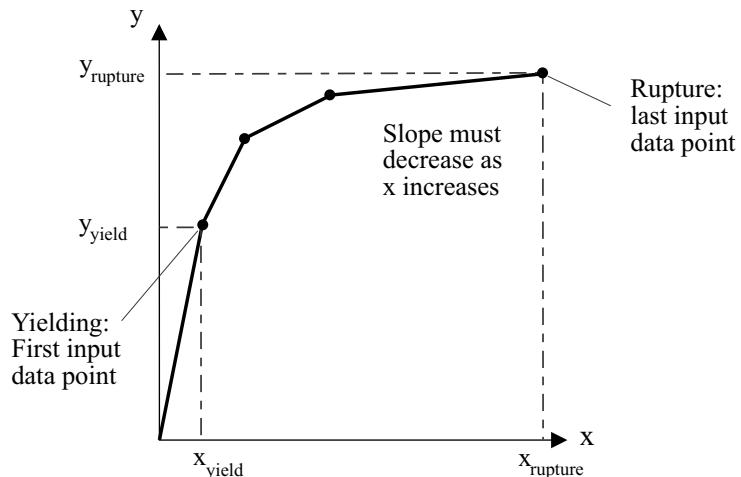
Bending moment/curvature relationship: The relationship can either be symmetric or non-symmetric with respect to the sign of the curvature. The relationship can depend on the axial force and can be different in axial tension and axial compression.

Torsional moment/twist angle per unit length relationship: The relationship can either be symmetric or non-symmetric with respect to the sign of the twist angle per unit length. The relationship can depend on the axial force and can be different in axial tension and axial compression.

In the symmetric case (Figure 2.4-14), you enter only the positive section of the axial force/axial strain, bending moment/curvature or torsional moment/twist angle curve. The first data point always corresponds to yielding and the last data point always corresponds to rupture.

In the non-symmetric case (Figure 2.4-15), you enter the entire axial force/axial strain, bending moment/curvature or torsional moment/twist angle curve. The first data point always corresponds to negative rupture and the last data point always corresponds to positive rupture. One data point – the zero point – must be at the origin. The data point prior to the zero point corresponds to the negative yield and the data point after the zero point corresponds to the positive yield. A different number of data points can be used for the positive and negative sections of the curve.

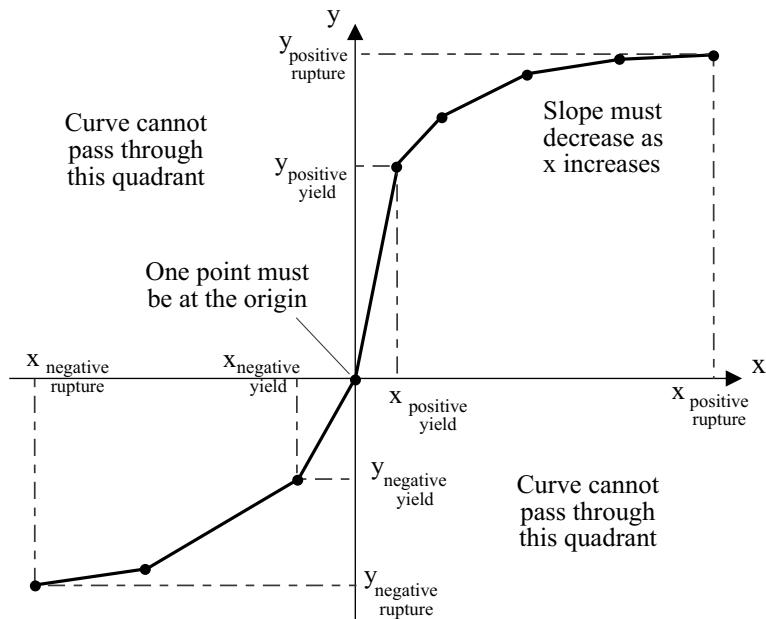
- To obtain bilinear plasticity, define a multilinear plasticity curve with only two segments.



(x, y) : (axial strain, axial force), or
 (curvature, bending moment) for a given axial force, or
 (angle of twist/unit length, torsional moment) for a given
 axial force

Figure 2.4-14: Symmetric elasto-plastic moment-curvature beam input

- Bending moment and torsion relationships can depend on the axial force and this dependence can be different in tension and in compression.
- The same element section can be plastic with respect to the axial deformation, but still elastic with respect to bending or torsion.
 The same remark applies for rupture.
- Rupture depends on the value of the accumulated effective plastic axial strain, accumulated effective plastic curvature or accumulated effective plastic angle of twist per unit length.



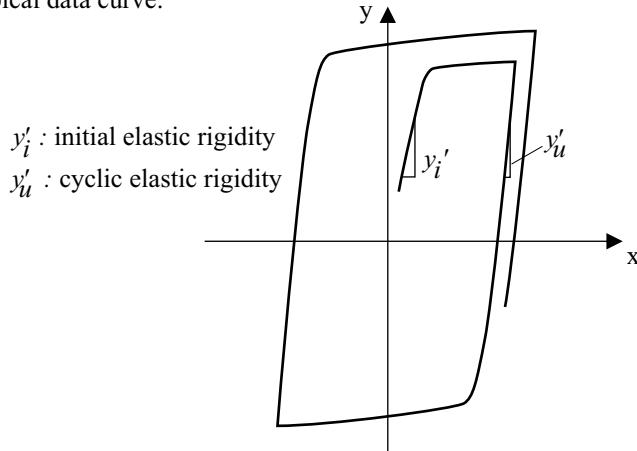
(x, y) : (axial strain, axial force), or
 (curvature, bending moment) for a given axial force, or
 (angle of twist/unit length, torsional moment) for a given
 axial force

First point: negative rupture point
 Point before origin: negative yield point
 Point after origin: positive yield point
 Last point: positive rupture point

Figure 2.4-15: Non-symmetric elasto-plastic moment-curvature beam input

- The elastic-plastic model includes a special option for cyclic behavior, in which the elastic rigidities when unloading first starts differ from the initial rigidities (see Figure 2.4-16). With this option, the elastic rigidity (in bending, torsion or axial deformation) changes as soon as the yield stress is reached. The ratio of the new cyclic elastic rigidity to the initial elastic rigidity is defined by user input. Once updated, the elastic rigidity remains constant for the rest of the analysis, for loading and unloading.

Typical data curve:



ADINA result (shown here for the bilinear case):

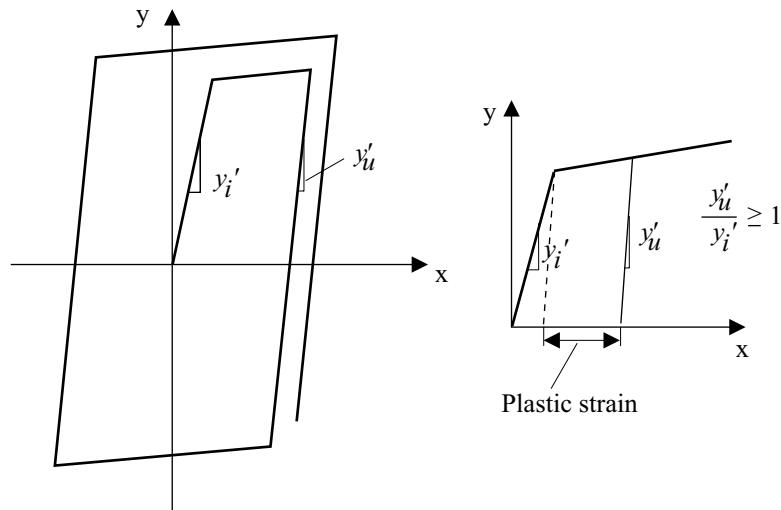


Figure 2.4-16: Typical material curve with a cyclic rigidity different than the initial rigidity

- In moment-curvature models, the cross-sectional area and the cross-sectional moments of inertia need to be input for the mass matrix computation (in case of mass proportional loading or dynamic analysis). They are used only for mass matrix computation.

- The results can be obtained in the form of element nodal forces and moments, or in the form of stress resultants given at the integration point locations on the element.

Stress resultants: Each element outputs, at the integration point locations on the element r-axis, the following information to the porthole file, based on the material model. This information is accessible in the AUI using the given variable names.

The integration point locations on the element r-axis are shown in Figure 2.4-17. In the AUI, these integration point locations are considered to be section points, not element result points (see Section 12.1.1).

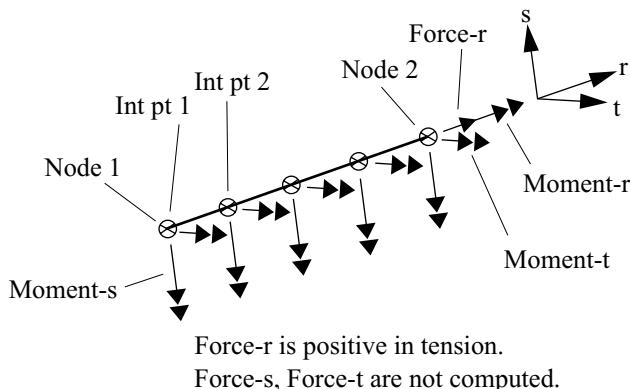


Figure 2.4-17: Stress resultant output option for the moment-curvature models

The results are always given in the element local coordinate system (r,s,t). Note that a positive value of the Moment-s at a section point location points in the negative s-direction.

Nonlinear elastic, user supplied: AXIAL_FORCE,
TORSIONAL_MOMENT, BENDING_MOMENT-S,
BENDING_MOMENT-T, AXIAL_STRAIN, TWIST,
CURVATURE-S, CURVATURE-T

Plastic-multilinear: PLASTIC_FLAG_AXIAL,
PLASTIC_FLAG_TORSION, PLASTIC_FLAG_BENDING-S,
PLASTIC_FLAG_BENDING-T, AXIAL_FORCE,
TORSIONAL_MOMENT, BENDING_MOMENT-S,
BENDING_MOMENT-T, AXIAL_STRAIN, TWIST,

```
CURVATURE-S, CURVATURE-T,  
PLASTIC_AXIAL_STRAIN, PLASTIC_TWIST,  
PLASTIC_CURVATURE-S, PLASTIC_CURVATURE-T,  
ACCUM_PLASTIC_AXIAL_STRAIN,  
ACCUM_PLASTIC_TWIST,  
ACCUM_PLASTIC_CURVATURE-S,  
ACCUM_PLASTIC_CURVATURE-T, YIELD_AXIAL_FORCE,  
YIELD_TORSIONAL_MOMENT,  
YIELD_BENDING_MOMENT-S,  
YIELD_BENDING_MOMENT-T
```

- Note that in a materially nonlinear analysis, the element nodal moments are in general not equal to the moment stress resultants at the integration points located at the nodes. The reason for this is that in such cases the beam element does not exactly satisfy the element internal equilibrium on the differential level; that is, the element is a "true finite element". The differential equilibrium is, of course, satisfied more accurately as the finite element mesh is refined.
- Transverse forces are not computed at integration points since the ADINA beam element formulation uses the equilibrium equations of the complete element to directly determine the transverse forces at the element end nodes.
- In the case of beam elements with rigid ends, the element stress resultants are computed at the integration point locations corresponding to the flexible part of the element, excluding the rigid ends.

Nodal forces: Same as for the elastic beam element.

2.4.5 Beam-tied element

- The beam-tied element is used for the modeling of a shaft connecting shell surfaces. It consists of a beam element with additional constraint equations. These constraints couple the torsional angular displacements of the beam to the in-plane displacement of the shell surfaces. Specifically the torsional angle at one end of the beam is computed from the in-plane displacements of the nodes of the shell elements connected to the beam end.

- A typical use of the beam-tied element is shown in Figure 2.4-18. Note that the mesh around the bolt-shaft must be fine enough so that the first ring of shell elements around the bolt-shaft ends span an area about the size of the actual connection device: too coarse a mesh will result in an over-constrained model. Therefore, the use of the beam-tied element is restricted to the modeling of a single isolated shaft. This feature is not meant to be used for the modeling of riveted, or spot-welded, or line-welded plates.

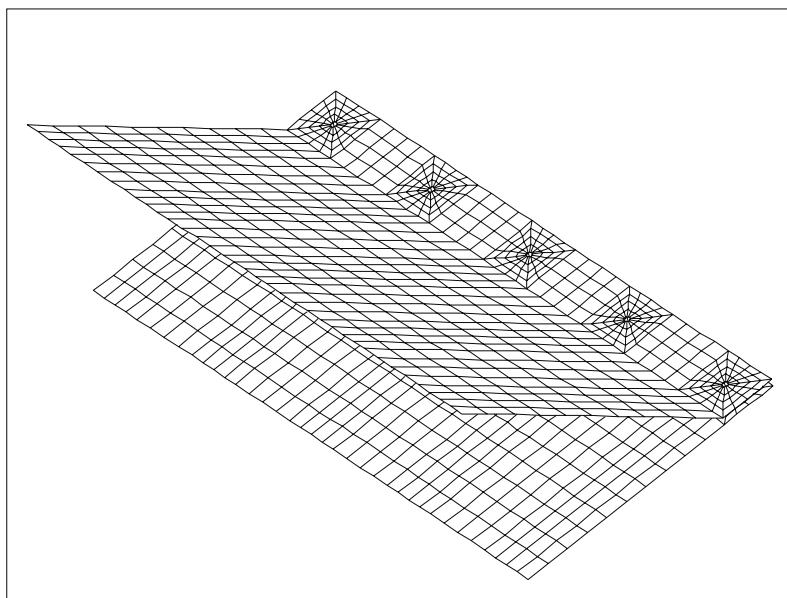


Figure 2.4-18: Example in which the beam-tied element is used

- The automatically generated constraint equations can be applied at either one or two ends of the beam-tied element. If applied at only one end, the beam-tied element is allowed to spin (please note that a mechanism is then introduced in the model).
- Constraint equations are not automatically generated at those nodes where the individual shell normals at the node differ than more than a preset angle from the average shell normal at the node (this preset angle is called the bolt angle).

- The beam-tied element must have a finite length, i.e. the end nodes must be further apart than a minimum distance. This minimum distance can be relative (proportional to the model dimensions) or absolute. For nodes that are closer than this minimum distance, the spring-tied element should be used instead, see Section 2.9.3.

2.4.6 Bolt element

- A bolt element is a pre-stressed Hermitian beam or pipe element with a given initial force. The bolt force is held constant during the equilibrium iterations at the time of solution start, so that the initial force at the start of the “zero” solution step is the input initial force, no matter what other external loads or initial strains/stresses are input.
- During the solution for the first, second and successive solution steps, the bolt force can vary depending on the loads applied to the model.
- Bolt elements are created in a beam or pipe element group with the bolt option activated, e.g., EGROUP BEAM ... OPTION = BOLT. The initial force is entered either on the geometry (e.g., AUI command LINE-ELEMENTDATA BEAM) or directly specified on the elements (e.g., AUI command EDATA).
- The formulation used for the bolt elements can either be small displacements or large displacements. Any cross-section available for the beam element can be used, but only the isotropic elastic material model can be used. For the pipe-bolt option, only a circular cross-section is applicable.
- In the bolt force calculations we iterate as follows:

$${}^0\mathbf{K}^{(i-1)}\Delta\mathbf{U}^{(i)} = \mathbf{R}_I^{(i-1)} - {}^0\mathbf{F}^{(i-1)}$$

and

$${}^0\mathbf{U}^{(i)} = {}^0\mathbf{U}^{(i-1)} + \Delta\mathbf{U}^{(i)}$$

where $\mathbf{R}_I^{(i-1)}$ is the consistent nodal point force corresponding to the forces in the bolt elements.

- The bolt force convergence is checked for every bolt element m :

$$\frac{|R_m - R_m^0|}{R_m^0} < \text{TOLL}$$

where R_m^0 is the user-input bolt force for element m , and R_m is the current force in bolt element m .

- The $\mathbf{R}_I^{(i-1)}$ vector is calculated from

$$\mathbf{R}_I^{(i-1)} = \int_{\Omega_V} \mathbf{B}^{(i-1)T} \boldsymbol{\tau}^{(i-1)} dV$$

- For every bolt element m , the stress vector corresponds to the mechanical deformations plus updated initial stresses:

$$\boldsymbol{\tau}_m^{(i)} = \boldsymbol{\tau}_m^{(0)} + \alpha \cdot \boldsymbol{\tau}_m^{(i-1)}$$

where $\boldsymbol{\tau}_m^0 = \frac{\mathbf{R}_m^0}{A_m}$, A_m = cross-sectional area of bolt element m

and α is a stress-level based factor.

2.4.7 Element mass matrices

- The beam element can be used with a lumped or a consistent mass matrix, except for explicit dynamic analysis which always uses a lumped mass.
- The consistent mass matrix of the beam element is evaluated in closed form, and does not include the effect of shear deformations. This matrix is derived in the following reference.

ref. J.S. Przemieniecki, *Theory of Matrix Structural*

Analysis, McGraw-Hill Book Co., 1968.

- The beam element can be used with a lumped or a consistent mass matrix, except for explicit dynamic analysis which always uses a lumped mass.
- The consistent mass matrix of the beam element is evaluated in closed form, and does not include the effect of shear deformations.
- The lumped mass for translational degrees of freedom is $M / 2$ where M is the total mass of the element.

The rotational lumped mass for all except explicit dynamic

$$\text{analysis is } M_{rr} = \frac{2}{3} \cdot \frac{M}{2} \cdot \frac{I_{rr}}{A}, \text{ in which } I_{rr} = \text{polar moment of}$$

inertia of the beam cross-section and A = beam cross-sectional area. This lumped mass is applied to all rotational degrees of freedom in order to obtain a diagonal mass matrix in any coordinate system.

The rotational lumped mass for explicit dynamic analysis is

$$M_{rr} = 3 \cdot \frac{M}{2} \cdot \frac{I_m}{A} \text{ where } I_m = \max(I_{ss}, I_{tt}) \text{ is the maximum}$$

bending moment of inertia of the beam. This lumped mass is applied to all rotational degrees of freedom. Note that this scaling of rotational masses ensures that the rotational degrees of freedom do not affect the critical stable time step of the element.

- The rotational lumped masses can be multiplied by a user-specified multiplier ETA (except in explicit analysis).

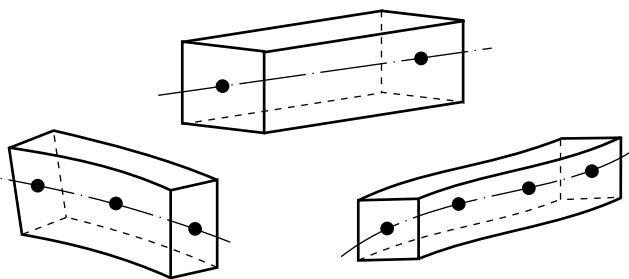
2.5 Iso-beam elements; axisymmetric shell elements

2.5.1 General considerations

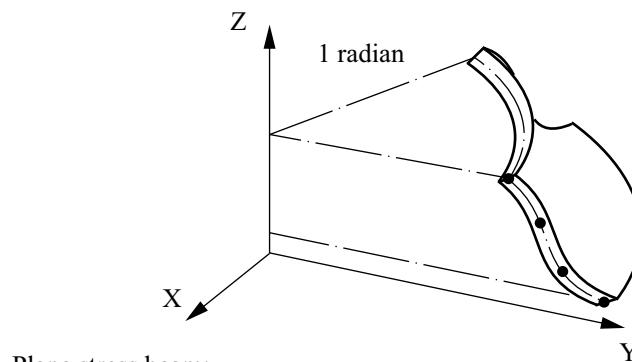
- The iso-beam elements (which include the axisymmetric shell elements) can be employed in the following forms, see Fig. 2.5-1:
plane stress 2-D beam with three degrees of freedom per node,
plane strain 2-D beam with three degrees of freedom per node,
axisymmetric shell with three degrees of freedom per node,
general 3-D beam with six degrees of freedom per node.

- The plane stress 2-D beam element is identical to the 3-D beam element but constrained to act only in the YZ plane. Hence, all motion of the 2-D plane stress beam element must occur in the YZ plane (see Fig. 2.5-2).
- The difference between the plane strain and the plane stress 2-D beam elements is that for the plane strain element, it is assumed that the out-of-plane strain ε_{xx} is equal to zero whereas the out-of-plane stress σ_{xx} is equal to zero for the plane stress element.
- Note that it can be significantly more effective to use the 2-D beam option instead of the general 3-D beam option, since then the numerical integration is only performed in two dimensions and the number of degrees of freedom is also reduced.

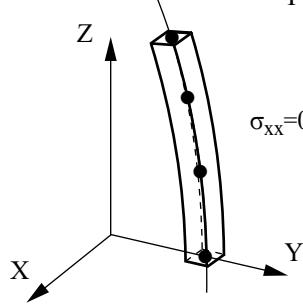
Examples of general 3-D iso-beam elements:



Axisymmetric shell:



Plane stress beam:



Plane strain beam:

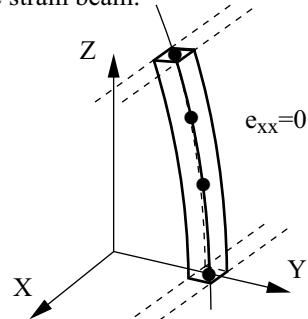
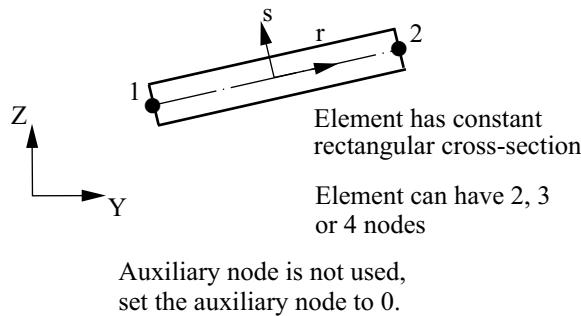
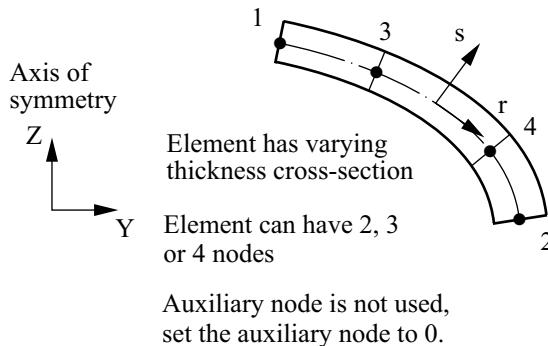


Figure 2.5-1: 2-D beam, axisymmetric shell, and 3-D beam elements



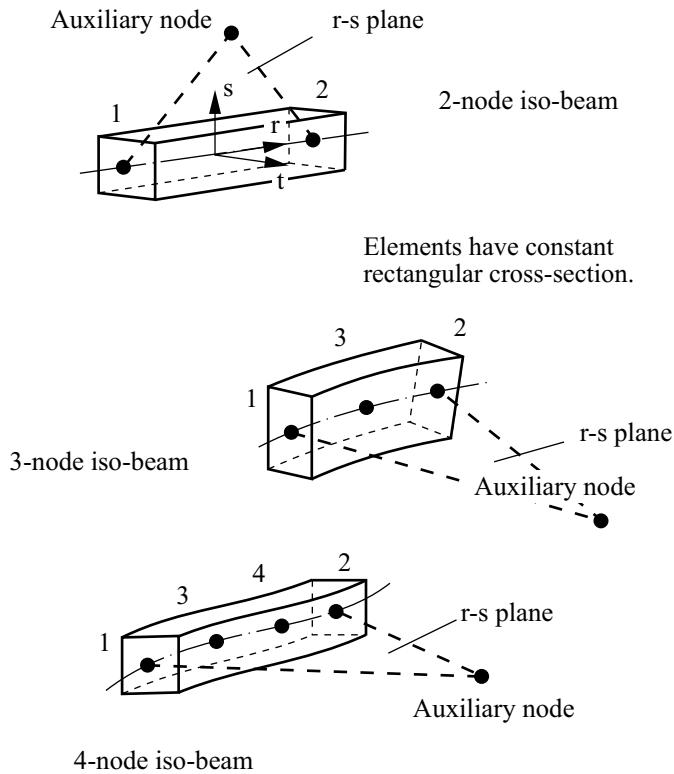
a) 2-node plane stress or plane strain beam element



b) 4-node axisymmetric shell element

Figure 2.5-2: Local node numbering; natural coordinate system

- The axisymmetric shell element formulation is an extension of the 2-D beam element formulation in that the axisymmetric hoop stress/strain components are included in the model. The axisymmetric shell element must be defined in the YZ plane, and lie in the +Y half plane.
- The cross-sectional areas of each of these elements are assumed to be rectangular. The 2-D and 3-D beam elements can only be assigned a constant area cross-section. The axisymmetric shell element can be assigned a varying thickness.



c) General 3-D isoparametric beam elements.

Figure 2.5-2: (continued)

- The elements can be employed with 2, 3 or 4 nodes. The 3 and 4-node elements can be curved, but it should be noted that the element nodes must initially lie in one plane (which defines the r-s plane).
- To model the torsional stiffness of the 3-D beam element accurately, the warping behavior is represented as described in the following reference:

ref. K.J. Bathe and A. Chaudhary, "On the Displacement Formulation of Torsion of Shafts with Rectangular Cross-Sections", *Int. J. Num. Meth. in Eng.*, Vol. 18, pp. 1565-1568, 1982.

- Note that the formulation directly models shear deformations in an approximate manner; namely, the shear deformations are assumed to be constant over the cross-section (see ref. KJB, Fig. 5.18, p. 398). This corresponds to using shear factors equal to 1.0. The elements can be employed to model thin and thick beams and shells.
- Some applications using the element are published in the following reference.

ref. K.J. Bathe and P.M. Wiener, "On Elastic-Plastic Analysis of I-Beams in Bending and Torsion", *Computers and Structures*, Vol. 17, pp. 711-718, 1983.

- The isoparametric beam element is available in ADINA primarily to model
 - ▶ Curved beams
 - ▶ Stiffeners to shells, when the shell element (described in Section 2.7) is used
 - ▶ Beams in large displacements
 - ▶ Axisymmetric shells under axisymmetric loading

2.5.2 Numerical integration

ref. KJB
Sections 5.4.1
and 6.5.1

- The element matrices and vectors are formulated using the isoparametric interpolation, and Gauss or Newton-Cotes numerical integration is used to evaluate these matrices in all analyses. For the 2-D beam and axisymmetric shell elements, numerical integration is only performed in the r-s plane, hence these elements are considerably less expensive in terms of computer time than the general 3-D element. The locations and labeling of the integration points are illustrated in Fig. 2.5-3.
- The elements are based upon a mixed interpolation of displacements and stresses.

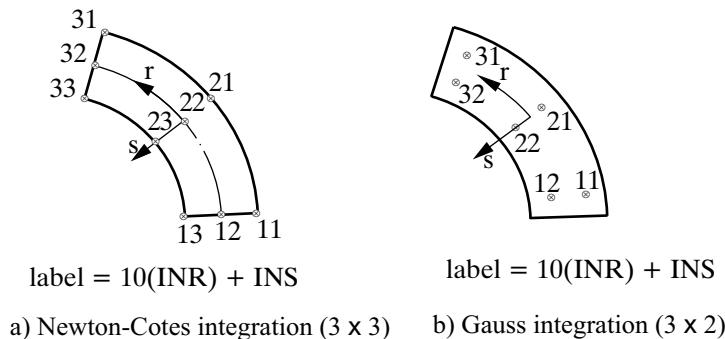


Figure 2.5-3: Examples of integration point numbering for 2-D beam, axisymmetric shell, and 3-D beam elements

- Only the default integration order along the r-direction should be used for the 2-, 3- and 4-node elements, i.e.:
 - ▶ 2-node iso-beam: 1-point integration
 - ▶ 3-node iso-beam: 2-point integration
 - ▶ 4-node iso-beam: 3-point integration

These elements do not contain any spurious zero energy modes, do not lock and are efficient in general nonlinear analysis.

- If, however, the default r-integration order is not used, the results of an analysis can change drastically, for example, if 2-point integration along the r direction is specified for the 2-node element model shown in Fig. 2.5-4, the resulting displacement (Δ) is 0.0126 (i.e., locking is observed).
- For the 3-D beam element, you can choose either 4x4 Gauss or 7x7 Newton-Cotes integration over the beam cross-section (along s and t).

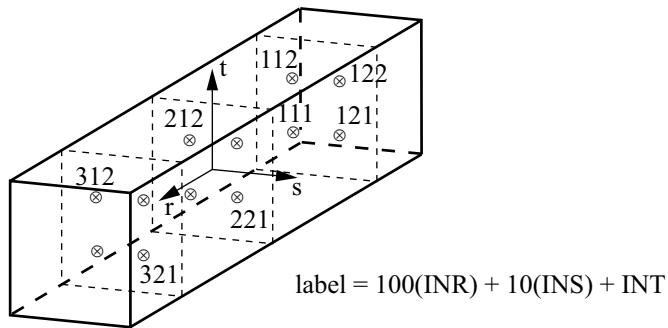
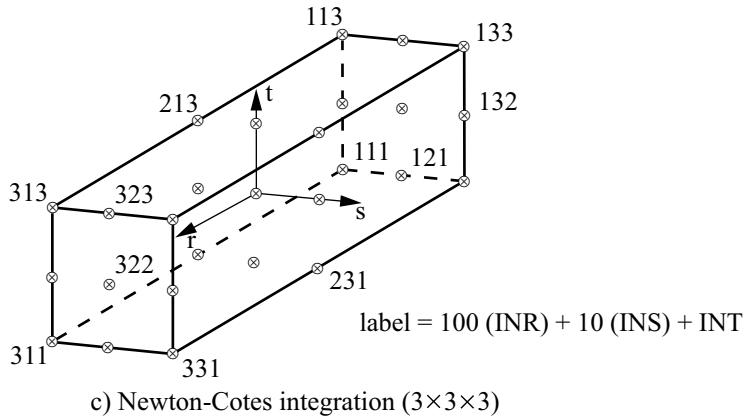


Figure 2.5-3: (continued)

2.5.3 Linear iso-beam elements

ref. KJB
Section 5.4.1

- The formulation of the element is presented in Chapter 5 of ref. KJB.
- It is assumed that the displacements, rotations, and strains are infinitesimally small and the elastic-isotropic material is used.

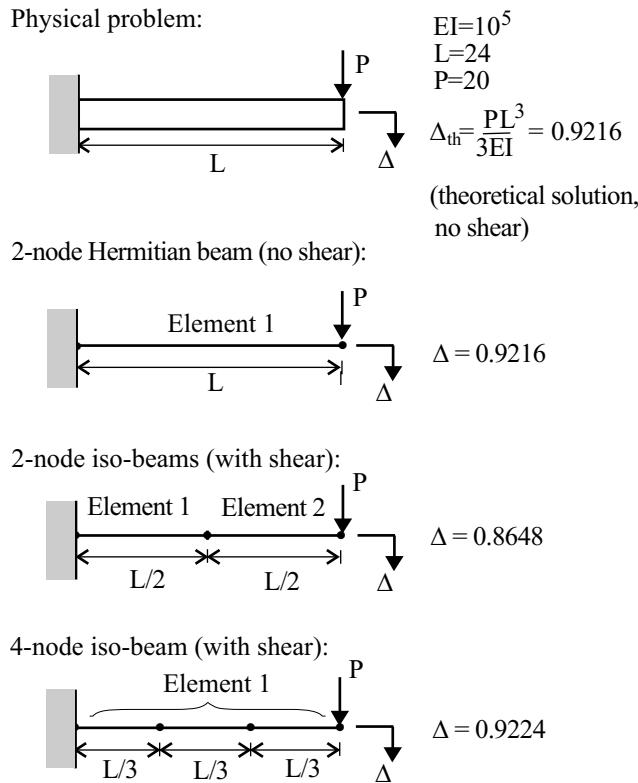
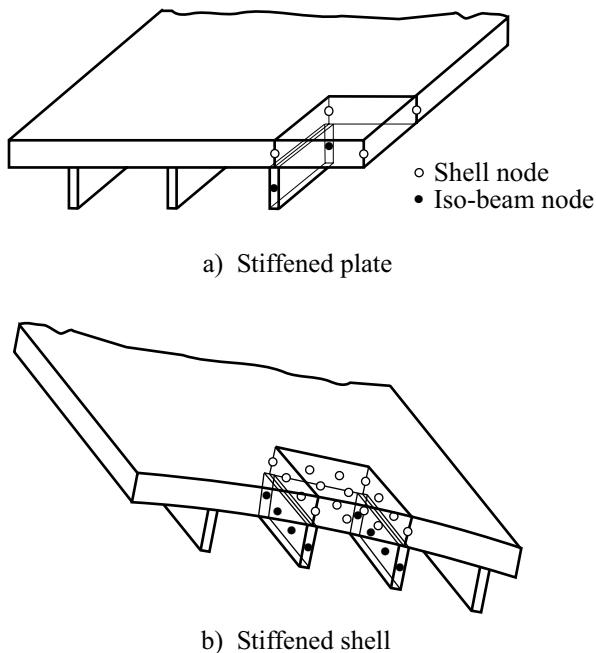


Figure 2.5-4: Results of the analysis of a cantilever

- Since the iso-beam element stiffness matrix is calculated using numerical integration, it is clear that the 2-node Hermitian beam element (see Section 2.4.1) is more effective when straight beam members in linear elastic analysis are considered. Hence, for the linear analysis of many structures (such as frames, buildings, shafts) only the 2-node Hermitian beam element should be used.
- Even when considering the linear analysis of a curved beam member, it is frequently more cost-effective to model this beam as an assemblage of straight 2-node Hermitian beam elements than to use the curved iso-beam element.

- Note that the iso-beam element can only be employed with a rectangular cross-section, whereas the linear 2-node Hermitian beam element can be used for any cross-section which can be defined by the principal moments of inertia, cross-sectional area and the shear areas input to ADINA.
- In linear analysis, the iso-beam element is primarily useful for modeling stiffeners to shells when the 4-node or 8-node or 16-node shell element (see Section 2.7) is employed. In this case the 2-, 3- or 4-node iso-beam elements, respectively, together with the shell elements can provide an effective finite element discretization of the stiffened shell structure (see Fig. 2.5-5).



Note: Select master nodes and use rigid links to tie the shell nodes and the iso-beam nodes together.

Figure 2.5-5: Models of stiffened plate and stiffened shell

- The solution results obtained using the isoparametric beam elements are compared with those obtained using the 2-node Hermitian beam elements in Fig. 2.5-4. Considering the tip loaded cantilever shown, a 2-node Hermitian beam element representation gives the "exact" analytical value of displacements and section forces/moment (compared with Bernoulli-Euler beam theory). The 2-node iso-beam element yields approximate results, while the 4-node iso-beam element is very accurate. However, as is typical in finite element analyses, the discrepancy in the solution becomes negligibly small when a sufficiently fine mesh is used.
- The forces/moment at the element nodes are (in linear analysis)

$$\mathbf{F} = \mathbf{K}\mathbf{U}$$

where \mathbf{K} is the element stiffness matrix and \mathbf{U} is the vector of nodal point displacements. Hence, at internal element nodes the forces/moment in \mathbf{F} will be equal to the forces/moment, applied externally or by adjoining elements, and are not the internal section forces/moment.

2.5.4 Nonlinear iso-beam elements

ref. KJB • The formulation of the element is presented in Chapter 6 of ref. *Section 6.5.1* KJB.

- The element can be used with the following material models: **elastic, plastic-bilinear, plastic-multilinear, thermo-isotropic, thermo-plastic, creep, plastic-creep, multilinear-plastic-creep, creep-variable, plastic-creep-variable, multilinear-plastic-creep-variable, shape-memory alloy.**
- The element can be used either with a **small displacement** or a **large displacement** formulation. In the large displacement formulation, large displacements and rotations are allowed. In all cases, only small strains are assumed.

All of the material models in the above list can be used with either formulation. The use of a linear material with the small displacement formulation corresponds to a linear formulation, and the use of a nonlinear material with the small displacement

formulation corresponds to a materially-nonlinear-only formulation.

- The iso-beam element can be particularly effective in geometric nonlinear analysis of straight and curved members, because the change in geometry due to the large displacements is modeled accurately.

ref. KJB
Section 6.6.3

- The element matrices are evaluated using Gauss or Newton-Cotes numerical integration. In elastic-plastic analysis, the stress-strain matrix is modified to include the effects of plasticity. This stress-strain matrix is based on the classical flow theory with the von Mises yield condition and is derived from the three-dimensional stress-strain law with the appropriate stresses and strains set to zero.
- For the elasto-plastic and the thermo-elasto-plastic and creep material models, the effective-stress-function algorithm of the references given below is used.

ref. M. Kojić and K.J. Bathe, "The Effective-Stress-Function Algorithm for Thermo-Elasto-Plasticity and Creep", *Int. J. Num. Meth. Engng.*, Vol. 24, No. 8, pp. 509-532, 1987.

ref. M. Kojić and K.J. Bathe, "Thermo-Elastic-Plastic and Creep Analysis of Shell Structures", *Computers & Structures*, Vol. 26, No 1/2, pp. 135-143, 1987.

2.5.5 Axisymmetric shell element

- The axisymmetric shell element can be thought of as a special case of the iso-beam formulation (in the same way as an axisymmetric element is a special case of the 2-D solid element).
- For the axisymmetric shell element, the hoop (circumferential) stress and strain components are included in the iso-beam formulation.
- The use of this element can be significantly more effective than

using axisymmetric 2-D solid elements when the shell structure to be modeled is thin.

- One radian of the axisymmetric shell is modeled when using this element (as with the 2-D axisymmetric solid element).

2.5.6 Element mass matrices

- The iso-beam element can be used with a lumped or a consistent mass matrix, except for explicit dynamic analysis which always uses a lumped mass.

*ref. KJB
Section 5.4.1* • The consistent mass matrix is calculated using the isoparametric formulation with the displacement interpolations given on p. 408 of ref. KJB.

- The lumped mass for degree of freedom i is $M \cdot \left(\frac{\ell_i}{L} \right)$, where M

is the total mass, L is the element length and ℓ_i is a fraction of the length associated with node i . The rotational mass is

$$\frac{1}{3} \cdot M \cdot \frac{\ell_i}{L} \left(\frac{1}{6} (b^2 + d^2) \right). \text{ For a 2-node iso-beam element}$$

$$\ell_1 = \ell_2 = \frac{L}{2}, \text{ while for a 4-node isobeam element } \ell_1 = \ell_2 = \frac{L}{6} \text{ and}$$

$$\ell_3 = \ell_4 = \frac{L}{3} \text{ (see Fig. 2.5-6).}$$

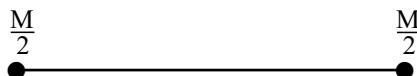
- The rotational lumped mass can be multiplied by a user-specified scalar ETA (except in explicit dynamic analysis).

2.5.7 Element output

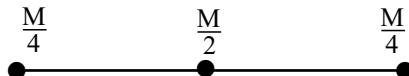
- You can request that ADINA either print or save stresses or forces.

Stresses: Each element outputs, at its integration points, the following information to the porthole file, based on the material

model. This information is accessible in the AUI using the given variable names.



a) 2-node element



Note: M = total mass
of element (2-D or 3-D
beam elements)

b) 3-node element



c) 4-node element

Figure 2.5-6: Construction of element lumped translational mass matrix for the beam option of the iso-beam element

General 3-D iso-beam, elastic-isotropic material: STRESS-RR, STRESS-RS, STRESS-RT, STRESS-SS, STRAIN-RR, STRAIN-RS, STRAIN-RT, STRAIN-SS, FE_EFFECTIVE_STRESS

Plane stress iso-beam, elastic-isotropic material: STRESS-RR, STRESS-RS, STRESS-SS, STRAIN-RR, STRAIN-RS, STRAIN-SS, FE_EFFECTIVE_STRESS

Plane strain iso-beam or axisymmetric shell, elastic-isotropic material: STRESS-RR, STRESS-RS, STRESS-TT, STRESS-SS, STRAIN-RR, STRAIN-RS, STRAIN-TT, STRAIN-SS, FE_EFFECTIVE_STRESS

General 3-D iso-beam, elastic-isotropic material, thermal effects: STRESS-RR, STRESS-RS, STRESS-RT, STRESS-SS, STRAIN-RR, STRAIN-RS, STRAIN-RT, STRAIN-SS,

FE_EFFECTIVE_STRESS, THERMAL_STRAIN,
ELEMENT_TEMPERATURE

Plane stress iso-beam, elastic-isotropic material, thermal effects:
STRESS-RR, STRESS-RS, STRESS-SS, STRAIN-RR,
STRAIN-RS, STRAIN-SS, FE_EFFECTIVE_STRESS,
THERMAL_STRAIN, ELEMENT_TEMPERATURE

Plane strain iso-beam or axisymmetric shell, elastic-isotropic
material, thermal effects: STRESS-RR, STRESS-RS,
STRESS-TT, STRESS-SS, STRAIN-RR, STRAIN-RS,
STRAIN-TT, STRAIN-SS, FE_EFFECTIVE_STRESS,
THERMAL_STRAIN, ELEMENT_TEMPERATURE

Thermo-isotropic material: STRESS(RST),
FE_EFFECTIVE_STRESS, STRAIN(RST),
THERMAL_STRAIN(RST), ELEMENT_TEMPERATURE

Plastic-bilinear, plastic-multilinear materials: PLASTIC_FLAG,
STRESS(RST), STRAIN(RST),
PLASTIC_STRAIN(RST), THERMAL_STRAIN(RST),
FE_EFFECTIVE_STRESS, YIELD_STRESS,
ACCUM_EFF_PLASTIC_STRAIN, ELEMENT_TEMPERATURE

Thermo-plastic, creep, plastic-creep, multilinear-plastic-creep,
creep-variable, plastic-creep-variable, multilinear-plastic-creep-
variable materials: PLASTIC_FLAG,
NUMBER_OF_SUBINCREMENTS, STRESS(RST),
FE_EFFECTIVE_STRESS,
STRAIN(RST), YIELD_STRESS,
PLASTIC_STRAIN(RST),
ACCUM_EFF_PLASTIC_STRAIN, CREEP_STRAIN(RST),
THERMAL_STRAIN(RST), ELEMENT_TEMPERATURE,
EFFECTIVE_CREEP_STRAIN

In the above lists,

STRESS(RST) = STRESS-RR, STRESS-RS,
STRESS-RT, STRESS-TT, STRESS-SS

with similar definitions for the other abbreviations used above.

See Section 12.1.1 for the definitions of those variables that are
not self-explanatory.

ref. KJB
Section 6.3

Nodal forces: Nodal point forces are obtained using the relation

$$\int_{t+\Delta t}^t \mathbf{F} = \int_{t+\Delta t}^t \mathbf{B}^T \mathbf{\tau} dV$$

where \mathbf{B} is the strain-displacement matrix, the stresses are stored in $\mathbf{\tau}$, V represents the volume, and the superscript $t + \Delta t$ refers to the conditions at time (load step) $t + \Delta t$.

These forces are accessible in the AUI using the variable names NODAL_FORCE-R, NODAL_FORCE-S, NODAL_FORCE-T, NODAL_MOMENT-R, NODAL_MOMENT-S, NODAL_MOMENT-T.

The end forces/momenta are computed at the element local nodes. In the AUI, element local nodes are defined as element points of type label. For example, to access the result computed at element 5, local node 2, define an element point of type label with element number 5, label number 2.

2.6 Plate/shell elements

- The plate/shell element is a 3-node flat triangular element that can be employed to model thin plates and shells.
- The element can be used with the following material models: **elastic-isotropic**, **elastic-orthotropic** and **Ilyushin** (plastic).
- The element can be used with a **small or large displacement** formulation. In the large displacement formulation, very large displacements and rotations are assumed. In both formulations, the strains are assumed small.

All of the material models in the above list can be used with either formulation. The use of a linear material with the small displacement formulation corresponds to a linear formulation, and the use of a nonlinear material with the small displacement formulation corresponds to a materially-nonlinear-only formulation.

2.6.1 Formulation of the element

- The element has 6 degrees of freedom per node.
- The stiffness matrix is constructed by superimposing the bending and membrane parts in the current local coordinate system.
The bending part is the discrete Kirchhoff bending element described in the following reference:

ref. J.-L. Batoz, K.J. Bathe, and L.W. Ho, "A Study of Three-Node Triangular Plate Bending Elements", *Int. J. Num. Meth. in Eng.*, Vol. 15, pp. 1771-1812, 1980.

The membrane part is the constant strain plane stress 3-node element (see ref. KJB, pp. 202-205).

- The complete element formulation is summarized and discussed in the following references:

ref. K.J. Bathe and L.W. Ho, "A Simple and Effective Element for Analysis of General Shell Structures", *J. Computers and Structures*, Vol. 13, pp. 673-681, 1981.

ref. K.J. Bathe, E. Dvorkin, and L.W. Ho, "Our Discrete-Kirchhoff and Isoparametric Shell Elements for Nonlinear Analysis – An Assessment", *J. Computers and Structures*, Vol. 16, pp. 89-98, 1983.

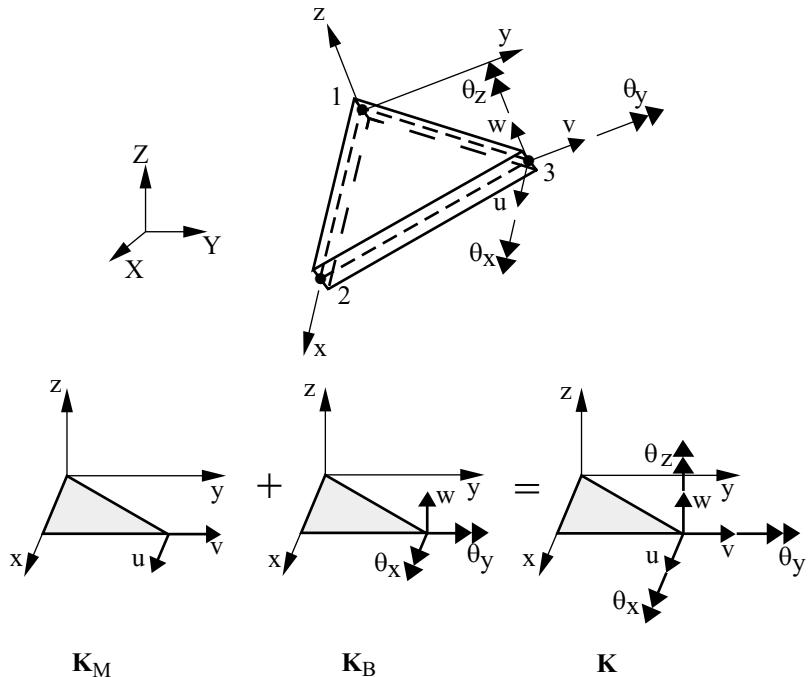
- Figure 2.6-1 illustrates how the element stiffness matrix is constructed. Note that in the local coordinate system the matrices \mathbf{K}_B and \mathbf{K}_M do not contain stiffness corresponding to the rotation θ_z . In the program, this stiffness is set equal to

$$K_{\theta_z} = \text{a value of } (k_{\theta_x}, k_{\theta_x}) \times 10^{-4}$$

in order to remove the zero stiffness but yet not affect the analysis results significantly.

- In the linear analysis of flat plates that lie parallel to the global coordinate planes, the rotational degrees of freedom normal to the

plane of the plates would best be deleted, since the plate/shell elements do not provide actual stiffness to these degrees of freedom. Fig. 2.6-2 illustrates these considerations.



$$\mathbf{K} = (18 \times 18) \begin{bmatrix} \mathbf{K}_M & & \\ & \mathbf{K}_B & \\ & & \mathbf{K}_{\theta_Z} \end{bmatrix}$$

\mathbf{K}_M is the local stiffness matrix of the constant strain plane stress element (6×6)

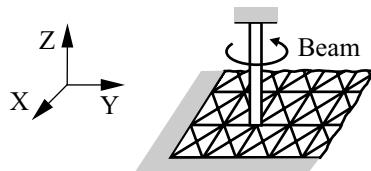
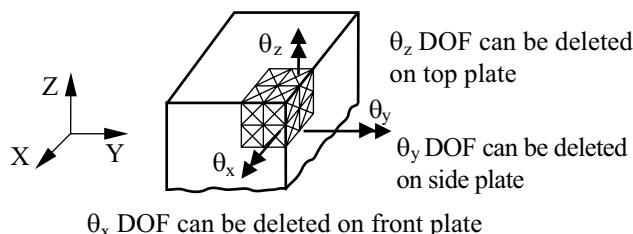
\mathbf{K}_B is the local stiffness matrix of the discrete Kirchhoff plate bending element (9×9)

\mathbf{K}_{θ_Z} is the artificial stiffness matrix for the rotation θ_Z (3×3)

Figure 2.6-1: Local node numbering; natural coordinate system; construction of total element stiffness matrix

- Considering the formulation of the element, the following observations should also be noted:

- ▶ The element does not model shear deformations and is only applicable to the analysis of thin plates and shells.
- ▶ The element does not contain any spurious zero energy mode.
- ▶ The element is applicable to the analysis of very thin plates and shells (it does not ever lock, see Section 2.7.10).
- ▶ Since the element is triangular, a mesh orientation effect can be noted in analyses when a mesh is used that is not spatially isotropic. Figs. 2.6-3 and 2.6-4 illustrate how meshes that are more appropriate can be constructed.



Note: Torque is taken by the beam only (except for the very small value resisted by the K_{θ_Z} stiffness of the plate).

Figure 2.6-2: Considerations for the zero stiffness corresponding to the rotational degree of freedom normal to the plate

- ▶ In the element formulation, stress resultants (membrane forces, bending moments) are employed, so that the integration points are only located on the midsurface of the element.

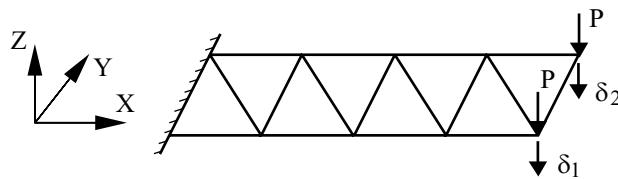
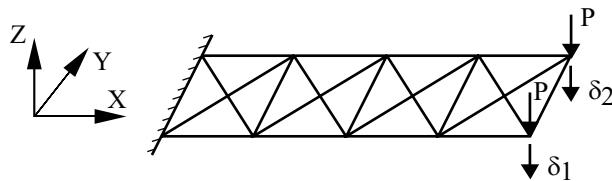
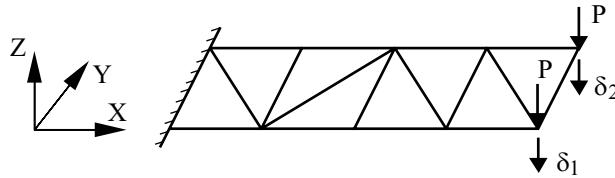
(a) Cantilever plate model, $\delta_1 \neq \delta_2$ (b) Cantilever plate model, $\delta_1 = \delta_2$ (c) Cantilever plate model, $\delta_1 \neq \delta_2$

Figure 2.6-3: Models of a cantilever indicating effect of discretization on the solution results

- Numerical integration is performed using the Hammer's formulas for triangles (see Fig. 2.6-5):
 - ▶ In elastic analysis, the membrane stiffness is evaluated in closed-form and the uncoupled bending stiffness is evaluated using a three-point (interior) scheme over which you have no control.
 - ▶ In elasto-plastic analyses, the membrane and bending stiffnesses are in general coupled. In this case both stiffnesses are integrated using numerical integration.

- The one-point integration shown in Fig. 2.6-5 corresponds to a reduced integration scheme and is therefore not recommended.
- Note that, in elastic analysis, the membrane forces are constant within the element.

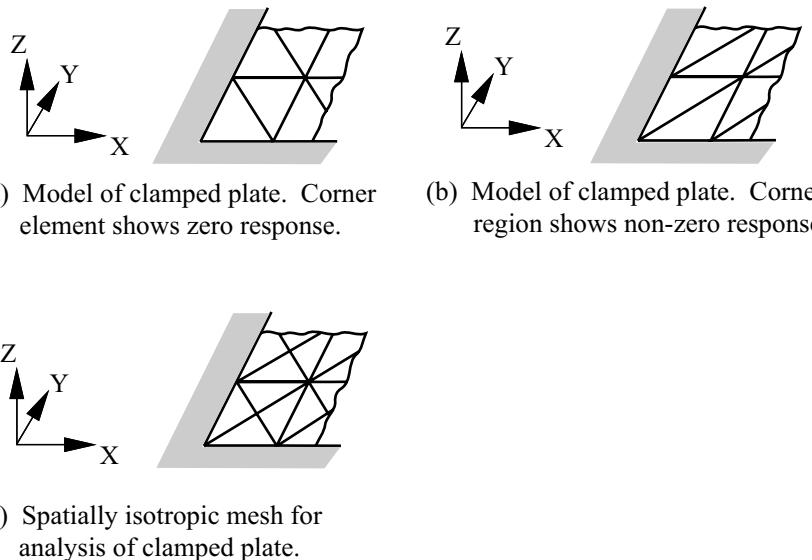


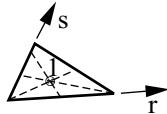
Figure 2.6-4: Models of a plate indicating effect of discretization on the solution results

2.6.2 Elasto-plastic analysis

- The elasto-plastic model for the plate/shell element in ADINA is based on the flow theory and the use of the Ilyushin yield condition. This yield condition operates directly on the stress resultants, i.e., the membrane forces and bending moments, and hence numerical integration is only performed over the midsurface of the element. The Ilyushin yield condition and its use with the element are described in Section 3.4.4.

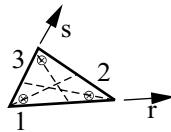
ref. K.J. Bathe, E. Dvorkin, and L.W. Ho, "Our Discrete-Kirchhoff and Isoparametric Shell Elements – An Assessment", *J. Computers and Structures*, Vol. 16, pp. 89-98, 1983.

One integration point (centroid):



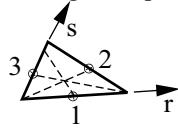
	r	s
1	0.33333	0.33333

Three integration points (interior):



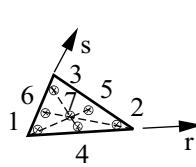
	r	s
1	0.16667	0.16667
2	0.66667	0.16667
3	0.16667	0.66667

Three integration points (midside):



	r	s
1	0.50000	0.
2	0.50000	0.50000
3	0.	0.50000

Seven integration points (interior):



	r	s
1	0.10129	0.10129
2	0.79763	0.10129
3	0.10129	0.79763
4	0.47014	0.05972
5	0.47014	0.47014
6	0.05972	0.47014
7	0.33333	0.33333

Figure 2.6-5: Hammer's integration point locations

- In practice the three-point integration schemes (see Fig. 2.6-5) are most effective for elasto-plastic analysis. The seven-point scheme can also be used to provide more output locations for the stress resultants.

2.6.3 Element mass matrices

- The plate/shell element can be employed with a lumped or a

consistent mass matrix (which is actually only an approximate consistent mass matrix).

- The translational lumped mass assigned to each node is $\frac{M}{3}$ and

the rotational lumped mass is $\frac{M}{3} \cdot \frac{t^2}{6}$, in which t is the thickness of the plate element.

- The element consistent mass matrix is calculated using the same linear displacement functions for all translational degrees of freedom of the element. Since the actual bending displacements are of cubic order, the mass matrix is approximate (see ref. KJB, Section 4.2.4).
- The rotational lumped mass can be multiplied by a user-specified scalar ETA (except in explicit analysis).

2.6.4 Element output

- You can request that ADINA either print or save stress resultants or nodal point forces.

Stress resultants: Each element outputs, at its integration points, the following information to the porthole file, based on the material model. This information is accessible in the AUI using the given variable names.

Elastic-isotropic, elastic-orthotropic: MEMBRANE_FORCE-XL,
MEMBRANE_FORCE-YL, MEMBRANE_FORCE-XLYL,
BENDING_MOMENT-XL, BENDING_MOMENT-YL,
BENDING_MOMENT-XLYL, MEMBRANE_STRAIN-XL,
MEMBRANE_STRAIN-YL, MEMBRANE_STRAIN-XLYL,
CURVATURE-XL, CURVATURE-YL, CURVATURE-XLYL

Ilyushin: PLASTIC_FLAG, MEMBRANE_FORCE-XL,
MEMBRANE_FORCE-YL, MEMBRANE_FORCE-XLYL,
BENDING_MOMENT-XL, BENDING_MOMENT-YL,
BENDING_MOMENT-XLYL, MEMBRANE_STRAIN-XL,

MEMBRANE_STRAIN-YL, MEMBRANE_STRAIN-XLYL,
CURVATURE-XL, CURVATURE-YL, CURVATURE-XLYL

The indices XL and YL indicate that the results are output with respect to the element local coordinate system.

See Section 12.1.1 for the definitions of those variables that are not self-explanatory.

Nodal forces: Nodal point force vectors which correspond to the element membrane forces and bending moments can also be requested in ADINA.

The nodal forces are accessible in the AUI using the variable names NODAL_FORCE-X, NODAL_FORCE-Y, NODAL_FORCE-Z, NODAL_MOMENT-X, NODAL_MOMENT-Y, NODAL_MOMENT-Z.

2.7 Shell elements

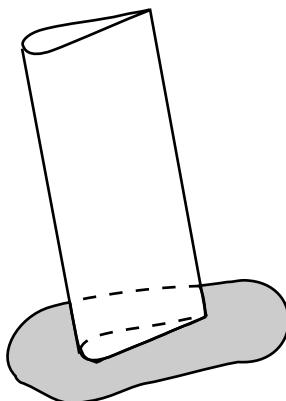
- The shell element is a 4- to 32-node (degenerate) isoparametric element that can be employed to model thick and thin general shell structures (see Fig. 2.7-1). However, depending on the application, the appropriate number of nodes on the element must be employed (see Fig. 2.7-2 and Section 2.7.10).

2.7.1 Basic assumptions in element formulation

*ref. KJB
Sections 5.4.2
and 6.5.2*

- The basic equations used in the formulation of the element are given in ref. KJB.
- The shell element is formulated treating the shell as a three-dimensional continuum with the following two assumptions used in the Timoshenko beam theory and the Reissner/Mindlin plate theory:

Analysis of a turbine blade:



Analysis of a shell roof:

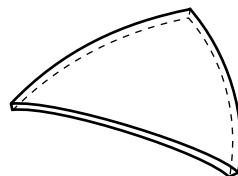


Figure 2.7-1: Some possible applications of shell elements

Assumption 1: Material particles that originally lie on a straight line "normal" to the midsurface of the structure remain on that straight line during the deformations.

Assumption 2: The stress in the direction normal to the midsurface of the structure is zero.

For the Timoshenko beam theory, the structure is the beam, and for the Reissner/Mindlin plate theory, the structure is the plate under consideration. In shell analysis, these assumptions correspond to a very general shell theory.

- In the calculations of the shell element matrices the following geometric quantities are used:
 - ▶ The coordinates of the node k that lies on the shell element midsurface at ${}^t x_k$, ${}^t y_k$, ${}^t z_k$ (see Fig. 2.7-3); (the left superscript denotes the configuration at time t)
 - ▶ The director vectors ${}^t \mathbf{V}_n^k$ pointing in the direction "normal" to the shell midsurface

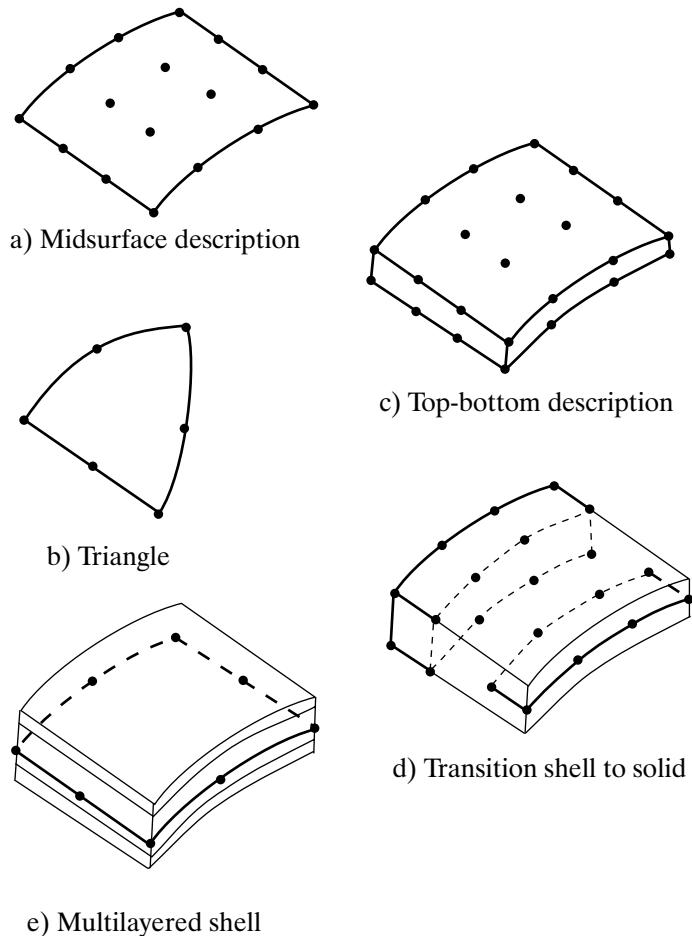


Figure 2.7-2: Examples of shell elements

ref. KJB
Fig. 5.33
page 437

- The shell thickness, a_k , at the nodal points measured in the direction of the director vectors $'\mathbf{V}_n^k$ (see Fig. 2.7-4 and Section 2.7.2).

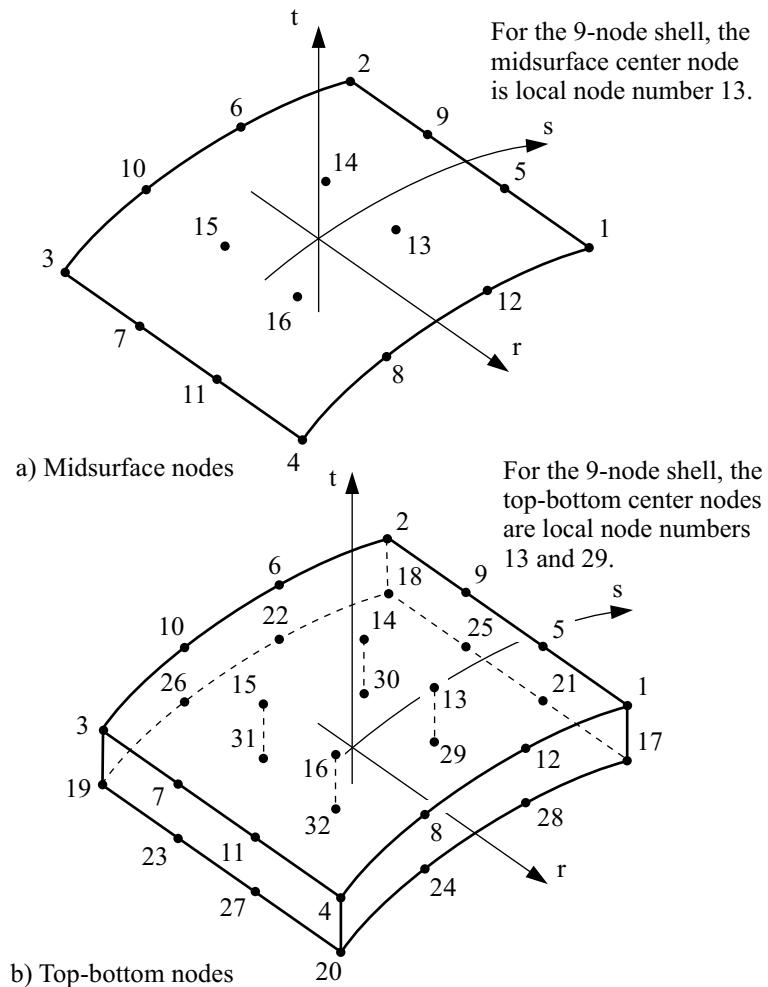


Figure 2.7-3: Some conventions for the shell element; local node numbering; natural coordinate system

- Fig. 2.7-4 shows a 4-node shell element with the shell midsurface nodal points and the nodal point director vectors. Using the midsurface nodal point coordinates, the shell midsurface is interpolated using the interpolation functions $h_k(r, s)$ given in ref. KJB. Similarly, using the vectors ${}^t \mathbf{V}_n$, the director vector ${}^t \mathbf{V}_n$ at any point P on the midsurface is obtained by interpolation using

the functions $h_k(r, s)$. Hence, with the shell thickness at nodal point k equal to a_k and the isoparametric coordinate t measured in the direction of $'\mathbf{V}_n$, the geometry of the shell at any time t is defined by

$$\begin{aligned} {}^t x &= \sum_{k=1}^q h_k {}^t x_k + \frac{t}{2} \sum_{k=1}^q a_k h_k {}^t V_{nx}^k \\ {}^t y &= \sum_{k=1}^q h_k {}^t y_k + \frac{t}{2} \sum_{k=1}^q a_k h_k {}^t V_{ny}^k \\ {}^t z &= \sum_{k=1}^q h_k {}^t z_k + \frac{t}{2} \sum_{k=1}^q a_k h_k {}^t V_{nz}^k \end{aligned}$$

shell midsurface interpolation		interpolation for material points not on the midsurface
-----------------------------------	--	--

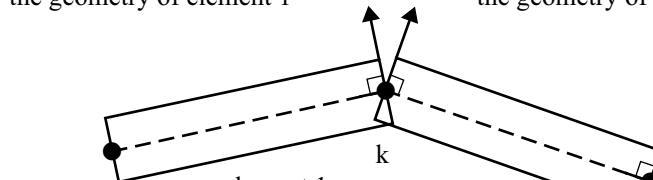
where q is the number of element nodes and ${}^t V_{nx}^k, {}^t V_{ny}^k, {}^t V_{nz}^k$ are the direction cosines of the shell director vector $'\mathbf{V}_n^k$.

- The direction vectors at nodes can be either directly input or automatically generated by the program. When they are generated by the program, they can be created as the normal vectors of a geometrical surface, or as the averaged vectors of the surrounding elements, which may not be exactly normal to the corresponding geometric surface.
- The assumption 1 on the kinematic behavior of the shell enters the finite element solution in that the particles along the director vector $'\mathbf{V}_n$ (interpolated from the nodal point director vectors $'\mathbf{V}_n^k$) remain on a straight line during deformation.

Note that in the finite element solution, the vector $'\mathbf{V}_n$ is not necessarily exactly normal to the shell midsurface. Figure 2.7-5(a) demonstrates this observation for a very simple case, considering the shell initial configuration. Furthermore, even if $'\mathbf{V}_n$ is originally normal to the shell midsurface, after deformations have

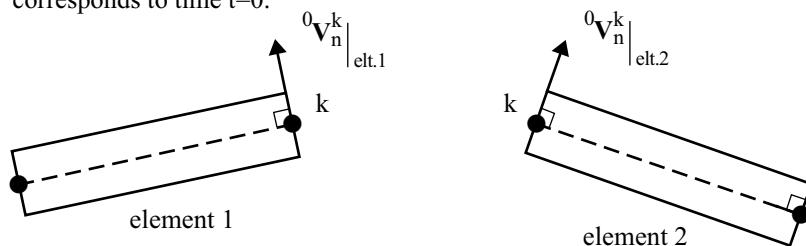
taken place this vector will in general not be exactly perpendicular to the midsurface because of shear deformations (see Fig. 2.7-5(b)).

Program-calculated midsurface normal vector at node k using the geometry of element 1



The configuration shown corresponds to time $t=0$.

Program-calculated midsurface normal vector at node k using the geometry of element 2



- (a) Program-calculated normal vectors (two such vectors) at node k. These vectors are used as director vectors for the respective elements. Node k has 6 DOFs.

Figure 2.7-4: Convention for shell element thickness

ref. KJB
Section 5.4.2
page 440

- The assumption 2 on the stress situation enters the finite element solution in a manner that is dependent on the formulation employed:

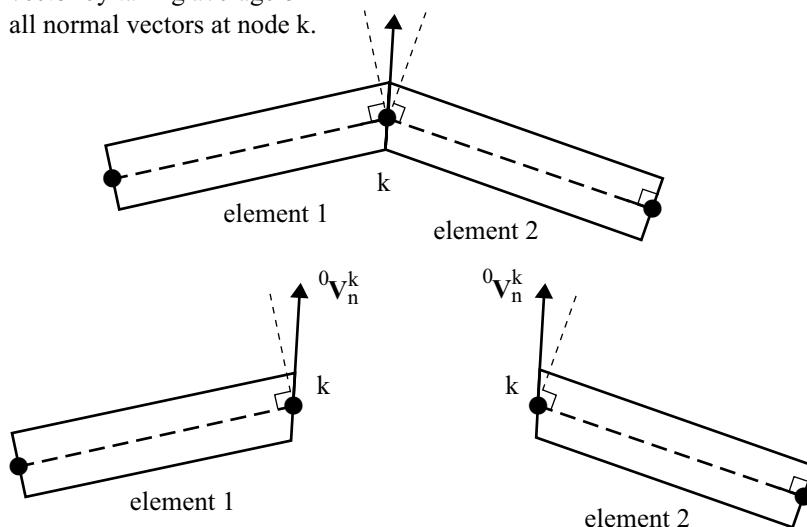
All formulations except for the large displacement/large strain ULH shell element: The stress in the t -direction (i.e., in the direction of $'\mathbf{V}_n'$) is imposed to be zero. This is achieved by using the stress-strain relationship in the \bar{r}, \bar{s}, t coordinate system, shown in Fig. 2.7-6(a), with the condition that the stress in the direction t is zero.

Large displacement/large strain ULH shell element: The stress in the \hat{t} -direction (not necessarily in the direction of $'\mathbf{V}_n'$ is

imposed to be zero. This is achieved by using the stress-strain relationship in the $\hat{r}, \hat{s}, \hat{t}$ coordinate system, shown in Fig. 2.7-6(b), with the condition that the stress in the direction \hat{t} is zero.

- Note that if the nodal point midsurface director vectors are input or generated (see Section 2.7.3) to be initially "exactly" normal to the midsurface of the shell element (the coordinates of this midsurface are obtained by interpolation from the nodal points on the midsurface), then assumption 1 is also exactly satisfied in the finite element solution, but assumption 2 is only approximately fulfilled in geometric nonlinear analysis, because the directions of the initially normal vectors are updated.

Program calculates director vector by taking average of all normal vectors at node k.



b) Program-calculated director vector (single vector)
at node k. Node k has 5 DOFs.

Figure 2.7-4: (continued)

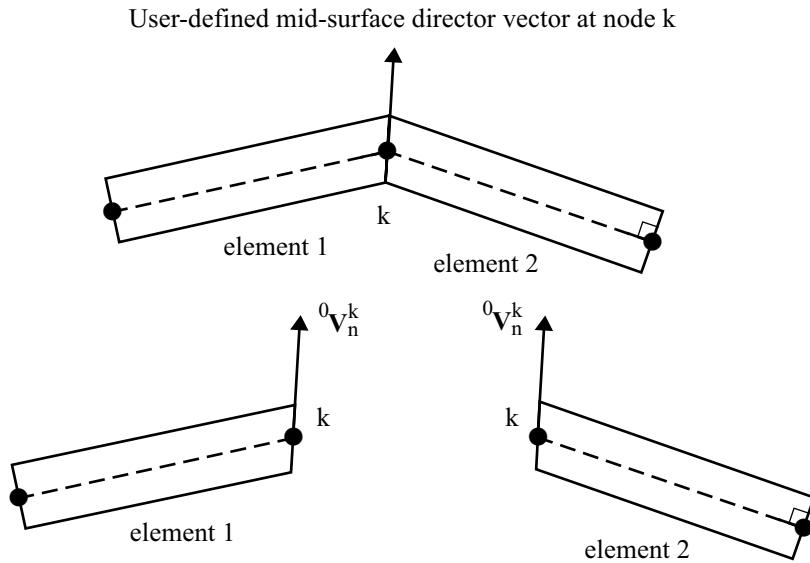
ref. KJB
pp. 399, 440

- The transverse shear deformations are assumed by default to be constant across the shell thickness. The use of the correction factor of 5/6 can be specified to improve the prediction of the displacement response.

- The interpolation of the geometry of the shell element is always as described above, but for a specific solution time the current coordinates of the midsurface nodal points are used, and the current director vectors are employed. The midsurface nodal point coordinates are updated by the translational displacements of the nodes and the director vectors are updated using the rotations at the nodes (rotation increments in large displacement analysis).
- Incompatible modes can be used in conjunction with the 4-node shell element. The theory used is analogous to the theory used for the 2-D solid element, see Section 2.2.1 and ref KJB, Section 4.4.1. The incompatible modes are added to the midsurface displacement interpolations; only the membrane action of the 4-node shell element is affected.

A typical use of the incompatible modes feature would be to improve the in-plane bending response of the 4-node element.

The incompatible modes feature is only available for 4-node single layer shell elements in which all nodes are on the midsurface of the element. The incompatible modes feature is available in linear and nonlinear analysis, for all formulations except for the ULJ formulation.

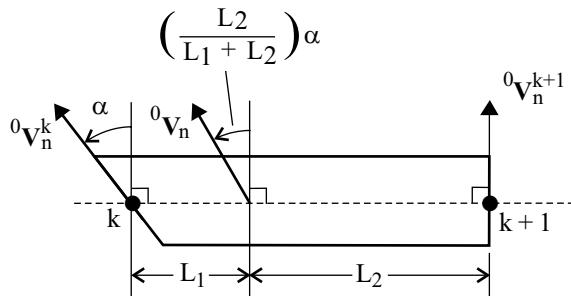


c) User input director vector (single vector)
at node k. Node k has 5 DOFs.

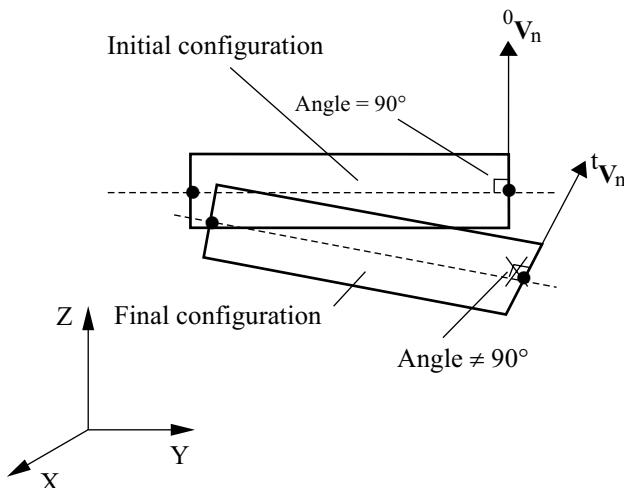
Figure 2.7-4: (continued)

2.7.2 Material models and formulations

- The shell element can be used with the following material models: **elastic-isotropic**, **elastic-orthotropic**, **plastic-bilinear**, **plastic-multilinear**, **plastic-orthotropic**, **thermo-isotropic**, **thermo-orthotropic**, **thermo-plastic**, **creep**, **plastic-creep**, **multilinear-plastic-creep**, **creep-variable**, **plastic-creep-variable**, **multilinear-plastic-creep-variable**, **viscoelastic**.
- The shell element can be used with a small displacement/small strain, large displacement/small strain, large displacement/large strain ULJ formulation or a large displacement/large strain ULH formulation. The small displacement/small strain and the large displacement/small strain formulations can be used with any of the above material models. The large displacement/large strain ULJ formulation can be used with the plastic-bilinear, plastic-multilinear or the plastic-orthotropic material models. The large displacement/large strain ULH formulation can be used with the plastic-bilinear or plastic-multilinear material models.



a) Interpolation of director vector inside element



b) Change in direction of director vector due to displacements and deformations (with shear)

Figure 2.7-5: Example of change in direction of director vector due to deformations. At time t the vector tV_n is not normal to the shell midsurface.

In the small displacement formulation/small strain formulation, the displacements and rotations are assumed to be infinitesimally small. In the large displacement formulation/small strain formulation, the displacements and rotations can be large, but the strains are assumed to be small. In the large displacement/large strain ULJ formulation, the total strains can be large, but the incremental strain for each time step should be small (< 1%). In the large displacement/large strain ULH formulation, the total strains and also the incremental strains can be large.

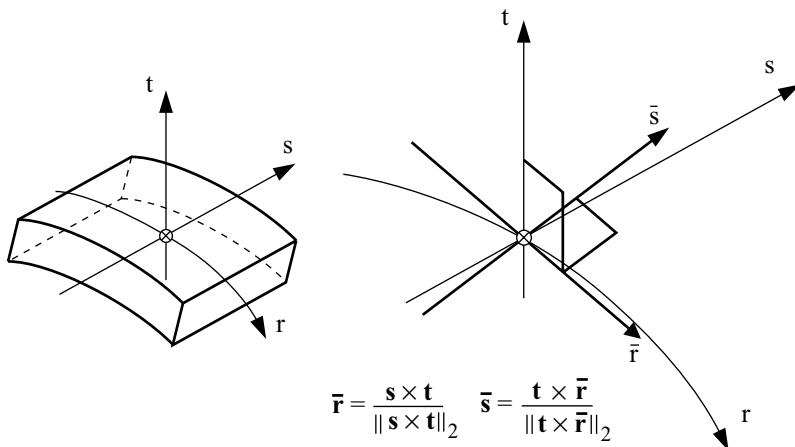


Figure 2.7-6a: Definition of the local Cartesian system (\bar{r} , \bar{s} , \bar{t}) at an integration point

The large displacement/large strain formulation can only be used in conjunction with 3-node, 4-node, 9-node or 16-node single layer shell elements, in which the shell geometry is described in terms of midsurface nodes.

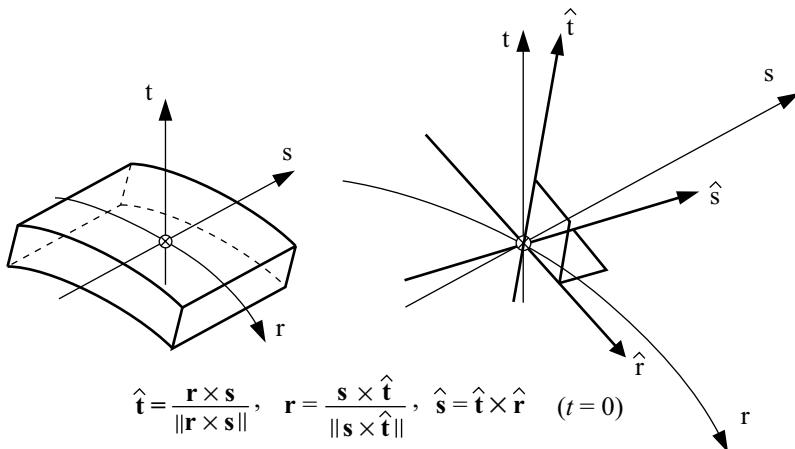


Figure 2.7-6b: Definition of the midsurface Cartesian system (\hat{r} , \hat{s} , \hat{t}) at an integration point

The use of a linear material with the small displacement/small strain formulation corresponds to a linear formulation, and the use of a nonlinear material with the small displacement/small strain formulation corresponds to a materially-nonlinear-only formulation.

- The effects of displacement-dependent pressure loading are taken into account in the large displacement formulation.

In frequency analysis, the stiffness matrix is optionally updated by deformation-dependent pressure loads acting onto shell elements. The update can improve the obtained frequencies and mode shapes when the shell structure is relatively flexible and deformation-dependent pressure loads are applied.

2.7.3 Shell nodal point degrees of freedom

- Either 5 or 6 degrees of freedom can be assigned at a shell element midsurface node.

5 degrees of freedom node: A node " k " that is assigned 5 degrees of freedom incorporates the following assumptions:

- ▶ The translations u_k , v_k , w_k are referred to the global Cartesian system (or to the skew system if such a system is defined at the node)
- ▶ Only one director vector (denoted at time = 0 as ${}^0\mathbf{V}_n^k$) is associated with the node. You can directly enter the director vector or let the program calculate it (see Fig. 2.7-4). In the latter case, the program calculates the director vector by taking the average of all normal vectors (one normal vector is generated per shell element attached to node k) at the node.

If two (or more) elements attached to the node have oppositely directed normals, ADINA reverses the oppositely directed normals, so that all normals attached to the node have (nearly) the same direction.

- ▶ The rotations α_k , β_k are referred to the local midsurface system (see Fig. 2.7-7) defined at time = 0 by

$${}^0\mathbf{V}_1^k = \frac{\mathbf{Y} \times {}^0\mathbf{V}_n^k}{\|\mathbf{Y} \times {}^0\mathbf{V}_n^k\|_2}$$

$${}^0\mathbf{V}_2^k = {}^0\mathbf{V}_n^k \times {}^0\mathbf{V}_1^k$$

For the special case when the ${}^0\mathbf{V}_n^k$ vector is parallel to the Y axis, the program uses the following conventions:

$${}^0\mathbf{V}_1^k \equiv \mathbf{Z} \quad {}^0\mathbf{V}_2^k \equiv \mathbf{X} \quad \text{when} \quad {}^0\mathbf{V}_n^k \equiv +\mathbf{Y}$$

and

$${}^0\mathbf{V}_1^k \equiv -\mathbf{Z} \quad {}^0\mathbf{V}_2^k \equiv \mathbf{X} \quad \text{when} \quad {}^0\mathbf{V}_n^k \equiv -\mathbf{Y}$$

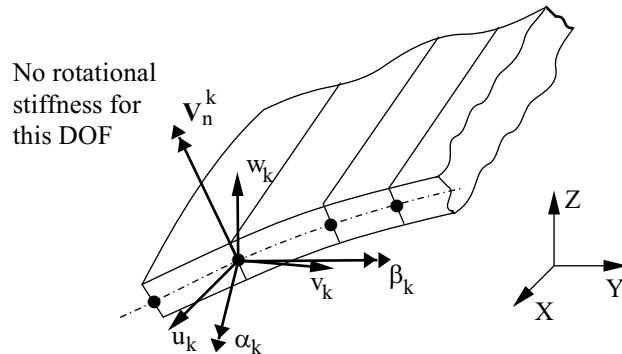


Figure 2.7-7: Shell degrees of freedom at node k

When using the large displacement formulation, the definitions of ${}^0\mathbf{V}_1^k$ and ${}^0\mathbf{V}_2^k$ are only used at time=0 (in the initial configuration) after which the vectors ${}^t\mathbf{V}_n^k$ and ${}^t\mathbf{V}_1^k$ are updated using incremental rotations at the nodal points, and ${}^t\mathbf{V}_2^k$ is calculated by the cross-product ${}^t\mathbf{V}_2^k = {}^t\mathbf{V}_n^k \times {}^t\mathbf{V}_1^k$.

- The rotational degree of freedom along ${}^0\mathbf{V}_n^k$ is automatically deleted by ADINA.

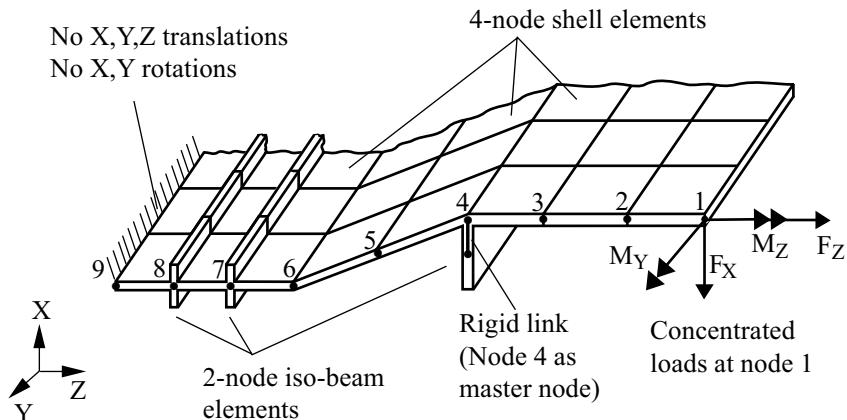
6 degrees of freedom node: A node "k" that is assigned 6 degrees of freedom incorporates the following assumptions:

- ▶ The translations u_k , v_k , w_k are referred to the global Cartesian system or any skew system assigned to node k .
 - ▶ A director vector would in general not be input for this node. (Note that any director vector input is ignored for a shell node with six degrees of freedom.)
 - ▶ The program generates as many normal vectors at node k as there are shell elements attached to the node (see Fig. 2.7-4(a)). Hence each individual shell element establishes at node k a vector normal to its midsurface. The components of the shell element matrices corresponding to the rotational degrees of freedom at this node are first formulated in the local midsurface system defined by the normal vector and then rotated to the global Cartesian system (or to the skew system, if defined at the node).
 - ▶ The three rotational degrees of freedom at node k referred to the global Cartesian system (or to the skew system, if defined at the node) can be free or be deleted.
- Some modeling recommendations are given below for the use of the shell elements:
 - ▶ Always let all director vectors be established by the program; director vectors should only be defined via input if specific director vectors are required in the modeling.
 - ▶ Always specify 5 degrees of freedom at all shell midsurface nodes except for the following cases in which 6 degrees of freedom should be used:
 - (i) shell elements intersecting at an angle,
 - (ii) coupling of shell elements with other types of structural elements such as isoparametric beams (e.g., in the modeling of stiffened shells),
 - (iii) coupling of rigid links (see Section 5.11.2) to the shell midsurface nodes

- (iv) imposing specific boundary conditions.

All of the above considerations are illustrated in the schematic example shown in Fig. 2.7-8. Note that in the example, the use of 6 degrees of freedom at node 1 together with the deletion of the X-rotation degree of freedom is only applicable in small displacement analysis.

If a large displacement formulation is used in this example, node 1 must be assigned 5 degrees of freedom and the applied concentrated moments must then refer to the local midsurface system at that node, which consists of the global directions for u_k , v_k , w_k and the local directions ' \mathbf{V}_1^k ' and ' \mathbf{V}_2^k ' for the incremental rotations α_k and β_k . (Alternatively, 6 degrees of freedom could be used if a beam of very small stiffness is defined along the edge of the plate.)



Node	Number of DOF	Degree of freedom						Reference system
		1	2	3	4	5	6	
1	6	✓	✓	✓	—	✓	✓	Global
2	5	✓	✓	✓	✓	✓	—*	Midsurface
3	5	✓	✓	✓	✓	✓	—*	Midsurface
4	6	✓	✓	✓	✓	✓	✓	Global
5	5	✓	✓	✓	✓	✓	—*	Midsurface
6	6	✓	✓	✓	✓	✓	✓	Global
7	6	✓	✓	✓	✓	✓	✓	Global
8	6	✓	✓	✓	✓	✓	✓	Global
9	6	—	—	—	—	—	✓	Global

— = Deleted because of no stiffness at this DOF

✓ = Free DOF

* = The AUI automatically deletes the 6th DOF
when a midsurface system is specified for the node

Figure 2.7-8: Example on the recommended use of shell elements

2.7.4 Transition elements

- The shell elements can also be employed as transition elements. A transition element is obtained by using, instead of a shell midsurface node, one node on the top surface and one node on the bottom surface of the shell element. Each of these nodes has then only 3 degrees of freedom corresponding to translations in the global X, Y, Z coordinate directions. Fig. 2.7-9(a) shows a cubic transition element.

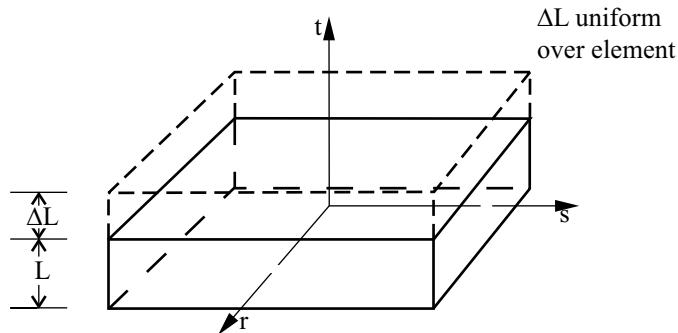
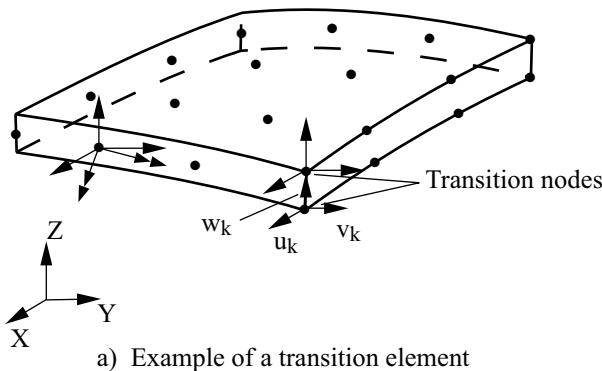


Figure 2.7-9: Transition elements

- Note that although the degrees of freedom at a shell transition node are those of a three-dimensional isoparametric element, the assumption that the stress "normal" to the midsurface of the shell (in the t -direction) is zero distinguishes a transition element from a 3-D solid element even when a shell element has only transition nodes.
- In particular, if only transition nodes are used on the element, then the element has a zero energy mode corresponding to a uniform strain in the t -direction. Fig. 2.7-9(b) shows this zero-energy mode. For a 3-D solid element, this mode corresponds, of course, to a uniform strain in the t -direction.

If not all nodes on a transition element are transition nodes, i.e.,

there is at least one midsurface node, then the zero-energy mode shown in Fig. 2.7-9(b) is not present.

- The transition element is particularly useful in modeling shell to shell and shell to solid intersections, see Fig. 2.7-10 and the following reference:

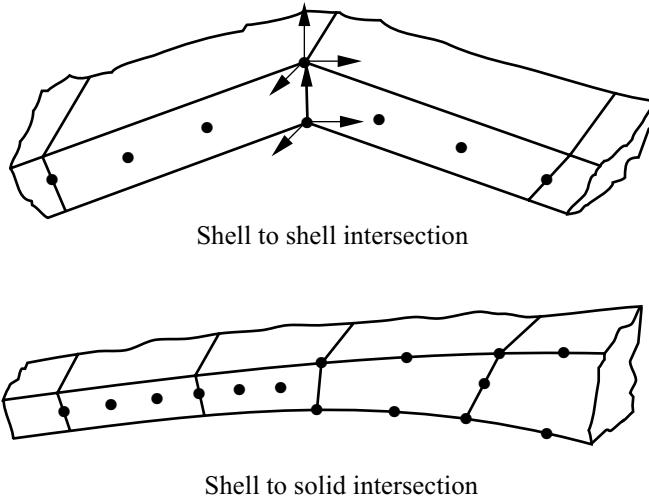


Figure 2.7-10: Examples of use of transition element

ref. K.J. Bathe and L.W. Ho, "Some Results in the Analysis of Thin Shell Structures," in *Nonlinear Finite Element Analysis in Structural Mechanics*, W. Wunderlich et al (eds.), Springer-Verlag, 1981.

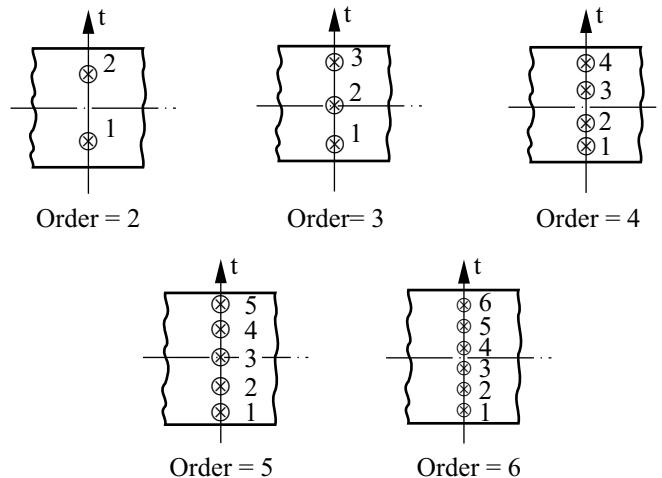
2.7.5 Numerical integration

ref. KJB
Section 5.5

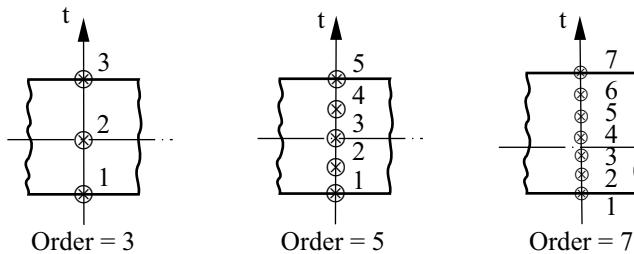
- Numerical integration is used for the evaluation of the element matrices, and the default integration or a higher order should always be used; then no spurious zero energy modes are present.
- Gauss numerical integration is used in the r-s plane. For the regular 4-node element, the default order is 2×2 integration. For the regular 8-node and 16-node elements, the default integrations are respectively 3×3 and 4×4 point integration.

*ref. KJB
Section 6.8.4*

- Either Gauss or Newton-Cotes numerical integration is used through the shell thickness. Usually, 2-point Gauss or 3-point Newton-Cotes integration is appropriate for an elastic material, but a higher integration order may be more effective for elastic-plastic analysis (see Fig. 2.7-11).



a) Gauss integration in the thickness direction

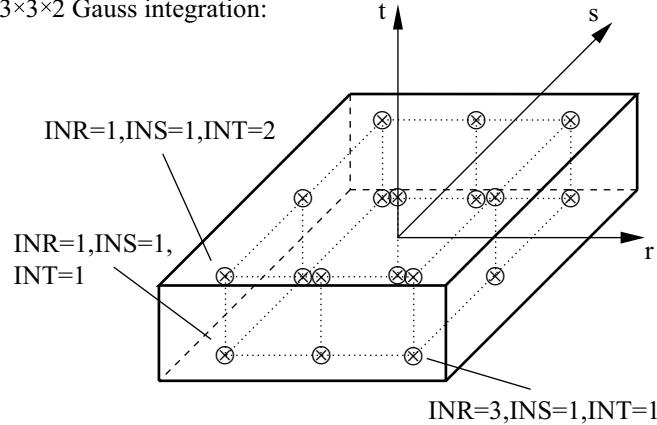


b) Newton-Cotes integration in the thickness direction

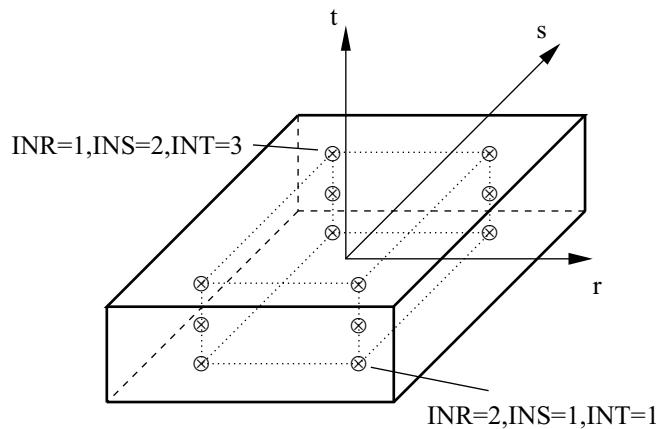
Figure 2.7-11: Numerical integration through the shell element thickness

- The labeling of the integration points for quadrilateral and triangular elements is given in Fig. 2.7-12. The locations of the integration points in the r-s plane are the same as for the 2-D solid elements (see Section 2.2.3).

$3 \times 3 \times 2$ Gauss integration:



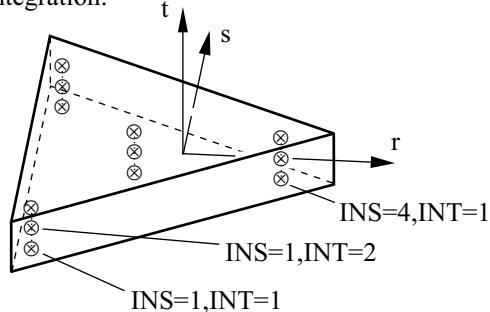
$2 \times 2 \times 3$ Newton-Cotes integration:



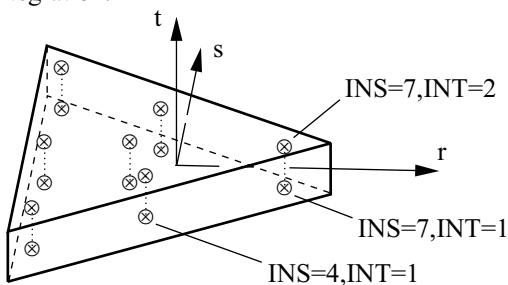
a) Examples of integration point labeling for quadrilateral shell elements

Figure 2.7-12: Examples of integration point labeling

4×3 integration:



7×2 integration:



b) Examples of integration point labeling for triangular shell elements

Figure 2.7-12: (continued)

2.7.6 Composite shell elements

- The composite shell elements are kinematically formulated in the same way as the single layer shell elements, but
 - ▶ An arbitrary number of layers can be used to make up the total thickness of the shell.
 - ▶ Each layer can be assigned one of the different material models available. The element is nonlinear if any of the material models is nonlinear, or if the large displacement formulation is used.
 - ▶ The computation of accurate transverse shear stresses based on a three-dimensional theory can be requested. This way the

condition of zero transverse shear stresses at the top and bottom surfaces of the shell are satisfied, and the transverse shear stresses are continuous at layer interfaces.

- Layers are numbered in sequential order starting from 1 at the bottom of the shell.
- The layer thicknesses can be assigned using one of two general approaches:
 - ▶ Specify the total element thickness and the percentage of thickness for each layer.
 - ▶ Specify the data of each layer (or ply) in terms of the weight per unit surface of the fiber W_f , the density of the fiber ρ_f and the fiber volume fraction of the fiber-matrix compound ϕ_f .

The AUI then computes the thickness of each layer

automatically using the formula $h = \frac{W_f}{\rho_f \phi_f}$, in which it is

assumed that the ply is made of uniaxial fibers, so that the fiber thickness fraction perpendicular to the plate midsurface is proportional to the fiber volume fraction. The total thickness of the multilayered shell is then the sum of the ply thicknesses.

This approach is especially useful when the layers are fiber-matrix composites.

- In order to take into account the change of material properties from one layer to another, numerical integration of the mass and stiffness matrices is performed layer by layer using reduced natural coordinates through the thickness of the element (see Fig. 2.7-13 and 2.7-14). The relation between the element natural coordinate t and the reduced natural coordinate t^n of layer n is:

$$t = -1 + \frac{1}{a} \left[2 \left(\sum_{i=1}^n \ell^i \right) - \ell^n (1 - t^n) \right] \quad (2.7-1)$$

with

t = element natural coordinate through the thickness

t^n = layer n natural coordinate through the thickness

ℓ^i = thickness of layer i

a = total element thickness

a and ℓ^i are functions of r and s .

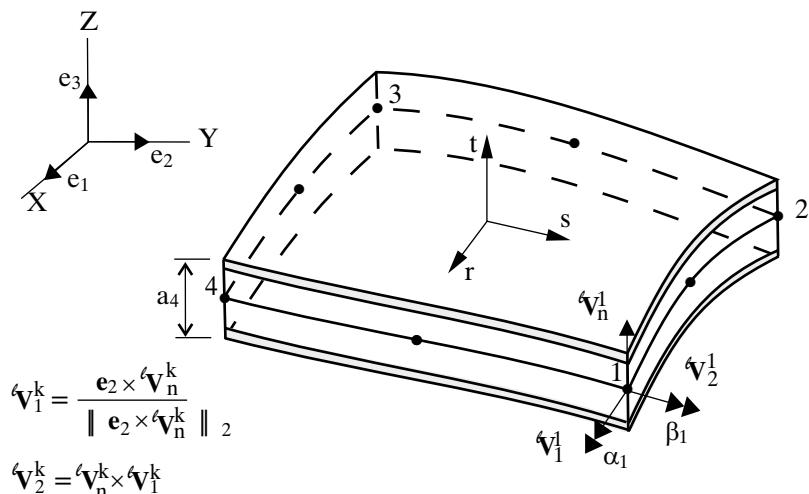


Figure 2.7-13: 8-node composite shell element

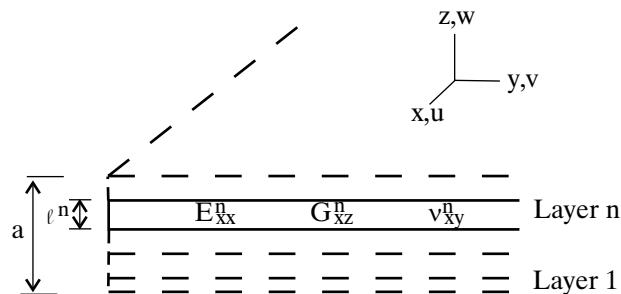


Figure 2.7-14: Multilayered plate

The geometry of layer n is given by:

$${}^{\ell}x_i = \sum_{k=1}^N h_k {}^{\ell}x_i^k + \sum_{k=1}^N \left[\left(-\frac{a_k}{2} + \sum_{i=1}^n \ell_k^i - \frac{\ell_k^n}{2} \right) + t^n \frac{\ell_k^n}{2} \right] h_k {}^{\ell}V_{ni}^k \quad (2.7-2)$$

with

${}^{\ell}x_i$ = coordinates of a point inside layer n

N = number of nodes

h_k = interpolation functions

${}^{\ell}x_i^k$ = Cartesian coordinates of node k

${}^{\ell}V_{ni}^k$ = components of normal vector ${}^{\ell}\mathbf{V}_n^k$

a_k = total element thickness at node k

ℓ_k^i = thickness of layer i at node k

$\ell_{[]}$ = 0 in the initial configuration, 1 in the deformed configuration

In the above formula, the expression

$$m_k^n = -\frac{a_k}{2} + \sum_{i=1}^n \ell_k^i - \frac{\ell_k^n}{2} \quad (2.7-3)$$

corresponds to the distance at node k between the element midsurface and the midsurface of layer n .

Accordingly, the displacements in layer n are:

$$u_i = \sum_{k=1}^N h_k u_i^k + \sum_{k=1}^N \left(m_k^n + \frac{t^n \ell_k^n}{2} \right) \left(-{}^0V_{2i}^k \alpha_k + {}^0V_{1i}^k \beta_k \right) \quad (2.7-4)$$

with

u_i^k = components of nodal displacements at node k

α_k, β_k = rotations of ${}^0\mathbf{V}_n^k$ about ${}^0\mathbf{V}_1^k$ and ${}^0\mathbf{V}_2^k$ (see Fig. 2.7-13)

Using the expressions of the coordinates and displacements

defined in Eqs. 2.7-2 and 2.7-4, the contribution of each layer to the element stiffness and mass matrices can be evaluated.

- In the analysis of thick plates and shells or in the analysis of multilayered structures such as composite sandwiches with low shear rigidity, the transverse shear deformation energy may not be negligible compared to the flexural energy and high transverse shear stresses may be encountered. In these cases, the shell kinematic formulation must be amended to obtain more accurate expressions of the shear deformation energy and transverse shear stresses.

By combining for a plate in cylindrical bending, the plate equilibrium equations and the 3-dimensional equilibrium equations, an expression of the shear stress which satisfies stress continuity at the layer interfaces and zero boundary values at the top and bottom surfaces of the shell can be obtained. Simultaneously, a shear reduction factor is obtained which can be used to improve the shear deformation energy approximation.

The expression of the shear factor is given by

$$k_{xz} = \frac{I_{xx}^2}{\int_{-\frac{a}{2}}^{\frac{a}{2}} G_{xz} dz \int_{-\frac{a}{2}}^{\frac{a}{2}} \frac{g_{xz}^2}{G_{xz}} dz} \quad (2.7-5)$$

and the transverse shear stress in terms of the shear strain value $\bar{\gamma}_{xz}$ is

$$\sigma_{xz}(z) = \bar{\gamma}_{xz} \frac{I_{xx} g_{xz}(z)}{\int_{-\frac{a}{2}}^{\frac{a}{2}} \frac{g_{xz}}{G_{xz}} dz} \quad (2.7-6)$$

where

$$I_{xx} = \int_{-\frac{a}{2}}^{\frac{a}{2}} D_{xx} (z - z_0)^2 dz$$

$$D_{xx}^n = \frac{E_{xx}^n}{1 - \nu_{xy}^n \nu_{yx}^n}$$

D_{xx} = D_{xx}^n in layer n

E_{xx} = Young's modulus

G_{xx} = shear modulus
 ν_{xy}^n, ν_{yx}^n = Poisson's ratios
 a = total thickness

$$g_{xx}(z) = - \int_{-\frac{a}{2}}^{\frac{a}{2}} D_{xx}(\xi - z_0) d\xi$$

Equations 2.7-5 and 2.7-6 are used in ADINA to calculate shear correction factors for each element and shear stress profiles at each integration point.

The shear correction factor calculation is only available for the linear elastic and linear orthotropic material models.

ref. O. Guillermin, M. Kojić, K.J. Bathe, "Linear and Nonlinear Analysis of Composite Shells," *Proceedings, STRUCOME 90, DATAID AS & I*, Paris, France, November 1990.

2.7.7 Mass matrices

- The shell element can be employed with a lumped or a consistent mass matrix, except for explicit dynamic analysis which always uses a lumped mass.
- The consistent mass matrix is calculated using the isoparametric formulation with the shell element interpolation functions.
- The lumped mass for translational degrees of freedom of midsurface nodes is M/n where M is the total element mass and n is the number of nodes. No special distributory concepts are employed to distinguish between corner and midside nodes, or to account for element distortion.

The rotational lumped mass for all except explicit dynamic analysis is, $\frac{M}{n} \cdot \frac{1}{12} (t_{av}^2)$ where t_{av} is the average shell thickness. The same rotational mass matrix is assumed for 5- and 6-degree of freedom nodes, and is applied to all rotational degrees of freedom.

The rotational lumped mass for explicit dynamic analysis is

$\frac{M}{n} \cdot \frac{1}{12} (t_{av}^2 + A^2)$, where t_{av} is the average shell thickness and A is the cross-sectional area. The rotational masses are scaled up to ensure that the rotational degrees of freedom do not reduce the critical time step for shell elements. The same rotational mass matrix is assumed for 5- and 6-degree of freedom nodes and is applied to all rotational degrees of freedom.

- The rotational lumped mass can be multiplied by a user-specified scalar ETA (except in explicit analysis).

2.7.8 Anisotropic failure criteria

- The study of material failure in composite structures has shown that various criteria can be used to assess failure. The following failure criteria are provided in ADINA for the analysis of shell structures using the elastic-isotropic, elastic-orthotropic, thermo-isotropic or thermo-orthotropic material models:

- ▶ Maximum stress failure criterion
- ▶ Maximum strain failure criterion
- ▶ Tsai-Hill failure criterion
- ▶ Tensor polynomial failure criterion
- ▶ Hashin failure criterion
- ▶ User-supplied failure criterion

- These criteria can be evaluated during the analysis to determine whether the material has failed. However, the stresses and material properties of the model are not changed as a consequence of this calculation. It is therefore intended only for use in linear elastic or thermo-elastic analysis, or as a stress state indicator in nonlinear analysis.
- The material failure is investigated at each integration point of each element during the analysis. The failure criteria values (and possibly modes) are printed and/or saved according to the ADINA stress printing and saving flags.
- The **maximum stress failure criterion** compares each component of the stress tensor (referred to the material principal

directions) to maximum stress values. The input constants for this criterion are:

- X_t = maximum tensile stress in the material first principal direction (a-dir.)
 X_c = maximum compressive stress in the material first principal direction
 Y_t = maximum tensile stress in the material second principal direction (b-dir.)
 Y_c = maximum compressive stress in the material second principal direction
 Z_t = maximum tensile stress in the material third principal direction (c-dir.)
 Z_c = maximum compressive stress in the material third principal direction
 S_{ab} = maximum absolute shear stress in the a-b plane
 S_{ac} = maximum absolute shear stress in the a-c plane
 S_{bc} = maximum absolute shear stress in the b-c plane

The failure of the material occurs when any of the following inequalities is not satisfied anymore:

$$\begin{aligned} X_c &< \tau_a &< X_t \\ Y_c &< \tau_b &< Y_t \\ Z_c &< \tau_c &< Z_t \\ -S_{ab} &< \tau_{ab} &< S_{ab} \\ -S_{ac} &< \tau_{ac} &< S_{ac} \\ -S_{bc} &< \tau_{bc} &< S_{bc} \end{aligned}$$

where $(\tau_a, \tau_b, \tau_c, \tau_{ab}, \tau_{ac}, \tau_{bc})$ is the stress vector referred to the principal material directions.

The above conditions are applicable to 3-D analysis. In plane stress analysis, the parameters Z_t , Z_c , S_{ac} and S_{bc} are not used.

- Note that the maximum stress criterion indicates the mode of failure. Note that this criterion does not take into account any interaction between failure modes.

- The **maximum strain failure criterion** compares each component of the strain tensor (referred to the material principal directions) to maximum strain values. The input constants for this criterion are:

X_{et}	=	maximum tensile strain in the material first principal direction (a-dir.)
X_{ec}	=	maximum compressive strain in the material first principal direction
Y_{et}	=	maximum tensile strain in the material second principal direction (b-dir.)
Y_{ec}	=	maximum compressive strain in the material second principal direction
Z_{et}	=	maximum tensile strain in the material third principal direction (c-dir.)
Z_{ec}	=	maximum compressive strain in the material third principal direction
S_{eab}	=	maximum absolute shear strain in the material a-b plane
S_{eac}	=	maximum absolute shear strain in the material a-c plane
S_{ebc}	=	maximum absolute shear strain in the material b-c plane

The failure of the material occurs when any one of the following inequalities is not satisfied anymore:

$$\begin{aligned} X_{ec} &< \varepsilon_a &< X_{et} \\ Y_{ec} &< \varepsilon_b &< Y_{et} \\ Z_{ec} &< \varepsilon_c &< Z_{et} \\ -S_{eab} &< \gamma_{ab} &< S_{eab} \\ -S_{eac} &< \gamma_{ac} &< S_{eac} \\ -S_{ebc} &< \gamma_{bc} &< S_{ebc} \end{aligned}$$

where $(\varepsilon_a, \varepsilon_b, \varepsilon_c, \gamma_{ab}, \gamma_{ac}, \gamma_{bc})$ is the strain vector referred to the principal material directions.

The above conditions are applicable to 3-D analysis. In plane stress analysis, however, the parameters Z_{et} , Z_{ec} , S_{eac} and S_{ebc} are not used.

- Note that the maximum strain criterion indicates the mode of failure. Note that this criterion does not take into account any interaction between failure modes.
- The **Tsai-Hill failure criterion** provides a yield criterion applicable to materials with anisotropic failure strengths. This is an extension of the von Mises yield criterion. Failure of the material occurs when the following inequality is not satisfied anymore:

$$(G+H)\tau_a^2 + (F+H)\tau_b^2 + (F+G)\tau_c^2 - 2H\tau_a\tau_b - 2G\tau_a\tau_c - 2F\tau_b\tau_c + 2L\tau_{bc}^2 + 2M\tau_{ac}^2 + 2N\tau_{ab}^2 < 1$$

with

$$F = \frac{1}{2} \left(-\frac{1}{X^2} + \frac{1}{Y^2} + \frac{1}{Z^2} \right)$$

$$G = \frac{1}{2} \left(\frac{1}{X^2} - \frac{1}{Y^2} + \frac{1}{Z^2} \right)$$

$$H = \frac{1}{2} \left(\frac{1}{X^2} + \frac{1}{Y^2} - \frac{1}{Z^2} \right)$$

$$L = \frac{1}{2 S_{bc}}$$

$$M = \frac{1}{2 S_{ac}}$$

$$N = \frac{1}{2 S_{ab}}$$

where

X = maximum absolute stress in the material first principal direction (a-dir.)

Y = maximum absolute stress in the material second principal direction (b-dir.)

Z = maximum absolute stress in the material third principal direction (c-dir.)

S_{ab} = maximum absolute shear stress in the a-b plane

S_{ac} = maximum absolute shear stress in the a-c plane

S_{bc} = maximum absolute shear stress in the b-c plane

In the case when plane stress conditions are used, the material behavior with respect to failure is considered as transversely isotropic ($Y=Z$), and the failure criterion reduces to

$$\frac{\tau_a^2}{X^2} - \frac{\tau_a \tau_b}{X^2} + \frac{\tau_b^2}{Y^2} + \frac{\tau_{ab}^2}{S_{ab}^2} < 1$$

In this case, the maximum allowable stresses Z , S_{bc} and S_{ac} are not used.

- Note that this criterion includes the effect of interactions between failure modes, but does not indicate a specific mode of failure.
- The **tensor polynomial failure criterion** assumes the existence of a failure surface in the stress space of the form:

$$F_i \tau_i + F_{ij} \tau_i \tau_j = 1 \quad \text{with } i, j = 1, \dots, 6$$

with the convention

$$\tau_1 = \tau_a ; \tau_2 = \tau_b ; \tau_3 = \tau_c$$

$$\tau_4 = \tau_{ab} ; \tau_5 = \tau_{ac} ; \tau_6 = \tau_{bc}$$

and with

$$F_1 = \frac{1}{X_{at}} + \frac{1}{X_{ac}} ; \quad F_{11} = -\frac{1}{X_{at} X_{ac}}$$

$$F_2 = \frac{1}{X_{bt}} + \frac{1}{X_{bc}} ; \quad F_{22} = -\frac{1}{X_{bt} X_{bc}}$$

$$F_3 = \frac{1}{X_{ct}} + \frac{1}{X_{cc}} ; \quad F_{33} = -\frac{1}{X_{ct} X_{cc}}$$

$$F_4, F_5, F_6 = 0$$

$$F_{44} = \frac{1}{S_{ab}} ; \quad F_{55} = \frac{1}{S_{ac}} ; \quad F_{66} = \frac{1}{S_{bc}}$$

where the following strength components need to be defined:

- X_{at} = maximum tensile stress in the material first principal direction (a-dir.)
- X_{ac} = maximum compressive stress in the material first principal direction
- X_{bt} = maximum tensile stress in the material second principal direction (b-dir.)
- X_{bc} = maximum compressive stress in the material second principal direction
- X_{ct} = maximum tensile stress in the material third principal direction (c-dir.)
- X_{cc} = maximum compressive stress in the material third principal direction
- S_{ab} = maximum absolute shear stress in the material a-b plane
- S_{ac} = maximum absolute shear stress in the material a-c plane
- S_{bc} = maximum absolute shear stress in the material b-c plane

For plane stress conditions, the parameters Z_t , Z_c , S_{bc} , S_{ac} , F_{bc} and F_{ac} are not used.

- The interaction strengths F_{ij} with $i \neq j$ can be input, or calculated by ADINA according to Hoffman's convention:

$$F_{ij} = -\left(\frac{1}{X_{it} X_{ic}} + \frac{1}{X_{jt} X_{jc}} \right)$$

with $i \neq j$ and $i, j = a, b, c$

- Note that this failure criterion includes interactions between the failure modes, but that the mode of failure is not given by the analysis.
- The **Hashin failure criterion** applies especially to fibrous composites where the fiber and the matrix failure mechanisms are distinct. In ADINA, we assume that the fibers are predominantly

aligned with the first material principal direction (a-direction), thus making the material transversely isotropic about the a-direction. Failure will occur when any of the following inequalities is not satisfied anymore:

$$\frac{\tau_a^2}{X_t^2} + \frac{(\tau_{ab}^2 + \tau_{ac}^2)}{S_{ab}^2} < 1 \quad (\text{if } \tau_a > 0) \quad (2.7-7)$$

$$\frac{\tau_a}{X_c} < 1 \quad (\text{if } \tau_a < 0) \quad (2.7-8)$$

$$\frac{(\tau_b + \tau_c)^2}{Y_t^2} + \frac{(\tau_{bc}^2 - \tau_b \tau_c)}{S_{tr}^2} + \frac{(\tau_{ab}^2 + \tau_{ac}^2)}{S_{ab}^2} < 1 \quad (\text{if } (\tau_b + \tau_c) > 0) \quad (2.7-9)$$

$$\frac{(\tau_b + \tau_c) \left[1 - \left(\frac{Y_c}{2S_{tr}} \right)^2 \right]}{Y_c} + \frac{(\tau_b + \tau_c)^2}{4S_{tr}^2} + \frac{(\tau_b^2 - \tau_b \tau_c)}{S_{tr}^2} + \frac{(\tau_{ab}^2 + \tau_{ac}^2)}{S_{ab}^2} < 1 \\ (\text{if } (\tau_b + \tau_c) < 0) \quad (2.7-10)$$

with

X_t = maximum tensile stress in the material first principal direction (a-dir.)

X_c = maximum compressive stress in the material first principal direction

Y_t = maximum tensile stress in the material second principal direction (b-dir.)

Y_c = maximum compressive stress in the material second principal direction

S_{ab} = maximum absolute shear stress in the a-b plane (note that $S_{ab} = S_{ac}$)

S_{tr} = maximum absolute shear stress in the b-c plane

The above inequalities are usually referred to as: tensile fiber failure (2.7-7), compressive fiber failure (2.7-8), tensile matrix failure (2.7-9), compressive matrix failure (2.7-10).

- The **user-supplied failure criterion** allows for the description of a failure envelope using up to four quadratic failure relations of the form:

$$F_i \tau_i + F_{ij} \tau_i \tau_j = 1 \quad \text{if} \quad \sum_{i=1}^6 \alpha_i \tau_i > 0$$

with

$$\tau_1 = \tau_a ; \tau_2 = \tau_b ; \tau_3 = \tau_c$$

$$\tau_4 = \tau_{bc} ; \tau_5 = \tau_{ac} ; \tau_6 = \tau_{ab}$$

The material failure occurs when any one of the failure surfaces is reached and the corresponding stress condition is satisfied at the same time.

- This flexible and very general user-supplied failure model allows you to input any quadratic failure criteria with failure mode interactions.
- The following references contain additional information about composite failure criteria:

ref. Jones R.M., *Mechanics of Composite Materials*, McGraw Hill, 1975.

ref. Tsai S.W., "A Survey of Macroscopic Failure Criteria for Composite Materials," *Journal of Reinforced Plastics and Composites*, Vol.3, pp.40-62, January 1984.

ref. Hashin Z., "Analysis of Composite Materials – A Survey," *Journal of Applied Mechanics*, Transactions of the ASME, Vol. 50, pp. 481-505, September 1983.

2.7.9 Element output

You can request that ADINA either print and save stresses or forces.

Stresses: Each element outputs, at its integration points, the following information to the porthole file, based on the material

model. This information is accessible in the AUI using the given variable names.

The results can be requested to be referred to the global (X,Y,Z) axes, or to the local (\bar{r}, \bar{s}, t) axes (see Fig. 2.7-6a), or to the midsurface $(\hat{r}, \hat{s}, \hat{t})$ axes (see Fig. 2.7-6b). For the large displacement/large strain ULH formulation, both the local and midsurface axes are the $(\hat{r}, \hat{s}, \hat{t})$ axes, so the results in both the local and midsurface systems are identical. In addition, for orthotropic material models, the results referred to the material axes can be requested.

Elastic-isotropic, elastic-orthotropic: STRESS (XYZ) ,
STRAIN (XYZ) , FE_EFFECTIVE_STRESS

Elastic-isotropic with thermal effects: STRESS (XYZ) ,
STRAIN (XYZ) , FE_EFFECTIVE_STRESS,
THERMAL_STRAIN, ELEMENT_TEMPERATURE

Thermo-isotropic: STRESS (XYZ) , STRAIN (XYZ) ,
THERMAL_STRAIN (XYZ) , FE_EFFECTIVE_STRESS,
ELEMENT_TEMPERATURE

Plastic-bilinear, plastic-multilinear, without thermal effects:
PLASTIC_FLAG, STRESS (XYZ) , STRAIN (XYZ) ,
PLASTIC_STRAIN (XYZ) , FE_EFFECTIVE_STRESS,
YIELD_STRESS, ACCUM_EFF_PLASTIC_STRAIN

Plastic-bilinear, plastic-multilinear with thermal effects:
PLASTIC_FLAG, STRESS (XYZ) , STRAIN (XYZ) ,
PLASTIC_STRAIN (XYZ) , FE_EFFECTIVE_STRESS,
YIELD_STRESS, ACCUM_EFF_PLASTIC_STRAIN,
THERMAL_STRAIN (XYZ) , ELEMENT_TEMPERATURE

Thermo-plastic, creep, plastic-creep, multilinear-plastic-creep,
creep-variable, plastic-creep-variable, multilinear-plastic-creep-
variable: PLASTIC_FLAG, NUMBER_OF_SUBINCREMENTS ,
STRESS (XYZ) , STRAIN (XYZ) ,
PLASTIC_STRAIN (XYZ) , CREEP_STRAIN (XYZ) ,
THERMAL_STRAIN (XYZ) , ELEMENT_TEMPERATURE ,

ACCUM_EFF_PLASTIC_STRAIN,
FE_EFFECTIVE_STRESS, YIELD_STRESS,
EFFECTIVE_CREEP_STRAIN

Plastic-orthotropic without thermal effects: PLASTIC_FLAG,
STRESS(XYZ), STRAIN(XYZ),
PLASTIC_STRAIN(XYZ), HILL_EFFECTIVE_STRESS,
YIELD_STRESS, ACCUM_EFF_PLASTIC_STRAIN

Plastic-orthotropic with thermal effects: PLASTIC_FLAG,
STRESS(XYZ), STRAIN(XYZ),
PLASTIC_STRAIN(XYZ), HILL_EFFECTIVE_STRESS,
YIELD_STRESS, ACCUM_EFF_PLASTIC_STRAIN,
THERMAL_STRAIN(XYZ), ELEMENT_TEMPERATURE

Viscoelastic: STRESS(XYZ), STRAIN(XYZ),
THERMAL_STRAIN(XYZ), ELEMENT_TEMPERATURE

In the above lists,

when results are referred to the global (X,Y,Z) axes,
STRESS(XYZ) = STRESS-XX, STRESS-YY,
STRESS-ZZ, STRESS-XY, STRESS-XZ,
STRESS-YZ

when results are referred to the (\bar{r}, \bar{s}, t) or $(\hat{r}, \hat{s}, \hat{t})$ axes,
STRESS(XYZ) = STRESS-RR, STRESS-SS,
STRESS-TT, STRESS-RS, STRESS-RT,
STRESS-ST

when results are referred to the material axes,
STRESS(XYZ) = STRESS-AA, STRESS-BB,
STRESS-CC, STRESS-AB, STRESS-AC,
STRESS-BC

with similar definitions for the other abbreviations used above.

See Section 12.1.1 for the definitions of those variables that are not self-explanatory.

- In addition, when a failure model is used, each element outputs, at each integration point, the following information to the porthole file, based on the failure criterion. This information is accessible in the AUI using the following variable names:

Maximum stress, maximum strain:

```
FAILURE_FLAG_TENSION-A,  
FAILURE_FLAG_COMPRESSION-A,  
FAILURE_FLAG_TENSION-B,  
FAILURE_FLAG_COMPRESSION-B,  
FAILURE_FLAG_TENSION-C,  
FAILURE_FLAG_COMPRESSION-C,  
FAILURE_FLAG_SHEAR-AB, FAILURE_FLAG_SHEAR-AC,  
FAILURE_FLAG_SHEAR-BC, FAILURE_FLAG
```

Tsai-Hill, tensor polynomial: FAILURE_FLAG,
FAILURE_CRITERION

Hashin: FAILURE_FLAG_TENS-FIBER,
FAILURE_FLAG_COMP-FIBER,
FAILURE_FLAG_TENS-MATRIX,
FAILURE_FLAG_COMP-MATRIX,
FAILURE_CRITERION_TENS-FIBER,
FAILURE_CRITERION_COMP-FIBER,
FAILURE_CRITERION_TENS-MATRIX,
FAILURE_CRITERION_COMP-MATRIX

User-supplied: FAILURE_FLAG_SURFACE-1,
FAILURE_FLAG_SURFACE-2,
FAILURE_FLAG_SURFACE-3,
FAILURE_FLAG_SURFACE-4, FAILURE_FLAG,
FAILURE_CRITERION_SURFACE-1,
FAILURE_CRITERION_SURFACE-2,
FAILURE_CRITERION_SURFACE-3,
FAILURE_CRITERION_SURFACE-4

See Section 12.1.1 for the definitions of those variables that are not self-explanatory.

Section results: The computation of stress resultants, forces and moments, the computation of membrane strains and curvatures, and the position of the neutral axes can be requested in ADINA. The latter is especially useful for multilayered shell elements.

These results are given at the locations corresponding to the projections of the integration points on the shell element midsurface (see Fig.2.7-15).

Fig. 2.7-16 shows the convention used for positive forces and moments. Table 2.7-1 contains the formulas used to evaluate the stress and strain resultants and the neutral axes positions. They are always referred to the local Cartesian system (\bar{r}, \bar{s}, t) (see Fig. 2.7-15).

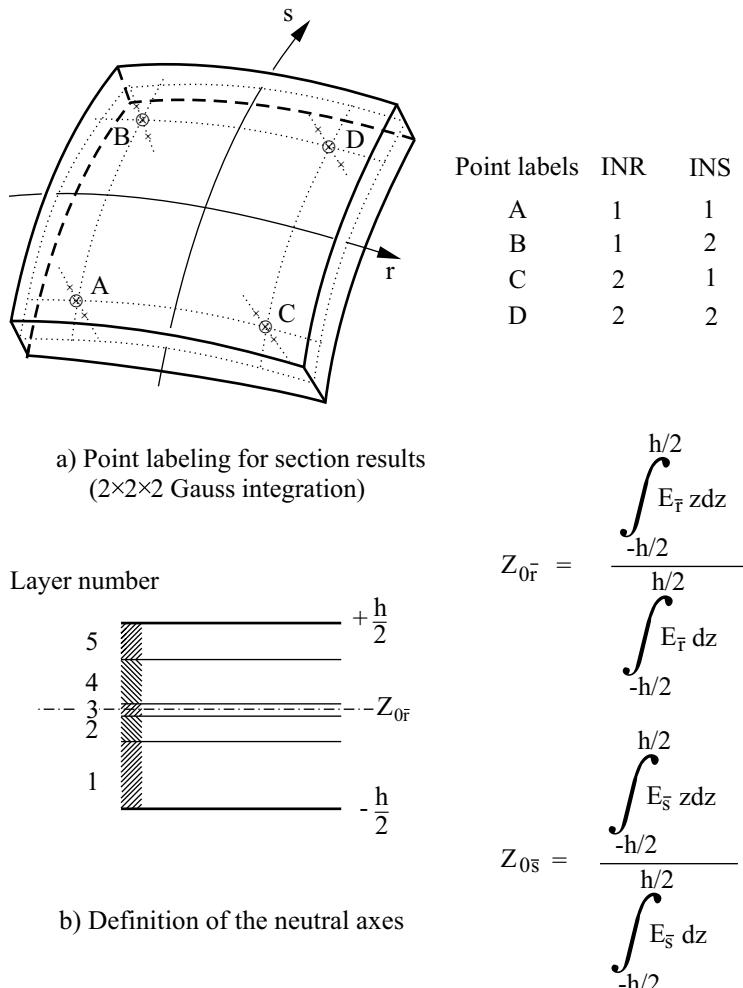


Figure 2.7-15: Conventions and definitions for section results output

Table 2.7-1: Section results output for the shell element

Stress resultants (forces and moments)	
$R_{\bar{r}\bar{r}} = \int_{-h/2}^{h/2} \sigma_{\bar{r}\bar{r}} dz$	$M_{\bar{r}\bar{r}} = \int_{-h/2}^{h/2} \sigma_{\bar{s}\bar{s}} z dz$
$R_{\bar{s}\bar{s}} = \int_{-h/2}^{h/2} \sigma_{\bar{s}\bar{s}} dz$	$M_{\bar{s}\bar{s}} = - \int_{-h/2}^{h/2} \sigma_{\bar{r}\bar{r}} z dz$
$R_{tt} = \int_{-h/2}^{h/2} \sigma_{tt} dz$	
$R_{\bar{r}\bar{s}} = \int_{-h/2}^{h/2} \sigma_{\bar{r}\bar{s}} dz$	$M_{\bar{r}\bar{s}} = \int_{-h/2}^{h/2} \sigma_{\bar{r}\bar{s}} z dz$
$R_{\bar{r}t} = \int_{-h/2}^{h/2} \sigma_{\bar{r}t} dz$	
$R_{\bar{s}t} = \int_{-h/2}^{h/2} \sigma_{\bar{s}t} dz$	
Membrane strains and curvatures	
$\epsilon_{m\bar{r}\bar{r}} = \epsilon_{I\bar{r}\bar{r}} - z_1 \frac{\epsilon_{2\bar{r}\bar{r}} - \epsilon_{I\bar{r}\bar{r}}}{(z_2 - z_1)}$	$\chi_{\bar{s}\bar{s}} = \frac{\epsilon_{2\bar{r}\bar{r}} - \epsilon_{I\bar{r}\bar{r}}}{(z_2 - z_1)}$
$\epsilon_{m\bar{s}\bar{s}} = \epsilon_{I\bar{s}\bar{s}} - z_1 \frac{\epsilon_{2\bar{s}\bar{s}} - \epsilon_{I\bar{s}\bar{s}}}{(z_2 - z_1)}$	$\chi_{\bar{r}\bar{r}} = \frac{\epsilon_{2\bar{s}\bar{s}} - \epsilon_{I\bar{s}\bar{s}}}{(z_2 - z_1)}$
$\epsilon_{m\bar{r}\bar{s}} = \epsilon_{I\bar{r}\bar{s}} - z_1 \frac{\epsilon_{2\bar{r}\bar{s}} - \epsilon_{I\bar{r}\bar{s}}}{(z_2 - z_1)}$	$\chi_{\bar{r}\bar{s}} = \frac{\epsilon_{2\bar{r}\bar{s}} - \epsilon_{I\bar{r}\bar{s}}}{(z_2 - z_1)}$

Indices 1 and 2 refer respectively to the lowest and highest integration points on a normal to the element midsurface.

Note that the section results are given with respect to the local Cartesian system of the shell elements.

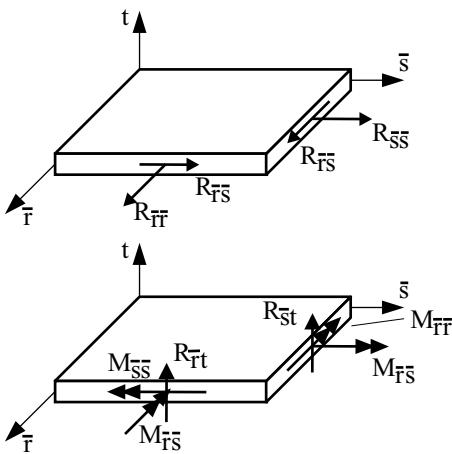


Figure 2.7-16: Nomenclature for shell stress resultants (positive)

The corresponding AUI variables are (for all material models):
 MEMBRANE_FORCE-RB, MEMBRANE_FORCE-SB,
 MEMBRANE_FORCE-RBSB, SHEAR_FORCE-RB,
 SHEAR_FORCE-SB, BENDING_MOMENT-RB,
 BENDING_MOMENT-SB, BENDING_MOMENT-RBSB,
 MEMBRANE_STRAIN-RB, MEMBRANE_STRAIN-SB,
 MEMBRANE_STRAIN-RBSB, CURVATURE-RB,
 CURVATURE-SB, CURVATURE-RBSB,
 SHEAR_STRAIN-RB, SHEAR_STRAIN-SB,
 NEUTRAL_AXIS_POSITION-RB,
 NEUTRAL_AXIS_POSITION-SB.

See Section 12.1.1 for the definitions of those variables that are not self-explanatory.

*ref. KJB
Example 5.11
pp. 358-359*

Nodal forces: The nodal point force vector which corresponds to the element internal stresses can also be requested in ADINA. The procedure for the calculation of the force vector is given in ref. KJB.

The nodal forces are accessible in the AUI using the variable names NODAL_FORCE-X, NODAL_FORCE-Y, NODAL_FORCE-Z, NODAL_MOMENT-X, NODAL_MOMENT-Y, NODAL_MOMENT-Z.

2.7.10 Selection of elements for analysis of thin and thick shells

- The following types of shell elements are available:

	Number of nodes	Single layer with mid-surface nodes only	Multilayer or top & bottom
Quadrilateral elements	4-node	MITC4	MITC4
	8-node	MITC8	MITC8
	9-node	MITC9	DISP9
	16-node	MITC16	DISP16
Triangular elements	3-node	MITC3	MITC4 Collapsed
	6-node	DISP7 Collapsed	DISP7 Collapsed
	9-node	DISP10 Collapsed	DISP10 Collapsed

The types of triangular elements are illustrated in Fig. 2.7-17.

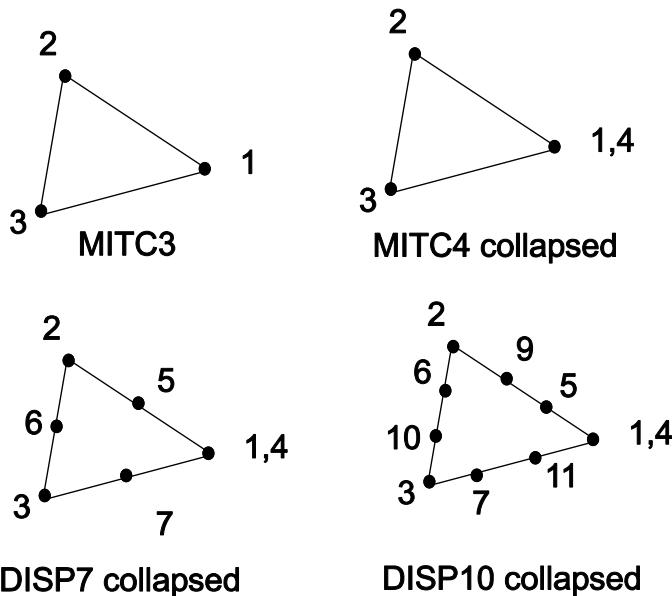


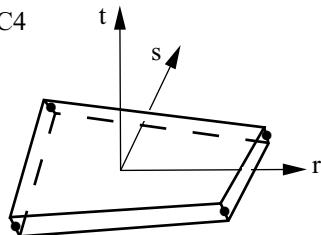
Figure 2.7-17. Types of triangular elements

- The most effective element for analysis of general shells is usually the 4-node element, shown in Fig. 2.7-18(a). This element

does not lock and has a high predictive capability and hence can be used for effective analysis of thin and thick shells. See references:

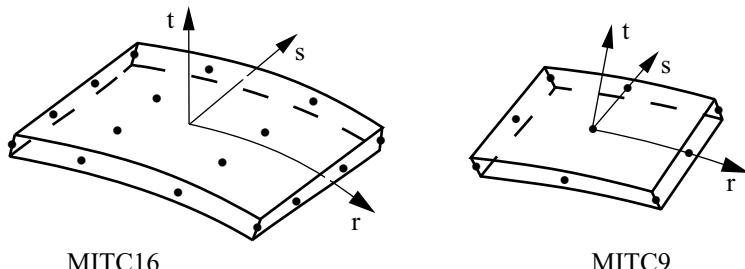
- ref. Dvorkin, E. and Bathe, K.J., "A Continuum Mechanics Based Four-Node Shell Element for General Nonlinear Analysis," *Engineering Computations*, Vol. 1, pp. 77-88, 1984.

Recommended element: MITC4



(a) 4-node shell element for thick and thin shells.

Sometimes also useful elements: MITC16 and MITC9



MITC16

MITC9

(b) 16-node cubic shell element for thick and thin shells.
9-node shell element for thick and thin shells.

Figure 2.7-18: Recommended elements for analysis of any shell

- ref. Bathe, K.J. and Dvorkin, E., "A Four-Node Plate Bending Element Based on Reissner/Mindlin Plate Theory and a Mixed Interpolation," *Int. J. Num. Meth. in Eng.*, Vol. 21, pp. 367-383, 1985.

-
- ref. Bathe, K.J. and Dvorkin, E., "A Formulation of General Shell Elements – The Use of Mixed Interpolation of Tensorial Components," *Int. J. Num. Meth. in Eng.*, Vol. 22 pp. 697-722, 1986.
- ref. Bucalem, M.L and Bathe, K.J., "Higher-Order MITC General Shell Elements," *Int. J. for Num. Meth in Engng.*, Vol. 36, pp. 3729-3754, 1993.
- ref. Bathe, K.J., Iosilevich, A. and Chapelle, D., "An Evaluation of the MITC Shell Elements," *Comp. Struct.*, Vol. 75, pp.1-30, 2000.
- ref. Lee, P.H. and Bathe, K.J., "Development of MITC isotropic triangular shell finite elements", *Comp. Struct.*, Vol.82, 945-962.

In linear analysis it can be effective to use the 9-node shell element (the MITC9 element) or the 16-node shell element (the MITC16 element) (see Fig. 2.7-18(b)). However, the use of these elements can be costly.

*ref. KJB
pp. 403-408*

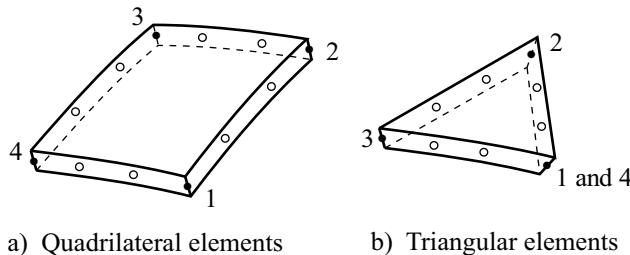
- The phenomenon of an element being much too stiff is in the literature referred to as element locking. In essence, the phenomenon arises because the interpolation functions used for an element are not able to represent zero (or very small) shearing or membrane strains. If the element cannot represent zero shearing strains, but the physical situation corresponds to zero (or very small) shearing strains, then the element becomes very stiff as its thickness over length ratio decreases.

Our MITC3, MITC4, MITC9 and MITC16 elements are implemented to overcome the locking problem. When you choose a 4-node shell element, the MITC4 element is used for quadrilateral elements and MITC3 is used for triangular elements. When you choose a 3-, 9- or 16-node single layered shell element, the MITC3, MITC9 and MITC16 elements are used.

- In order to arrive at an appropriate element idealization of a thin shell, it may be effective to consider the behavior of a single element in modeling a typical part of the shell. As an example, if a shell of thickness h and principal radii of curvatures R_1 and R_2 is to

be analyzed, a single element of this thickness and these radii and supported as a cantilever could be subjected to different simple stress states. The behavior of the single element when subjected to the simple stress states (e.g., constant bending moments) tells what size of element, and hence element idealization, can be used to solve the actual shell problem.

- For the analysis of thick shells, the elements depicted in Fig. 2.7-19 can be used as well. If necessary, these elements can also be employed together with the elements of Fig. 2.7-18 in the analysis of a thin shell, but then only a few elements should be employed to model a special region, where necessary, such as a transition region, a cut-out, and a triangular region. Also, the elements should be small enough in size.



a) Quadrilateral elements b) Triangular elements

- Nodes 1 to 4 (corner nodes) must be input
- Optional nodes

Figure 2.7-19: Elements for analysis of thick shells

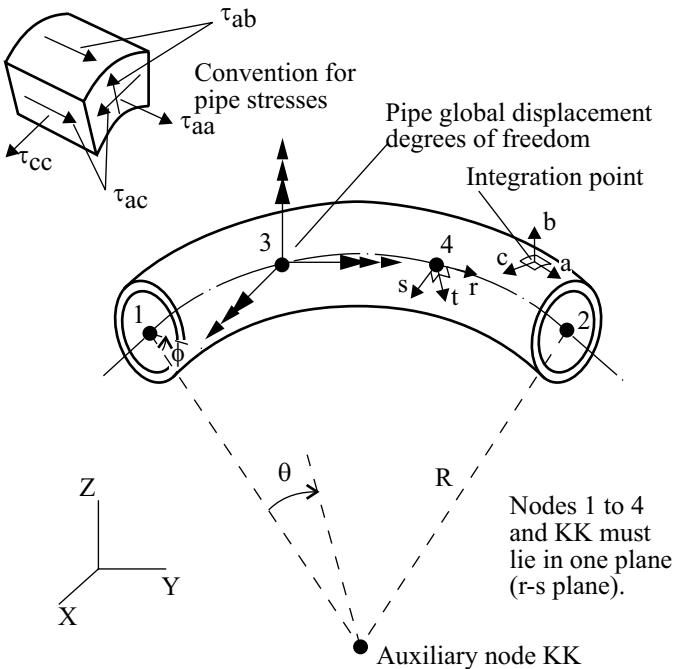
- It is recommended that, as often as possible, only the 3-node, 4-node, 9-node and 16-node elements be employed.
- Note that the elements in Fig. 2.7-19 can also be used as transition elements by assigning top and bottom surface nodes instead of a midsurface node.
- The shell elements can be used in conjunction with the bolt-shaft elements (Section 2.4.5) and bolt-spring elements (Section 2.9.3) to model plates connected by bolts.

2.8 Pipe elements

- Two types of pipe element are available: **pipe-beam element** (no ovalization/warping of the cross-section, the element has 2 or 4 nodes); **pipe-shell element** (with ovalization/warping, the element has 4 nodes). Figure 2.8-1 shows the conventions used.

- Assemblages of pipe elements can be subjected to internal pressure, see Section 2.8.2.
- The pipe-beam element has 6 displacement degrees of freedom per node (3 translations and 3 rotations, see Fig. 2.8-1).
The basic equations used in the formulation of the pipe-beam element are given in ref. KJB.

ref. KJB
Sections 5.4.1,
6.5.1



Element coordinate system convention:

longitudinal direction = a-direction (or η)

radial direction = b-direction (or ζ)

circumferential direction = c-direction (or ξ)

Figure 2.8-1: Pipe element configuration and displacement degrees of freedom and stress convention

- The pipe-shell element has, in addition to the 6 pipe-beam degrees of freedom, also 3 or 6 ovalization and 3 or 6 warping degrees of freedom per node allowing the cross-section to ovalize and warp.

The formulation of the pipe-shell element (ovalization/warping included) is described in the following references:

- ref. K.J. Bathe and C.A. Almeida, "A Simple and Effective Pipe Elbow Element – Linear Analysis," *J. Appl. Mech., Transactions of the ASME*, Vol. 47, No. 1, pp. 93-100, 1980.

ref. K.J. Bathe, C.A. Almeida and L.W. Ho, "A Simple and Effective Pipe Elbow Element – Some Nonlinear Capabilities," *Comp. & Structures*, Vol. 17, No. 5/6, pp. 659-669, 1983.

- The pipe-shell formulation also takes into account the effects of stiffening due to internal pressure, and the effects of interactions between pipes and flanges, and pipes with different radii, see Fig. 2.8-2. The formulations are described in the following references:

ref. K.J. Bathe and C.A. Almeida, "A Simple and Effective Pipe Elbow Element – Interaction Effects," *J. Appl. Mech., Transactions of The ASME*, Vol. 49, pp. 165-171, 1982.

ref. K.J. Bathe and C.A. Almeida, "A Simple and Effective Pipe Elbow Element – Pressure Stiffening Effects," *J. Appl. Mech., Transaction of the ASME*, Vol. 49, pp. 914-916, 1982.

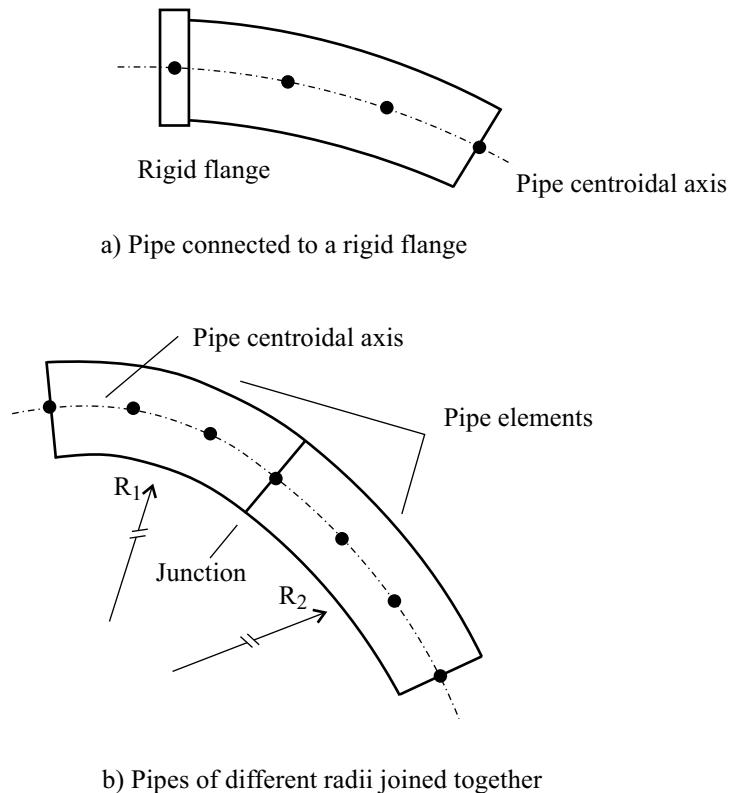


Figure 2.8-2: Interaction effects with the pipe element

- In the description of the pipe-shell element kinematics, reference is made to the in-plane and out-of-plane bending deformation modes. Note that the in-plane bending corresponds to the bending in the r-s plane and the out-of-plane bending corresponds to the bending in the r-t plane (see Fig. 2.8-1 and 2.8-3).
- Whereas the pipe-beam element models the usual beam strains only, the pipe-shell element models in addition ovalization and warping effects by including the following strains (see Fig. 2.8-1 for notation):
 - ▶ The ovalization is included by the von Karman ovalization modes with

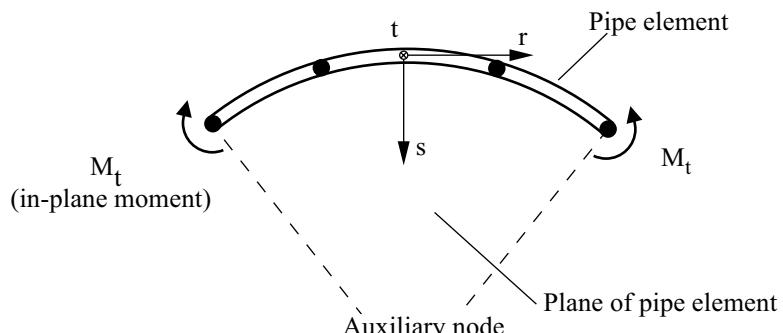
$$w_\xi = \frac{\partial w_\xi}{\partial \phi}$$

and

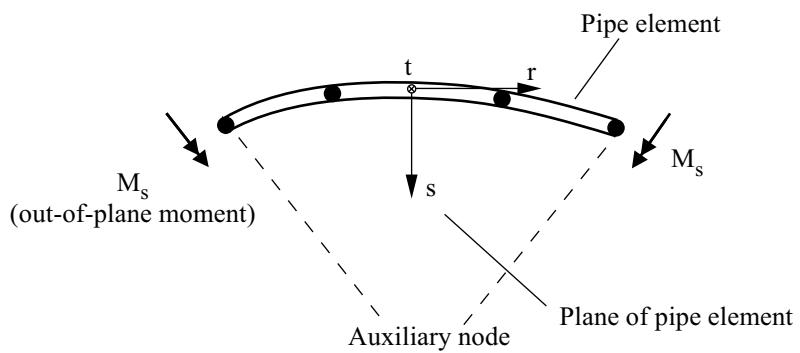
$$e_{aa}^{ov} = \frac{w_\xi \sin \phi - w_\zeta \cos \phi}{R - a \cos \phi} - \left(\frac{1}{(R - a \cos \phi)^2} \frac{\partial^2 w_\zeta}{\partial \theta^2} \right) \zeta$$

$$\gamma_{ab}^{ov} = 0 \quad (2.8-1)$$

$$\gamma_{ac}^{ov} = \frac{1}{R - a \cos \phi} \frac{\partial w_\zeta}{\partial \theta}$$



a) In-plane-only bending conditions



b) Out-of-plane bending conditions

Figure 2.8-3: Loading conditions for the pipe element

$$e_{cc}^{ov} = -\frac{1}{a^2} \left(w_\zeta + \frac{\partial^2 w_\zeta}{\partial \phi^2} \right) \zeta$$

- The warping is included using shell theory as

$$e_{aa}^w = \left(1 - \frac{\cos \phi}{R - a \cos \phi} \zeta \right) \left(\frac{1}{R - a \cos \phi} \right) \frac{\partial w_\eta}{\partial \theta}$$

$$\gamma_{ab}^w = 0$$

$$\gamma_{ac}^w = \frac{1}{a} \frac{\partial w_\eta}{\partial \phi} - \frac{w_\eta \sin \phi}{R - a \cos \phi} \quad (2.8-2)$$

$$e_{cc}^w = 0$$

Here, a is the pipe mean radius, R is the bend radius and θ and ϕ are the angles defined in Fig. 2.8-1.

In the formulation of the element the displacements w_ζ for the ovalization in Eq. 2.8-1 and w_η for the warping in Eq. 2.8-2 are interpolated by sine and cosine functions associated with the ovalization and warping degrees of freedom.

- Note that the above equations for the pipe-beam and pipe-shell elements give the same stress response if there is no ovalization. Hence, the ovalization and warping of the pipe-shell element can be thought of as an additional kinematic mode to the pipe-beam response which results in an appropriate reduction of stiffness of pipe bends.
- For the pipe-shell element, the following use of the ovalization/warping degrees of freedom is recommended:
 - Activate the in-plane ovalization/warping degrees of freedom if the pipe elements undergo only in-plane deformations, see Fig. 2.8-3(a).

- ▶ Activate the out-of-plane ovalization/warping degrees of freedom if the pipe elements undergo only out-of-plane deformations, see Fig. 2.8-3(b).
- ▶ Activate all ovalization/warping degrees of freedom if the pipe elements undergo general three-dimensional deformations.
- In order to model the presence of a flange at a pipe node (of a pipe-shell element), all ovalization/warping degrees of freedom should be deleted at the node. The deletion of all nodal ovalization/warping degrees of freedom implies that the pipe section at that node remains circular. In addition, the option of enforcing the zero-slope-of-pipe-skin must be employed to model the flange (see Fig. 2.8-4).
- Note that a symmetry condition implies zero warping but not necessarily zero ovalization.
- For a pipe elbow, a nondimensional geometric factor λ is defined as

$$\lambda = \frac{R\delta}{a^2 \sqrt{1-v^2}}$$

where R = pipe bend radius, δ = pipe wall thickness, a = pipe cross-section mean radius, v = Poisson's ratio.

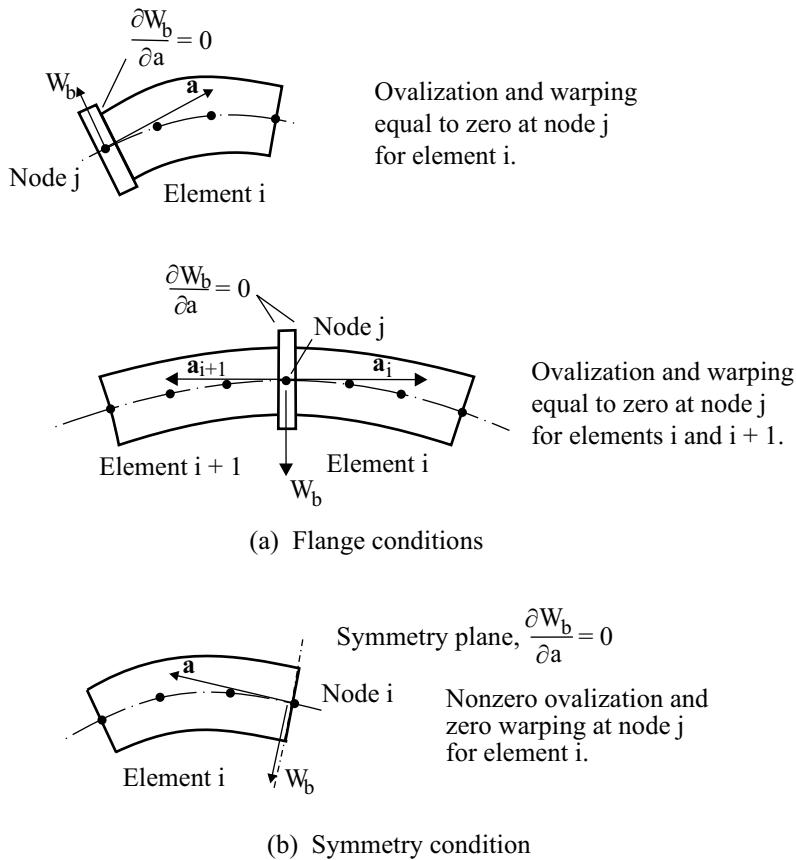


Figure 2.8-4: Zero-slope-of-pipe-skin conditions

With a decreasing value of λ , the ovalization/warping effects of the cross-section become more pronounced. Note that for a straight pipe element, $R \rightarrow \infty$ and thus the cross-section does not ovalize/warp except if connected to another pipe element with a finite bend radius.

2.8.1 Material models and formulations

- The pipe elements can be used with the following material models: **elastic-isotropic**, **plastic-bilinear**, **plastic-multilinear**, **thermo-isotropic**, **thermo-plastic**, **creep**, **plastic-creep**, **multilinear-plastic-creep**, **creep-variable**, **plastic-creep-variable**, **multilinear-plastic-creep-variable**.

- The pipe elements can be used with a **small displacement** or a **large displacement** formulation. In the small displacement formulation, the displacements and rotations are assumed to be infinitesimally small. In the large displacement formulation, the displacements and rotations are assumed to be large. In all cases, only small strains are assumed.

All of the material models in the above list can be used with either formulation. The use of a linear material with the small displacement formulation corresponds to a linear formulation, and the use of a nonlinear material with the small displacement formulation corresponds to a materially-nonlinear-only formulation.

In the large displacement formulation, large displacement effects are only included for the overall beam displacements and not for the ovalization degrees of freedom. Hence, large displacement effects are not accounted for in the element cross-sectional deformations.

- An element is to be considered a nonlinear pipe element if a large displacement analysis is performed and/or a nonlinear material is used and/or pipe internal pressure is present.

2.8.2 Pipe internal pressures

- If the element is loaded by internal pressure the load vector corresponding to that pressure is calculated as follows: First, two axial forces $\mathbf{F}_{al}^{(p)}$ and $\mathbf{F}_{a2}^{(p)}$ are acting at the element end nodes, as shown in Fig. 2.8-5. Their intensities are

$$F_{al}^{(p)} = \pi a_i^2 p_1 \quad ; \quad F_{a2}^{(p)} = \pi a_i^2 p_2$$

where p_1 and p_2 are the internal pressures at nodes 1 and 2 (input as nodal point pressures to the program), and a_i is the internal radius of the cross-section. Also, if the element is curved, transversal forces are generated by the internal pressure. The transversal force acting on the elementary axial length dL of the pipe axis is

$$d\mathbf{F}_T = -\frac{\pi a_i^2}{R} p \mathbf{s} dL$$

and the nodal transversal forces are obtained as

$$\mathbf{F}_k = \int_L h_k d\mathbf{F}_T$$

where h_k is the interpolation function corresponding to node k .

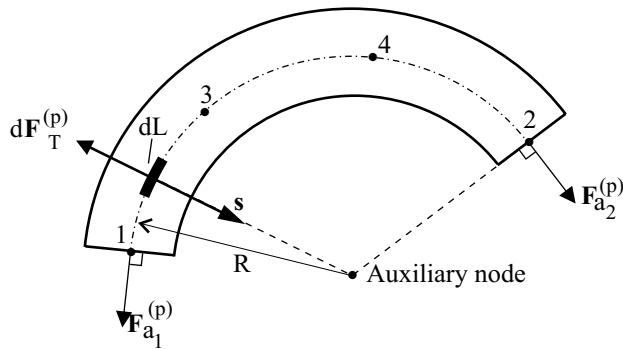
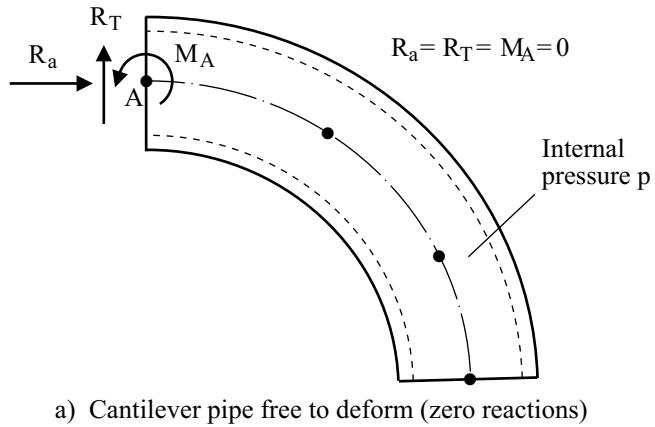
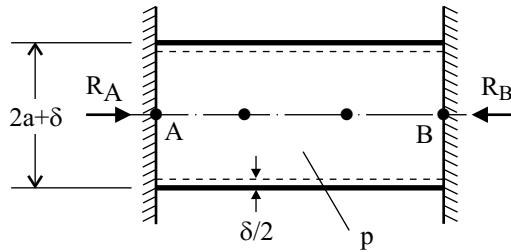


Figure 2.8-5: Forces due to pipe internal pressure

- The nodal forces due to the internal pressure for an element represent a self-balanced system. Consequently, if a structure is free to deform, as in the case of the cantilever in Fig. 2.8-6(a), the support reactions are equal to zero. However, in the case of a restrained system, the reactions at a support balance the forces and moments which are a result of the internal stresses and the nodal forces that are equivalent to the internal pressure, see Fig. 2.8-6(b).



a) Cantilever pipe free to deform (zero reactions)



In elastic deformations:

$$R_A = R_B = \pi a_i p (a_i - 2v a)$$

$$a_i = a - \delta/2$$

$$= \text{internal radius}$$

a = mean radius

v = Poisson's ratio

b) Fully restrained pipe

Figure 2.8-6: Support reactions for pipe loaded by internal pressure

- In the stress calculation the following assumptions are made regarding the internal pressure effects.
 - For the pipe-beam element it is assumed that the hoop stress $^{t+\Delta t} \sigma_{cc}$ is determined by the given pressure, $^{t+\Delta t} \sigma_{cc} = ^{t+\Delta t} \sigma_{cc}^{(p)}$, where $^{t+\Delta t} \sigma_{cc}^{(p)}$ is the hoop stress equivalent to the pressure $^{t+\Delta t} p$ (using thin-walled cylinder theory). Then, using that the normal stress through the pipe skin is equal to zero, the axial stress $^{t+\Delta t} \sigma_{aa}$ is obtained as

$${}^{t+\Delta t} \sigma_{aa} = {}^{t+\Delta t} E \left({}^{t+\Delta t} e_{aa} - {}^{t+\Delta t} e_{aa}^{IN} - {}^{t+\Delta t} e^{TH} \right) + {}^{t+\Delta t} \nu {}^{t+\Delta t} \sigma_{cc}^{(p)}$$

where ${}^{t+\Delta t} E$ and ${}^{t+\Delta t} \nu$ are the Young's modulus and Poisson's ratio, and ${}^{t+\Delta t} e^{TH}$ is the thermal strain corresponding to the temperature ${}^{t+\Delta t} \theta$; ${}^{t+\Delta t} e_{aa}$ and ${}^{t+\Delta t} e_{aa}^{IN}$ are the total mechanical and inelastic strains, respectively.

- For the pipe-shell element the basic condition regarding the deformation of the pipe skin is that the material is free to deform in the circumferential direction. Hence, the total hoop strain is

$$\begin{aligned} {}^{t+\Delta t} e_{cc} = & -{}^{t+\Delta t} \nu \left({}^{t+\Delta t} e_{aa} - {}^{t+\Delta t} e_{aa}^{IN} \right) - \frac{1}{2} {}^{t+\Delta t} e_{aa}^{IN} + {}^{t+\Delta t} e_{cc}^{OV} \\ & + (1 + \nu) {}^{t+\Delta t} e^{TH} + \left(\frac{1 - \nu^2}{E} \right) \sigma_{cc}^{(p)} \end{aligned}$$

where ${}^{t+\Delta t} e_{cc}^{OV}$ is the strain due to ovalization of the cross-section. This relation is based on the assumption that the inelastic deformation is incompressible (as it is in metal plasticity and creep).

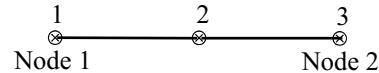
- Pressures at all nodal points of a piping model using the pipe-beam or pipe-shell elements (and other elements) must be defined if internal pressure effects are to be included in some of the elements. The pressures are only used for the pipe-beam and pipe-shell elements. See Section 5.7 for information about specifying internal pressures.

2.8.3 Numerical integration

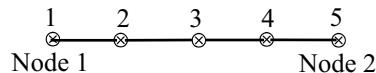
- The element stiffness and mass matrices and force vectors are in all formulations evaluated using Gauss or Newton-Cotes numerical integration in the axial and the thickness directions and the composite trapezoidal rule along the circumference of the pipe section. The locations and labeling of the integration points are given in Fig. 2.8-7.

Newton-Cotes formulas:

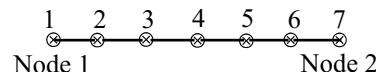
Order -3



Order -5

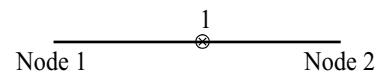


Order -7

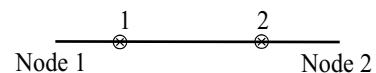


Gauss formulas:

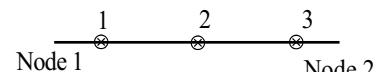
Order 1



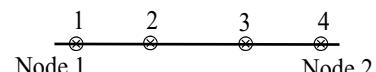
Order 2



Order 3

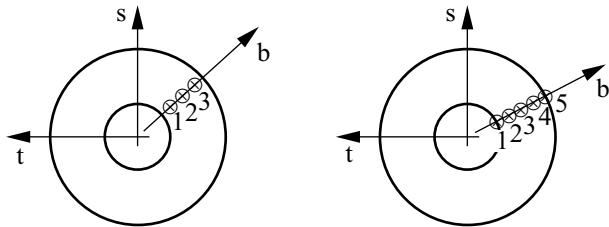


Order 4



a) Integration point locations on pipe centroidal axis (label INR)

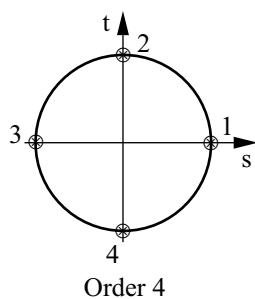
Figure 2.8-7: Locations and labeling of integration points



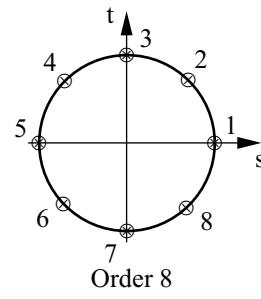
Gauss: order 3

Newton-Cotes: order 5

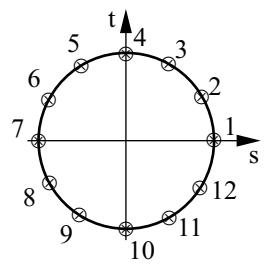
b) Integration point locations along thickness direction (label INB)



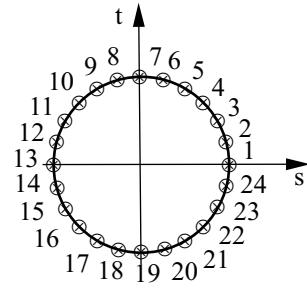
Order 4



Order 8



Order 12



Order 24

c) Integration point locations along circumferential direction (label INC)

Figure 2.8-7: (continued)

- For the pipe-beam elements the following integration orders should be used (and are default in ADINA):
 - ▶ 1-point and 3-point Gauss integration along the axial direction for the 2-node and 4-node elements, respectively

- ▶ 2-point Gauss integration through the thickness of the pipe skin
- ▶ 8-point integration along the circumference
- For the pipe-shell element the following integration orders should be used (and are default in ADINA):
 - ▶ 3-point Gauss integration along the axial direction
 - ▶ 2-point Gauss integration through the thickness of the pipe skin
 - ▶ 12-point integration along the circumference if only the in-plane ovalization degrees of freedom are activated
 - ▶ 24-point integration along the circumference if the out-of-plane ovalization degrees of freedom are activated
- The recommended numerical integration orders for nonlinear analysis are:
 - ▶ When no inelastic strain effects are considered, use the same integration orders as in linear analysis.
 - ▶ When material nonlinear conditions are considered, use a higher integration order through the pipe skin if the pipe skin progressive yielding or creep is to be represented. For the integration along the length of the elements and the circumference of the cross-section, an integration order higher than that used in linear analysis may for the same reason also be appropriate.

2.8.4 Bolt element

- A bolt element is a pre-stressed pipe element with a given initial force. The initial force is held constant during the equilibrium iterations at the time of solution start, so that the initial force at the start of the first solution step is the input initial force, no matter what other external loads or initial strains/stresses are input.

- During the solution for the first, second and successive solution steps, the bolt force can vary depending on the loads applied to the model.
- Bolt elements are created in a pipe element group with the bolt option activated. The initial force is entered either on the geometry (e.g., AUI command LINE-ELEMENTDATA PIPE) or directly specified on the elements (e.g., AUI command EDATA).
- The formulation used for the bolt elements can either be small displacements or large displacements. But only the isotropic elastic material model can be used.

2.8.5 Element mass matrices

*ref. KJB
Section
5.4.1*

- The pipe element can be used with a lumped or a consistent mass matrix, except for explicit dynamic analysis which always uses a lumped mass.
- The consistent mass matrix associates mass only with the pipe's translational and rotational degrees of freedom and is calculated using the isoparametric formulation with the displacement interpolations given on p. 408 of ref. KJB. Hence, no masses are associated with the ovalization/warping degrees of freedom.
- The lumped mass for translational degree of freedom i is $M \cdot \left(\frac{\ell_i}{L}\right)$ where M is the total element mass, L is the element length and ℓ_i is a fraction of the length associated with node i . The rotational lumped mass is $\frac{1}{3} \cdot M \cdot \left(\frac{\ell_i}{L}\right) \cdot \left(\frac{1}{4} (d_o^2 + d_i^2)\right)$ where d_o and d_i are the outside and inside diameters of the pipe. The lumped ovalization and warping mass for each of the ovalization and warping degrees of freedom is $M \cdot \left(\frac{\ell_i}{L}\right)$.
- The rotational lumped mass can be multiplied by a user-specified scalar ETA (except in explicit analysis).

2.8.6 Element output

You can request that ADINA either print or save stresses or forces.

Stresses: Each element outputs, at its integration points, the following information to the porthole file, based on the material model. This information is accessible in the AUI using the given variable names.

Elastic-isotropic: STRESS (ABC) , STRAIN (ABC) ,
FE_EFFECTIVE_STRESS,
EQUIV_INTERNAL_AXIAL_PRESSURE,
EQUIV_INTERNAL_HOOP_PRESSURE

Elastic-isotropic with thermal effects: STRESS (ABC) ,
STRAIN(ABC) , FE_EFFECTIVE_STRESS,
EQUIV_INTERNAL_AXIAL_PRESSURE,
EQUIV_INTERNAL_HOOP_PRESSURE , THERMAL_STRAIN,
ELEMENT_TEMPERATURE

Thermo-isotropic: STRESS (ABC) , STRAIN (ABC) ,
THERMAL_STRAIN (ABC) ,
EQUIV_INTERNAL_AXIAL_PRESSURE,
EQUIV_INTERNAL_HOOP_PRESSURE,
ELEMENT_TEMPERATURE

Plastic-bilinear, plastic-multilinear: PLASTIC_FLAG,
STRESS (ABC) , STRAIN (ABC) ,
PLASTIC_STRAIN (ABC) , THERMAL_STRAIN (ABC) ,
EQUIV_INTERNAL_AXIAL_PRESSURE,
EQUIV_INTERNAL_HOOP_PRESSURE,
FE_EFFECTIVE_STRESS , YIELD_STRESS,
ACCUM_EFF_PLASTIC_STRAIN , ELEMENT_TEMPERATURE

Thermo-plastic, creep, plastic-creep, multilinear-plastic-creep,
creep-variable, plastic-creep-variable, multilinear-plastic-creep-
variable: PLASTIC_FLAG , NUMBER_OF_SUBINCREMENTS ,
STRESS (ABC) , STRAIN (ABC) ,
PLASTIC_STRAIN (ABC) , CREEP_STRAIN (ABC) ,
THERMAL_STRAIN (ABC) , ELEMENT_TEMPERATURE ,
ACCUM_EFF_PLASTIC_STRAIN ,
FE_EFFECTIVE_STRESS , YIELD_STRESS ,

EQUIV_INTERNAL_AXIAL_PRESSURE,
EQUIV_INTERNAL_HOOP_PRESSURE,
EFFECTIVE_CREEP_STRAIN

In the above lists,

STRESS (ABC) = STRESS-AA, STRESS-AB, STRESS-AC, STRESS-CC, STRESS-BB

with similar definitions for the other abbreviations used above. See Fig. 2.8-1 for the pipe stress coordinate system convention.

See Section 12.1.1 for the definitions of those variables that are not self-explanatory.

Nodal forces: In linear analysis, the forces/moment at the element nodes are

$$\mathbf{F} = \mathbf{K} \mathbf{U}$$

where \mathbf{K} is the element stiffness matrix and \mathbf{U} is the vector of nodal point displacements which may include ovalizations and warpings. Hence, the forces/moment in \mathbf{F} will be equal to the forces/moment applied externally or by adjoining elements, and are not necessarily the internal section forces/moment.

The nodal forces are accessible in the AUI using the variable names NODAL_FORCE-R, NODAL_FORCE-S, NODAL_FORCE-T, NODAL_MOMENT-R, NODAL_MOMENT-S, NODAL_MOMENT-T.

The end forces/moment are computed at the element local nodes. In the AUI, element local nodes are defined as element points of type label. For example, to access the result computed at element 5, local node 2, define an element point of type label with element number 5, label number 2.

2.8.7 Recommendations on use of elements

- The pipe elements are available in ADINA to model general pipe networks subjected to general boundary and loading conditions including internal pressure and thermal loading.
- When ovalization effects are insignificant (straight pipes or pipes of large thickness) the 2- or 4-node pipe-beam element

should be used. In this case, linear or nonlinear behavior can be modeled with large displacements but small strains.

- When ovalization effects are important, the 4-node pipe-shell element should be used. In this case, linear or nonlinear behavior can be modeled with the assumptions that the displacements are always relatively small and that the strains are small. The displacements are always relatively small because in the calculation of the flexibility due to ovalization/warping the pipe radius is constant (and equal to the initial pipe radius) and no large displacement strain terms are included in the ovalization/warping calculations. The only large displacement strain terms accounted for are those corresponding to the beam behavior.

However, if the large displacement option is employed, the change of the pipe radius is taken into account in the calculation of the element nodal forces due to the pipe internal pressure.

ref. K.J. Bathe and C.A. Almeida, "A Simple and Effective Pipe Elbow Element – Linear Analysis," *J. Appl. Mech., Transactions of the ASME*, Vol. 47, No. 1, pp. 93-100, 1980.

- Since the pipe element stiffness matrix is calculated using numerical integration, it is clear that the linear 2-node Hermitian beam element (see Section 2.4.1) for which the stiffness matrix is evaluated in closed-form can be much more cost-effective in analyzing linear piping systems with negligible ovalization effects.

2.9 General and spring/damper/mass elements

2.9.1 General elements

- General elements are linear elements which can have from one node to as many nodes as there are in the structure.
- General elements are useful in the definition of special elements whose formulations are not directly available in ADINA. General elements are also useful in providing flexibility in obtaining desirable stiffness/mass/damping matrices.

- Note that in a dynamic analysis, the only other alternative besides the use of general elements in introducing a consistent damping matrix is to specify Rayleigh damping.
- You directly enter the stiffness matrix (and in dynamic analysis, the mass and damping matrices) of the general element.

The dimension of the stiffness/mass/damping matrices cannot exceed ND, defined as NDOF×IELD, where IELD is the number of nodes of the general element and NDOF is the number of active degrees of freedom. The components for the stiffness/mass/damping matrices must be defined for all ND degrees of freedom. Note that only the upper triangular half of the matrices needs to be input (the stiffness/mass/damping matrices must be symmetric).

For example, for a 4-node 3-D solid element with one active degree of freedom, ND = 1 × 4 = 4; the stiffness matrix is 4 × 4 symmetric, and the matrix is entered as follows:

```
matrix stiffness 1 4
1 0.5E8 -0.5E7 -0.1E8 -0.5E7 0.1E8
2 0.5E8 -0.5E7 -0.1E8
3 0.5E8 -0.5E7
4 0.5E8
dataend
```

- An alternative is to define a user-supplied general element, see Section 2.9.4.

Skew system input option: Either all matrices are input in the global coordinate system and proper transformations are made by ADINA if there are skew systems at the element group nodes, or all matrices are input in the coordinate systems used at the nodes and no transformations between global and skew systems are made in ADINA (SKEWSYSTEMS = YES in the EGROUP GENERAL or EGROUP SPRING command).

- Forces are calculated for general elements using

$$\mathbf{F} = \mathbf{K} \mathbf{U}$$

where \mathbf{U} stores the displacements for the ND degrees of freedom.

The forces are accessible in the AUI using the variable names

NODAL_FORCE-X, NODAL_FORCE-Y, NODAL_FORCE-Z,
NODAL_MOMENT-X, NODAL_MOMENT-Y, NODAL_MOMENT-Z.

- Stresses can also be calculated for general elements if you define a stress-displacement matrix \mathbf{S} . The stress vector $\boldsymbol{\sigma}$ is obtained using

$$\boldsymbol{\sigma} = \mathbf{S} \mathbf{U}$$

Note that the matrix \mathbf{S} must be input as a full matrix and is of dimension NS×ND, where NS equals the number of stress components in $\boldsymbol{\sigma}$ (NS must be less than or equal to 600).

Each element outputs the vector $\boldsymbol{\sigma}$ to the porthole file. This vector is accessible in the AUI using the variable name GENERAL_ELEMENT_STRESS. You select the component of $\boldsymbol{\sigma}$ in the AUI by setting the label point number equal to the component number in $\boldsymbol{\sigma}$.

2.9.2 Linear and nonlinear spring/damper/mass elements

Linear spring single-degree-of-freedom element

- The stiffness matrix has only 1 component and corresponds to a grounded spring acting in the single (user-specified) degree of freedom, see Fig. 2.9-1(a).
- The mass matrix corresponds to a concentrated mass acting in the single degree of freedom, see Fig. 2.9-1(b).
- The damping matrix corresponds to a grounded damper acting in the single degree of freedom, see Fig. 2.9-1(c).

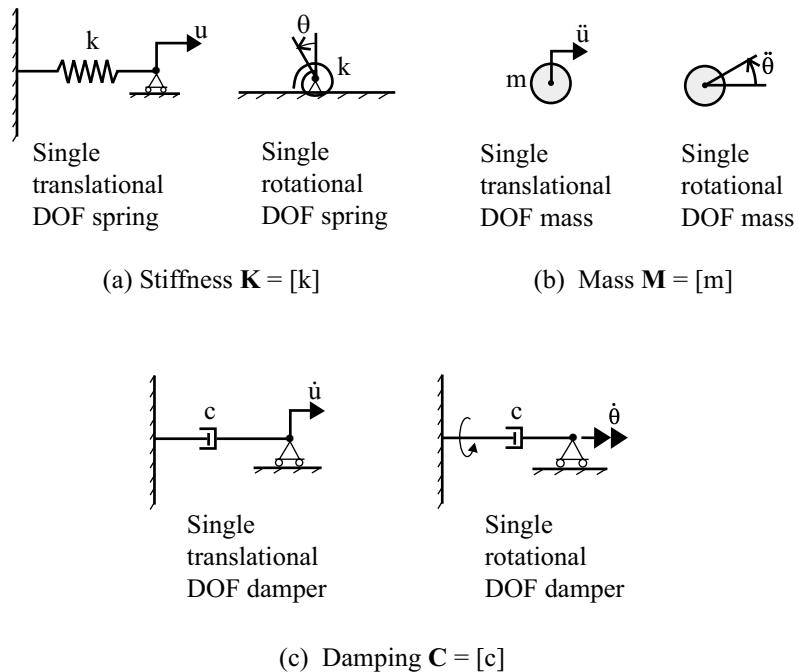


Figure 2.9-1: Single-degree-of-freedom element

Linear spring two-degree-of-freedom element

- The stiffness matrix is shown in Fig. 2.9-2(a) and corresponds to a spring coupling the two (user-specified) degrees of freedom.
- The lumped mass matrix is diagonal, see Fig. 2.9-2(b), and corresponds to two concentrated masses acting in the two degrees of freedom.
- The consistent mass matrix is obtained using a linear interpolation of the accelerations in the two degrees of freedom, see Fig. 2.9-2(b) and ref. KJB, Example 4.5, pp. 166-170.
- The damping matrix is shown in Fig. 2.9-2(c) and corresponds to a damper coupling the two degrees of freedom.

$$\mathbf{K} = \begin{bmatrix} k & -k \\ -k & k \end{bmatrix} \quad \mathbf{M}_{\text{lumped}} = \begin{bmatrix} m/2 & 0 \\ 0 & m/2 \end{bmatrix}$$

k = spring constant

a) Stiffness matrix

$$\mathbf{M}_{\text{consistent}} = \begin{bmatrix} m/3 & m/6 \\ m/6 & m/3 \end{bmatrix}$$

m = total mass
of element

$$\mathbf{C} = \begin{bmatrix} c & -c \\ -c & c \end{bmatrix}$$

b) Mass matrix

c = damping constant

c) Damping matrix

Figure 2.9-2: Two-degrees-of-freedom element matrices

Nonlinear spring/damper/mass element

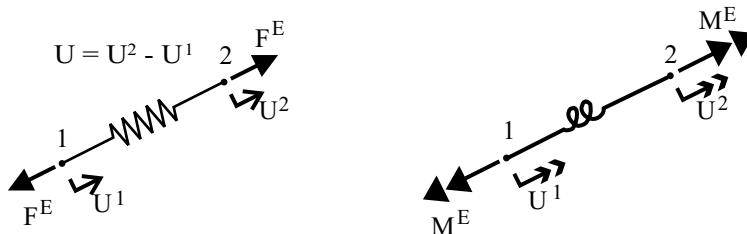
- A nonlinear "spring" element is available in ADINA for static and dynamic analysis. This element can be used to connect two nodes or to attach a node to the ground. It can be a translational or a rotational "spring" (see Figure 2.9-3). The stiffness, damping and mass of the element can each be either nonlinear or linear.
- Large displacements can be applied to nonlinear spring elements.
- In the case of a translational spring, the nonlinear stiffness is defined by a multilinear elastic force versus relative-displacement relationship. The corresponding curve can be different for negative relative-displacement than for positive relative-displacement. The input curve need not pass through the origin (0,0).

The nonlinear damping is given by a function of the form

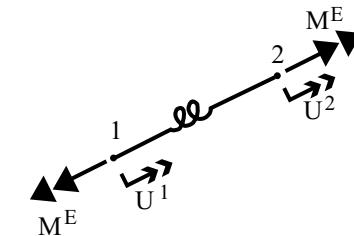
$|F^D| = C |\dot{U}|^N$ where F^D is the damping force in the element, \dot{U} is the relative velocity between the element end nodes, and C and N are real constants.

The mass of the element can vary with time (see Figure 2.9-4).

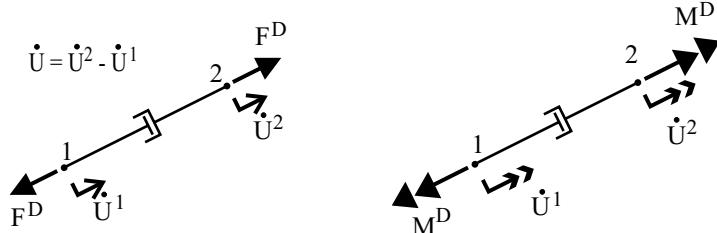
Nonlinear translational stiffness:



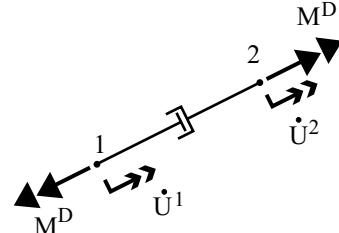
Nonlinear torsional stiffness:



Nonlinear translational damping:



Nonlinear torsional damping:



Nonlinear translational stiffness,
grounded spring:

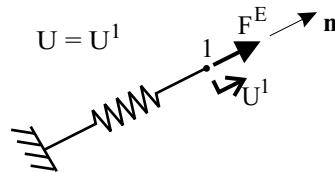
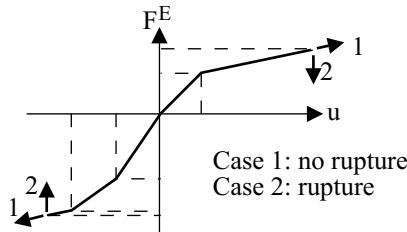


Figure 2.9-3: Description of the nonlinear spring/damper/mass element

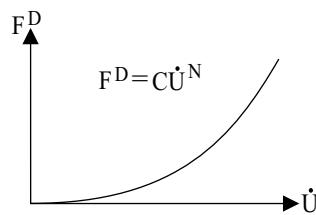
The relative displacement \mathbf{U} is defined as follows: Let \mathbf{U}^1 and \mathbf{U}^2 be the displacement vectors at local nodes 1 and 2, and ${}^0\mathbf{x}^1$ and ${}^0\mathbf{x}^2$ be the respective initial positions of these nodes.

For initially coincident nodes (${}^0\mathbf{x}^1 = {}^0\mathbf{x}^2$), $U = U_{id}^2 - U_{id}^{dx1}$ where id specifies a global direction in the global Cartesian coordinate system, and is specified in the input. $id = 1,2,3$ for the X, Y, Z translational directions and $id = 4,5,6$ for the rotation about the X, Y, Z directions. There is also an option (using $id = 0$) to use another arbitrary direction.

Force versus displacement or moment versus rotation:



Force versus velocity or moment versus torsional velocity:



Mass versus time:

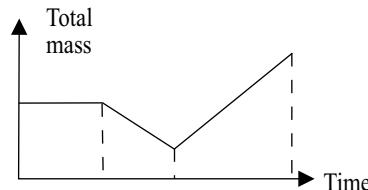


Figure 2.9-4: Nonlinear stiffness, damping and mass

If the nodes are not initially coincident, then the element can be used to define a translational spring along the direction given by the vector from node 1 to node 2. Let \mathbf{U}^T store only the translational displacements, then the relative translational

displacement is given by $U = (\mathbf{U}^2 - \mathbf{U}^1) \cdot \frac{(^0\mathbf{x}^2 - ^0\mathbf{x}^1)}{\|(^0\mathbf{x}^2 - ^0\mathbf{x}^1)\|}$, i.e., the

relative displacement is the difference in displacements in the direction of the line (spring) element.

- In a similar manner, a torsional spring is defined between the local nodes 1 and 2.
- Similar definitions are used for the relative velocity.
- Rupture based on the value of the relative displacement may or may not be included in the nonlinear spring element (see Figure 2.9-4). If rupture is not desired, the last segments of the elastic-force versus relative displacement curve are extrapolated when necessary. If rupture is desired, then the elastic force in the element is set to zero whenever the relative displacement is larger

(resp. smaller) than the last (resp. first) point on the curve. Note that this effect is reversible, as the force will be computed again according to the curve if the displacement value comes back into the curve displacement range.

- Note that the linear spring element and the nonlinear spring element act quite differently: the linear spring element connects specific degrees of freedom, whereas the nonlinear spring element connects nodes and acts in a specified direction. Likewise the input for these elements is also different.

Element output

- You can request that ADINA either print or save stress resultants or forces.

Stress resultants: Each element outputs the following information to the porthole file. This information is accessible in the AUI using the given variable names.

Linear element: GENERAL_ELEMENT_STRESS (NS values per element)

Nonlinear element: DAMPING_FORCE, ELASTIC_FORCE

Nodal forces: These are the elastic nodal forces (in the global Cartesian reference system) equivalent to the axial elastic stress.

The forces are accessible in the AUI using the variable names NODAL_FORCE-X, NODAL_FORCE-Y, NODAL_FORCE-Z, NODAL_MOMENT-X, NODAL_MOMENT-Y, NODAL_MOMENT-Z.

2.9.3 Spring-tied element

- The spring-tied element is used for the modeling of a connection between touching, or closely-spaced, shell surfaces. The spring-tied element consists of six spring elements and additional constraint equations. Spring-tied elements can also be used to model a connection between a shell surface and ground.

The constraint equations are the same as are employed for the beam-tied element, see Section 2.4.5.

- Note that the distance between the nodes of the spring-tied element must be smaller than a maximum distance. This maximum distance can be a relative value (proportional to the model dimensions) or an absolute value. For longer connections, the beam-tied element should be used.
- The spring-tied element must be used when the beam-tied element cannot be used due to element nodes being too close to each other.
- Also see the remarks about modeling given for the beam-tied element, see Section 2.4.5.

2.9.4 User-supplied element

- With this option you code a nonlinear general element in the ADINA subroutine CUSERG. Subroutine CUSERG calculates the element stiffness matrix and nodal force vector, and also prints the results and outputs the results into the porthole file.
- The following element types are allowed:
 - ▶ Triangular 3-node, 6-node, or 7-node 2-D solid element
 - ▶ Quadrilateral 4-node, 8-node, or 9-node 2-D solid element
 - ▶ Tetrahedral 4-node, 10-node, or 11-node 3-D solid element
 - ▶ Hexahedral 8-node, 20-node, 21-node, or 27-node 3-D solid element
 - ▶ 2-node, 3-node, or 4-node beam element
 - ▶ 4-node, 8-node, 9-node, or 16-node shell element
 - ▶ Variable number of nodes up to maximum 32 nodes transition shell element

The nodal connectivity of the user-supplied elements must be compatible with the ordinary ADINA elements so that the user-supplied elements can be generated by the AUI mesh generation

commands and so that the user-supplied elements can be loaded into the AUI for post-processing.

Six degrees of freedom per node is always assumed for the shell user-supplied element.

- It is necessary to include the user-supplied general element coding into subroutine CUSERG. A template for CUSERG is distributed in file ovl160u.f. The Makefile includes instructions for compiling and linking this file.
- The input to the AUI for the user-supplied general element consists of four parts: element group definition, matrix set definition, matrix definition and material definition.

The user-supplied general elements belong to a general element group, which is nonlinear.

As with any general element, the stiffness, mass and damping matrices must be provided (of course, the mass and damping matrices are necessary only for certain analysis types). The stiffness, mass and damping matrices are grouped into a matrix set and the matrix set number is included as part of the element group definition. In ordinary general elements (Section 2.9.1), you directly specify the stiffness matrix and other matrices. In user-supplied general elements, subroutine CUSERG calculates the stiffness matrix, and the mass matrix and damping matrix are calculated as in ordinary general elements.

The element subtype, material label, integration schemes, etc. are provided to ADINA through a matrix definition (in the AUI, the matrix definition is accessed as a user-supplied matrix from the Matrix Set dialog box). This matrix definition includes

- ▶ The element subtype (2-D solid, 3-D solid, beam or shell)
- ▶ The material (the material number from the user-supplied material definition). The material properties can be temperature-independent or temperature-dependent.
- ▶ The total number of user-provided integration points NUIPT. The default value of NUIPT is NUIT1*NUIT2*NUIT3.
- ▶ The integration order in the first local direction NUIT1. NUIT1 is ignored if NUIPT is entered.

- ▶ The integration order in the second local direction NUIT2. NUIT2 is ignored if NUIPT is entered.
- ▶ The integration order in the third local direction NUIT3. NUIT3 is ignored if NUIPT is entered.
- The calls from ADINA to CUSERG are divided into several phases, as controlled by integer variable KEY.

KEY=1: Integration point locations are to be provided, these points are used to store stress and strain results. The array XYZIPT provides these integration location coordinates and is arranged through the sequence of 1 to NUIPT points or through the local integration orders by NUIT1, NUIT2 and NUIT3. The definitions of NUIT1, NUIT2 and NUIT3 are the same as the ADINA 2-D solid, 3-D solid and shell element integration point orders. The array XYZ gives the element nodal coordinates and is passed from ADINA, then can be used for the calculations of integration point coordinates.

KEY=2: Element nodal forces RE are to be calculated based on the latest displacements DISP2 and the displacements DISP1 at the last time step. Also, the material properties and parameters can be used if necessary. RE has the dimension size of ND, the total number of element degrees of freedom. The nodal forces are the equilibrating forces during the equilibrium iterations and the resultant nodal forces in the converged solution. RE must be expressed in terms of the global coordinate system. We also provide an additional array REBM which can store the beam nodal forces in the local system for display purposes.

If variables are history dependent, they must be calculated and stored in the ARRAY and IARRAY arrays. Each column of these arrays stores the variables for one integration point.

KEY=3: The element stiffness matrix AS(ND,ND) is to be calculated, based on the element coordinates, latest displacements DISP2, last converged displacements DISP1, and the constants and parameters defined in the user-supplied material. Some material properties are temperature-dependent and are interpolated by ADINA before they are passed to CUSERG. Temperature dependent and independent properties

are used to calculate the material matrix.

Note that the RE vector and AS matrix must be expressed in terms of the global coordinate system.

KEY=4: Some element results are to be written to the porthole file, e.g. stress and strain components. These quantities must be assigned to the indicated addresses of the arrays RUPLOT and IUPLLOT in order to be displayed during postprocessing. Please see the source code comments for RUPLOT and IUPLLOT for more information.

- Some useful arguments passed to CUSERG are described below:

NG	Element group number
NEL	Element number
IELD	Number of nodes in the element
ND	Number of degrees of freedom in the element
NOD(IELD)	Global node numbers of the element
TEMP1(IELD)	Element temperature at time t
TEMP2(IELD)	Element temperature at time $t + \Delta t$
TREF	Reference temperature
SCP(99)	Solution control parameters
CTD(98,IELD)	Temperature-dependent constants at time t for each node
CTDD(98,IELD)	Temperature-dependent constants at time $t + \Delta t$ for each node
CTI(99)	Temperature-independent constants
ALFA(IELD)	Coefficient of thermal expansion for each node at time t
TIME	Solution time at the current time step.
DT	Time step increment.
IDEATH	Element birth/death options, =0; option not used =1; elements become active at time of birth =2; elements become inactive at time of death =3; elements become active at time of birth, then inactive at time of death
ETIME	Birth time of the current element.

ETIMV2	Death time of the current element.
LGTH1	Length of real working array ARRAY at each integration point
LGTH2	Length of integer working array IARRAY at each integration point

Steps for using the user-supplied element option. These steps are given in terms of the AUI command language.

- 1) Compile ovl160u.f and link as described in the Installation Notes for your computer.
- 2) Use the MATERIAL USER-SUPPLIED command to define the material properties, control parameters and working array sizes, i.e. CTI, CTD, SCP, LGHT1 and LGTH2.
- 3) Use the MATRIX USER-SUPPLIED command to define the stiffness matrix.
- 4) Use the MATRIX MASS and MATRIX DAMPING commands to define the mass and the damping matrices if necessary.
- 5) Use the MATRIXSET command to define the general element matrixset.
- 6) Enter the EGROUP GENERAL USER-SUPPLIED=YES MATRIXSET=(defined in the MATRIXSET command) command.
- 7) You can generate the user-supplied elements using the usual mesh generation commands, as long as the element subtype is restricted to 2-D solid, 3-D solid, beam and shell elements.
- 8) Define the rest of the input (i.e. boundary conditions, loadings) as usual and generate a data file. Run ADINA as usual and display the results as usual.

2.10 Displacement-based fluid elements

- The elements discussed in this section incorporate the following assumptions:
 - ▶ Inviscid, irrotational medium with no heat transfer
 - ▶ Compressible or almost incompressible medium
 - ▶ Relatively small displacements
 - ▶ No actual fluid flow
- For other types of fluid flow, use ADINA-F and ADINA-FSI (see the ADINA-F Theory and Modeling Guide for details of the formulations employed). In the remainder of this section, only the ADINA fluid elements are discussed.
- The types of problems for which the ADINA displacement-based fluid elements can be employed are:
 - ▶ Static analyses, where the pressure distribution in the fluid and the displacement and stress distribution in the structure is of interest
 - ▶ Frequency analyses, where natural frequencies and mode shapes of a structure/fluid medium are to be determined
 - ▶ Transient analyses, where a pressure wave propagates rapidly through the fluid, which does not undergo large motions (transient acoustic problems)

However, in practice, the use of the displacement-based elements is rather restricted to special applications in static and dynamic analyses. The potential based element (see Section 2.11) is much more general and is recommended for use for solving this class of problems.

- Simply stated, the displacement-based fluid elements can be thought of as derived from the solid two- and three-dimensional

elements (see Sections 2.2 and 2.3) by using an elastic stress-strain relation with a bulk modulus K and a zero shear modulus.

- The elements can be employed in 2-D and 3-D analyses. Two-dimensional fluid elements can be employed in planar and axisymmetric analyses. Two-dimensional fluid elements must be defined in the YZ plane, and axisymmetric elements must lie in the +Y half plane (all nodal point coordinates must have positive y values).
- Although the fluid elements can be employed using a large displacement formulation, the allowed fluid motion is relatively small in a practical analysis, because the elements must not become distorted. Actual flow of a fluid cannot be analyzed using the elements. Use ADINA-F or ADINA-FSI if the allowed fluid motion is large.
- Difficulties in the use of the elements and various experiences in solutions obtained are discussed in the following references:

- ref. K.J. Bathe and W. Hahn, "On Transient Analysis of Fluid-Structure Systems," *Computers & Structures*, Vol. 10, pp. 383-391, 1981.
- ref. J. Sundqvist, "An Application of ADINA to the Solution of Fluid-Structure Interaction Problems," *Computers & Structures*, Vol. 17, pp. 793-808, 1983.
- ref. L. Olson and K.J. Bathe, "A Study of Displacement-Based Fluid Finite Elements for Calculating Frequencies of Fluid and Fluid-Structure Systems," *Nuclear Engineering and Design*, Vol. 76, pp. 137-151, 1983.

For example, in Olson and Bathe, it is noted that problems involving structures moving through fluids that behave almost incompressibly (e.g., an ellipse vibrating on a spring in water) cannot be solved satisfactorily with the displacement-based fluid elements.

- In linear analysis, you can impose irrotational conditions in the element formulation. This is achieved using a penalty constraint.

Note that if the penalty constraint is imposed, rigid body rotations of the elements are no longer possible.

- These elements are defined within 2-D and 3-D fluid element groups (in ADINA). Set the formulation of the element group to either displacement-based without rotation penalty or displacement-based with rotation penalty (if you are using the AUI user interfaces, set the formulation with the “interpolation type” field).
- Pressures are output at the integration points. The integration point numbering is the same as the numbering convention used for the solid elements, see Sections 2.2.3 and 2.3.3.

The pressure is evaluated using the following relation:

$$p = -K e_v$$

where p is the pressure, K is the bulk modulus and e_v is the volumetric strain (Δ volume/volume).

The pressures and strain components are accessible in the AUI using the variable names FE_PRESSURE, STRAIN-XX, STRAIN-YY, STRAIN-ZZ, STRAIN-XY, STRAIN-XZ, STRAIN-YZ (STRAIN-XY and STRAIN-XZ are applicable only for 3-D elements). See Section 12.1.1 for the definitions of those variables that are not self-explanatory.

- You can also request nodal force output. The nodal forces are accessible in the AUI using the variable names NODAL_FORCE-X, NODAL_FORCE-Y, NODAL_FORCE-Z (NODAL_FORCE-X is applicable only for 3-D elements).

2.11 Potential-based fluid elements

- The elements discussed in this section incorporate the following assumptions:
 - ▶ Inviscid, irrotational medium with no heat transfer
 - ▶ Compressible or almost incompressible medium

- ▶ Relatively small displacements of the fluid boundary
- ▶ Actual fluid flow with velocities below the speed of sound (subsonic formulation) or no actual fluid flow (linear infinitesimal velocity formulation)
- For other types of fluid flow, use ADINA-F and ADINA-FSI (see the ADINA-F Theory and Modeling Guide for details of the formulations employed). In the remainder of this section, only the ADINA fluid elements are discussed.
- The types of problems for which the ADINA potential-based fluid elements can be employed are:
 - ▶ Static analyses, where the pressure distribution in the fluid and the displacement and stress distribution in the structure is of interest
 - ▶ Frequency analyses, where natural frequencies and mode shapes of a structure/fluid medium are to be determined
 - ▶ Transient analyses, where a pressure wave propagates rapidly through the fluid, which does not undergo large motions (transient acoustic problems)
 - ▶ Transient analyses, where fluid flows through the domain, and the boundaries of the domain undergo only small motions.
- The potential-based fluid elements can be employed in 2-D and 3-D analyses. Two-dimensional elements can be employed in planar and axisymmetric analyses. Two-dimensional elements must be defined in the YZ plane, and axisymmetric elements must lie in the +Y half plane (all nodal point coordinates must have non-negative y values).
- The potential-based fluid elements can be coupled with ADINA structural elements, as described in detail below. The structural motions cause fluid flows normal to the structural boundary, and the fluid pressures cause additional forces to act on the structure.

- The potential-based fluid elements can be coupled to a pressure boundary condition (i.e., no structure adjacent to the potential-based fluid element boundary). This feature can be used to model free surfaces.
- The potential-based fluid elements can be coupled directly to ADINA-F fluid elements. The ADINA-F fluid element motions cause potential-based fluid flows normal to the ADINA-F boundary, and the potential-based fluid pressures cause additional forces to act on the ADINA-F boundary.
- The potential-based elements can model (approximately) an infinite domain through the use of special infinite elements.
- Two formulations are available, a subsonic velocity formulation, which is nonlinear, and an infinitesimal velocity formulation, which is linear. These are described in detail in the following sections.
- In some analyses, either ADINA-F/ADINA-FSI or the potential-based fluid elements can be used in the modeling. ADINA-F/ADINA-FSI is far more general than the potential-based fluid elements and can model a much wider range of flow conditions. In addition, in ADINA-FSI, the fluid mesh need not be compatible with the structural mesh.

However, for the class of problems in which the assumptions given above are acceptable, the potential-based fluid elements are more efficient. This is because

- ▶ The number of degrees of freedom in the fluid region is less for the potential-based formulation. In 3-D analysis, each ADINA-F node requires a minimum of 4 degrees of freedom, whereas each node in the interior of a potential-based mesh requires only one degree of freedom.
- ▶ If the velocities are small, then the linear infinitesimal velocity formulation can be used. ADINA-F is always nonlinear.
- ▶ In addition, frequency analysis is not possible in ADINA-F.

- The potential-based element formulation has been extensively revised in version 8.0. Some of the features of earlier versions of ADINA are superseded by new features of ADINA 8.0. Features of earlier versions of ADINA that are superseded by new features of ADINA 8.0 include:

The P_0 degree of freedom (superseded in ADINA 7.4)
 Volume infinite elements (superseded in ADINA 8.0)

These features are retained in ADINA 8.0 so that models developed for earlier versions of ADINA still work in ADINA 8.0. However these features are not described here, and are not recommended for new models.

2.11.1 Theory: Subsonic velocity formulation

- Figure 2.11-1 shows a fluid region with volume and bounding surface. In the fluid, we use the basic equations of continuity and energy/momentum, as written in terms of the velocity potential:

$$\dot{\rho} + \nabla \cdot (\rho \nabla \phi) = 0 \quad (2.11-1)$$

and

$$h = \Omega(\mathbf{x}) - \dot{\phi} - \frac{1}{2} \nabla \phi \cdot \nabla \phi \quad (2.11-2)$$

where ρ is the density, ϕ is the velocity potential ($\mathbf{v} = \nabla \phi$ where \mathbf{v} is the fluid velocity), h is the specific enthalpy (defined as $h = \int \frac{dp}{\rho}$), p is the pressure and $\Omega(\mathbf{x})$ is the potential of the (conservative) body force accelerations at position \mathbf{x} . For example, when the body forces are due to gravity, $\nabla \Omega = \mathbf{g}$, where \mathbf{g} is the acceleration due to gravity.

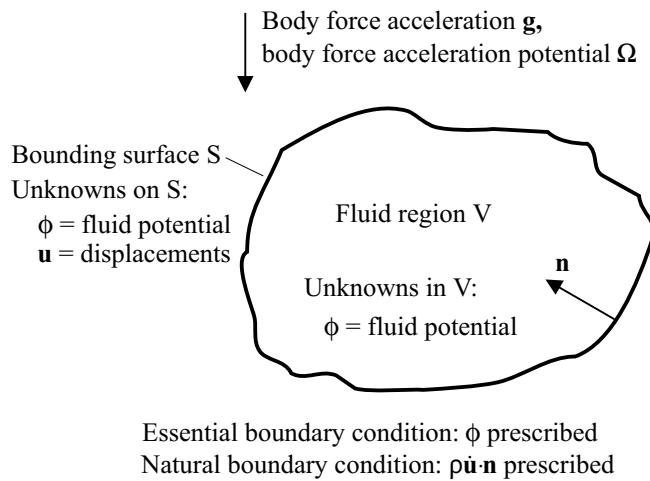


Figure 2.11-1: Fluid region

Equations (2.11-1) and (2.11-2) are valid for an inviscid irrotational fluid with no heat transfer. In particular, (2.11-2) is valid only when the pressure is a function of the density (and not of, for example, the density and temperature).

For the pressure-density relationship, we use the slightly compressible relationship

$$\frac{\rho}{\rho_0} = 1 + \frac{p}{\kappa} \quad (2.11-3)$$

where κ is the bulk modulus and ρ_0 is the nominal density. (2.11-3) then directly gives the density-enthalpy relationship and the pressure-enthalpy relationships:

$$\rho = \rho_0 \exp\left(\frac{\rho_0 h}{\kappa}\right), p = \kappa \left[\exp\left(\frac{\rho_0 h}{\kappa}\right) - 1 \right] \quad (2.11-4a,b)$$

The continuity equation (2.11-1) is then approximated using the standard Galerkin approach to obtain

$$\delta F_\phi = \int_V (\dot{\rho} + \circlearrowleft(\rho \circ \phi)) \delta \phi dV = 0 \quad (2.11-5)$$

where δ is the “variation of” symbol and δF_ϕ is the “variation in the mass flux rate”. (2.11-5) can be rewritten as

$$\delta F_\phi = \int_V (\dot{\rho} \delta \phi - \rho \nabla \phi \cdot \nabla \delta \phi) dV - \int_S \rho \nabla \phi \cdot \mathbf{n} \delta \phi dS \quad (2.11-6)$$

where S is the boundary of V and \mathbf{n} is the inwards normal on S . From this equation, we observe that the natural boundary condition is $\rho \circ \phi \cdot \mathbf{n}$ prescribed, in other words, prescribed mass flux rate.

At this point, we notice that ρ is a function of the fluid velocity and position through the density-enthalpy relationship (2.11-4a), because the enthalpy is a function of fluid velocity and position. In V , we compute the enthalpy using (2.11.2). Hence, in V , the density is a function only of the fluid potential and position. But on S , we anticipate that part or all of S might be a moving boundary with velocity $\dot{\mathbf{u}}(\mathbf{x})$. We adopt the convention that the fluid velocity \mathbf{v} used in the enthalpy calculation on S is computed in terms of both the boundary velocity $\dot{\mathbf{u}}$ and the internal fluid velocity $\circ \phi$ using

$$\mathbf{v}_n = (\dot{\mathbf{u}} \cdot \mathbf{n}) \mathbf{n}, \quad \mathbf{v}_t = \circ \phi - (\circ \phi \cdot \mathbf{n}) \mathbf{n}, \quad \mathbf{v} = \mathbf{v}_n + \mathbf{v}_t \quad (2.11-7a,b,c)$$

In other words, the fluid velocity normal to the surface is taken to be the velocity of the moving boundary and the remainder of the fluid velocity (which is tangential to the surface) is taken from the fluid potential. We also include the motions of S in the body force potential calculations in the enthalpy. Combining the above considerations, we take

$$h = \Omega(\mathbf{x} + \mathbf{u}) - \dot{\phi} - \frac{1}{2} \mathbf{v}_n \cdot \mathbf{v}_n - \frac{1}{2} \mathbf{v}_t \cdot \mathbf{v}_t \quad (2.11-8)$$

on S .

(2.11-6) can be written as

$$\delta F_\phi = \int_V \left(\frac{\partial \rho}{\partial h} \dot{h} \delta\phi - \rho \nabla \phi \cdot \nabla \delta\phi \right) dV + \int_S -\rho \dot{\mathbf{u}} \cdot \mathbf{n} \delta\phi dS \quad (2.11-9)$$

The surface integral is obtained using (2.11-7).

On S , we assume that the boundary motion $\mathbf{u}(\mathbf{x})$ is small enough so that the volume V and the normal \mathbf{n} can be assumed constant. In other words, we perform all integrations on the undeformed fluid boundary. We discuss the implications of this assumption later.

- From (2.11-9), if no boundary conditions are applied to S , the boundary condition $\rho \dot{\mathbf{u}} \cdot \mathbf{n} = 0$ is implied. This boundary condition corresponds to no fluid flow through the boundary.

From (2.11-9), the natural boundary condition is $\rho \dot{\mathbf{u}} \cdot \mathbf{n} = \text{prescribed}$. We use this boundary condition to apply the structural motions to the fluid domain. We also use this boundary condition to model infinite fluid regions. Both of these cases are discussed in more detail below.

Notice that there is no boundary condition corresponding to flow tangential to the boundary. In other words, fluid can slip tangential to the boundary without restriction.

Also from (2.11-9), the essential boundary condition $\phi = \text{prescribed}$ is possible. Through (2.11-2) this boundary condition corresponds to a partially prescribed enthalpy and hence a partially prescribed pressure. The enthalpy and pressure are not fully prescribed since $\circ\phi$ is not prescribed when ϕ is prescribed only on the boundary.

Modifications to equations of motion for the structure

- We assume that part of the boundary S is adjacent to the structure (Figure 2.11-2). The part of the boundary adjacent to the structure is denoted S_1 .

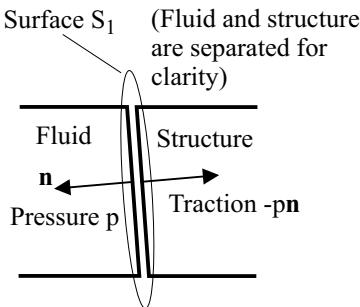


Figure 2.11-2: Forces on structure from fluid

The fluid pressure on S_1 provides additional forces on the structure adjacent to S_1 :

$$-\delta F_u = - \int_{S_1} p \mathbf{n} \cdot \delta \mathbf{u} dS_1 \quad (2.11-10)$$

where δF_u is the variation in the applied force vector and the minus signs are used in anticipation of F_u being considered an internal force vector in (2.11.12) below. The pressure is evaluated using

$$p = p(h) = p \left(\Omega(\mathbf{x} + \mathbf{u}) - \dot{\phi} - \frac{1}{2} \mathbf{v}_n \cdot \mathbf{v}_n - \frac{1}{2} \mathbf{v}_t \cdot \mathbf{v}_t \right) \quad (2.11-11)$$

We emphasize that \mathbf{n} points into the fluid (and out of the structure).

Finite element equations of motion

- Equations (2.11-6) and (2.11-9) are linearized using standard procedures to obtain

$$\begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\mathbf{M}_{FF} \end{bmatrix} \begin{bmatrix} \Delta \ddot{\mathbf{u}} \\ \Delta \ddot{\phi} \end{bmatrix} + \begin{bmatrix} \mathbf{C}_{UU} & \mathbf{C}_{UF} \\ \mathbf{C}_{FU} & -(\mathbf{C}_{FF} + (\mathbf{C}_{FF})_S) \end{bmatrix} \begin{bmatrix} \Delta \dot{\mathbf{u}} \\ \Delta \dot{\phi} \end{bmatrix} + \begin{bmatrix} \mathbf{K}_{UU} & \mathbf{K}_{UF} \\ \mathbf{K}_{FU} & -(\mathbf{K}_{FF} + (\mathbf{K}_{FF})_S) \end{bmatrix} \begin{bmatrix} \Delta \mathbf{u} \\ \Delta \phi \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix} - \begin{bmatrix} \mathbf{F}_U \\ \mathbf{F}_F + (\mathbf{F}_F)_S \end{bmatrix} \quad (2.11-12)$$

where the increment in the vector of unknown potentials ϕ is written $\Delta\phi$ and the increment in the vector of unknown displacements \mathbf{u} is written $\Delta\mathbf{u}$. (We drop the left superscript $t + \Delta t$ here and below for ease of writing.) In the linearization process, increments in both the nodal displacements and increments in the nodal potentials (and their time derivatives) are considered. In (2.11-12),

\mathbf{F}_U = vector from (2.11-10)

\mathbf{F}_F = vector from volume integration term in (2.11-9)

$(\mathbf{F}_F)_S$ = vector from surface integration term in (2.11-9)

and the matrices are obtained by linearization. Note that vectors and matrices with the subscript S are integrated over the surface. Also the matrices with displacement degrees of freedom are integrated over the surface.

The sum of \mathbf{F}_F and $(\mathbf{F}_F)_S$ can be interpreted as an “out-of-balance” mass flux vector. (2.11-12) is satisfied only if this sum is zero at every node in the fluid, that is, if the consistent mass fluxes at each of the nodes from all of the attached fluid elements sum to zero.

In (2.11-12), we do not include any of the structural system matrices. (2.11-12) only gives the contribution of the potential-based fluid elements to the system matrices. Of course, the rest of the structure will make additional contributions to the first row of the above equations.

- Equation (2.11-12) is a nonlinear system of equations. The nonlinearities come from several sources:
 - 1) The $-\frac{1}{2} \nabla \phi \cdot \nabla \phi$ term in h . This term is sometimes called the Bernoulli effect.
 - 2) The nonlinear relationship between the density, pressure and the enthalpy (2.11-4).

Because the system is nonlinear, equilibrium iterations must be employed for an accurate solution, just as in nonlinear structural analysis.

- There is no explicit restriction on the magnitude of the fluid velocities in equation (2.11-12). However, in practice, there are several restrictions
 - 1) The velocity must be smaller than the speed of sound, otherwise, the equations (2.11-9) become hyperbolic and cannot be solved using ordinary finite element techniques.
 - 2) The density change should not be too great, otherwise the change in mass flux may be quite different than the change in volume flux when the boundary velocities are perturbed.
 - 3) Also, the pressure-density relationship in (2.11-3) only holds for relatively small density changes.

With these restrictions, the Mach number of the fluid flow should not exceed about 0.3 or so.

- Equation (2.11-12) is in general nonsymmetric. But in the limit of very small velocities, equation (2.11-12) becomes symmetric (see equation (2.11-19) below). Even for finite velocities, equation (2.11-12) is “nearly” symmetric. This means that we can symmetrize (2.11-12) and use the usual symmetric equation solvers of ADINA for the solution of (2.11-12). The convergence rate is very fast for small velocities and becomes slower as the velocities become larger.

2.11.2 Theory: Infinitesimal velocity formulation

- If we assume that the velocities and the density changes are infinitesimally small, then the continuity equation (2.11-1) becomes

$$\dot{\rho} + \nabla \cdot (\rho \nabla \phi) \approx \dot{\rho} + \rho_0 \nabla^2 \phi \approx \frac{\rho_0 \dot{p}}{\kappa} + \rho_0 \nabla^2 \phi = 0 \quad (2.11-13)$$

The momentum/equilibrium equation (2.11-2) becomes

$$h \approx \frac{p}{\rho} \approx \Omega(\mathbf{x}) - \dot{\phi} \quad (2.11-14)$$

from which we see

$$p \approx \rho(\Omega(\mathbf{x}) - \dot{\phi}) \approx \rho_0(\Omega(\mathbf{x}) - \dot{\phi}) \quad (2.11-15)$$

Substituting (2.11-15) into (2.11-13) gives

$$-\rho_0 \ddot{\phi} + \kappa \nabla^2 \phi = -\rho_0 \dot{\Omega} \quad (2.11-16)$$

Equation (2.11-16) is a special form of the wave equation. It is linear in the solution variable ϕ . (2.11-16) can be written in variational form using standard techniques. The result is

$$\begin{aligned} & -\int_V \rho_0 \ddot{\phi} \delta\phi dV - \int_V \kappa \nabla \phi \cdot \delta \nabla \phi dV - \int_S \kappa \dot{\mathbf{u}} \cdot \mathbf{n} \delta\phi dS \\ &= -\int_V \rho_0 \dot{\Omega} \delta\phi dV \end{aligned} \quad (2.11-17)$$

The fluid pressure onto the structure becomes

$$\delta F_u = \int_{S_l} p \mathbf{n} \cdot \delta \mathbf{u} dS_l \approx \int_{S_l} \left(\rho_0 \Omega + \rho_0 \frac{\partial \Omega}{\partial \mathbf{x}} \cdot \mathbf{u} - \rho_0 \dot{\phi} \right) \mathbf{n} \cdot \delta \mathbf{u} dS_l \quad (2.11-18)$$

The finite element contributions to the system matrices corresponding to (2.11-17) and (2.11-18) are

$$\begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\mathbf{M}_{FF} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{U}} \\ \ddot{\phi} \end{bmatrix} + \begin{bmatrix} \mathbf{0} & \mathbf{C}_{FU}^T \\ \mathbf{C}_{FU} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{U}} \\ \dot{\phi} \end{bmatrix} + \begin{bmatrix} (\mathbf{K}_{UU})_S & \mathbf{0} \\ \mathbf{0} & -\mathbf{K}_{FF} \end{bmatrix} \begin{bmatrix} \mathbf{U} \\ \phi \end{bmatrix} = \begin{bmatrix} (\mathbf{R}_{UB})_S \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ -\dot{\mathbf{R}}_{FB} \end{bmatrix} \quad (2.11-19)$$

where

\mathbf{M}_{FF} = matrix from $\ddot{\phi} \delta\phi$ term in (2.11-17)

\mathbf{K}_{FF} = matrix from $\nabla\phi \cdot \delta\nabla\phi$ term in (2.11-17)

\mathbf{C}_{FU} = matrix from $\dot{\mathbf{u}} \cdot \mathbf{n} \delta\phi$ term in (2.11-17)

$(\mathbf{K}_{UU})_S$ = matrix from $\left(\rho_0 \frac{\partial \Omega}{\partial \mathbf{x}} \cdot \mathbf{u} \right) \mathbf{n} \cdot \delta \mathbf{u}$ term in (2.11-18)

$(\mathbf{R}_{UB})_S$ = loads vector from $(\rho_0 \Omega) \mathbf{n} \cdot \delta \mathbf{u}$ term in (2.11-18)

$\dot{\mathbf{R}}_{FB}$ = loads vector from $\rho_0 \dot{\Omega} \delta\phi$ term in (2.11-17)

\mathbf{U} = vector containing unknown nodal displacements

ϕ = vector containing unknown nodal fluid potentials.

We note that the term $(\mathbf{K}_{UU})_S$ is numerically very small compared with the rest of the structural stiffness matrix, when there is a structure adjacent to the fluid. But $(\mathbf{K}_{UU})_S$ is important in the case when there is no structure adjacent to the fluid.

- The left-hand-side of equation (2.11-19), with the exception of the term $(\mathbf{K}_{UU})_S$, is identical to the formulation presented in the following reference:

ref. L.G. Olson and K.J. Bathe, "Analysis of fluid-structure interactions. A direct symmetric coupled formulation based on the fluid velocity potential", *J. Computers and Structures*, Vol 21, No. 1/2, pp 21-32, 1985.

2.11.3 Theory: Infinite fluid regions

- Both the subsonic and infinitesimal velocity formulations

include special boundary conditions for the modeling of infinite fluid regions.

- The basic approach is to replace the infinite fluid region with a boundary condition that simulates the infinite fluid region. The boundary condition allows outwards-going waves to be propagated into the infinite fluid region without reflection.
- There are many approaches for setting up infinite elements. One approach that is physically intuitive is to use the results from acoustic analysis regarding outwards-going waves.

Planar waves: To illustrate the procedure, we consider a planar boundary, that is, a boundary on which plane acoustic waves propagate outwards through the boundary (Figure 2.11-3). If the waves have very small amplitude,

$$\Delta p = \rho c \Delta v \quad (2.11-20)$$

where Δp is the change in pressure applied to the boundary from the fluid on the outside of the boundary (in other words, the change in pressure applied to the boundary from the fluid that is not explicitly modeled), Δv is the change in the outwards velocity of the fluid and c is the speed of sound in the fluid. Hence the pressure on the boundary is

$$p = p_\infty + \rho c (v - v_\infty) \quad (2.11.21)$$

where v is the outwards velocity at the boundary, and p_∞ and v_∞ are the pressure and outwards velocity at infinity (p_∞ and v_∞ must be specified as part of the model definition).

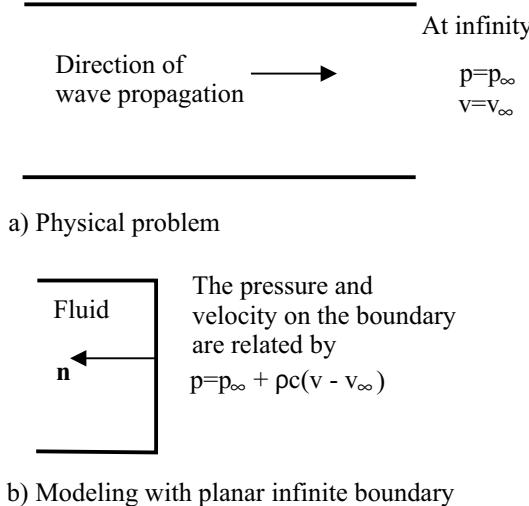


Figure 2.11-3: Planar infinite boundary

The variation in mass flow through the boundary is

$$\delta F_{\phi B} = - \int_{S_B} \rho \nabla \phi \cdot \mathbf{n} \delta \phi dS = \int_{S_B} \rho v \delta \phi dS \quad (2.11-22)$$

We need to express the mass flow through the boundary in terms of the potential on the boundary. This can be done as follows. First, we assume that there is no fluid flow tangential to the boundary and that the boundary does not move. Under those conditions,

$$p(h) = p(\Omega - \dot{\phi} - \frac{1}{2} v^2) \quad (2.11-23)$$

Given a value of $\dot{\phi}$, v can be numerically evaluated by combining equations (2.11-21) and (2.11-23) and hence, given a value of $\dot{\phi}$, (2.11-22) can be numerically evaluated. We use this procedure in the implementation of the planar infinite element for the subsonic velocity formulation.

As an illustrative example, in the special case when there are no body forces, the velocities are small, and the velocities and pressure

at infinity are zero

$$\rho v = \frac{p}{c} = \frac{-\rho \dot{\phi}}{c} \quad (2.11-24)$$

so that the variation in mass flow rate is, in this special case,

$$\delta F_{\phi B} = - \int_{S_B} \frac{\rho \dot{\phi}}{c} \delta \phi dS \quad (2.11-25)$$

We use (2.11-25) in the implementation of the planar infinite element for the infinitesimal velocity formulation.

Spherical waves: The same technique can be used for spherical waves, provided that the appropriate relationship is used for the relationship between p and ρv . The relationship that we use for spherical waves is based on the relationship between pressure and mass flux in the frequency domain

$$\rho v = \frac{p}{c} + \frac{p}{i\omega r} \quad (2.11-26)$$

where $i = \sqrt{-1}$, ω is the frequency of the outwards-going wave and r is the radius of the boundary. Here we assume that both the pressure and velocity at infinity are equal to zero and that the velocity itself is small. The equivalent of (2.11-26) in the time domain is

$$\rho v = \frac{p}{c} + \frac{\int p dt}{r} \quad (2.11-27)$$

We now neglect body force effects, so that $p = -\rho \dot{\phi}$ on the boundary. Then

$$\rho v = \frac{-\rho \dot{\phi}}{c} - \frac{\rho \phi}{r} \quad (2.11-28)$$

so that the variation in mass flow rate is

$$\delta F_{\phi B} = - \int_{S_B} \left(\frac{\rho \dot{\phi}}{c} + \frac{\rho \phi}{r} \right) \delta \phi dS \quad (2.11-29)$$

Note that this derivation holds regardless of the wave frequency ω

Cylindrical waves: The same technique and derivation can be used for cylindrical waves. For cylindrical waves, the relationship between pressure and mass flux in the frequency domain is

$$\rho v = \frac{p}{ic} \left[\frac{J_1(kr) - iY_1(kr)}{J_0(kr) - iY_0(kr)} \right] \quad (2.11-30)$$

where J_0, J_1, Y_0, Y_1 are the Bessel's functions of the first and second kinds of orders 0 and 1, and $kr = \frac{\omega r}{c}$. (2.11-30) cannot be transformed into the time domain, but for frequencies higher than about $\frac{\omega r}{c} = 1$, it turns out that (2.11-30) is well approximated by (2.11-26), so that the spherical wave derivation can be used, provided that the frequency is high enough.

- For planar, spherical and cylindrical infinite boundaries, only surface integrations are required. Also, no additional degrees of freedom are required on the boundary. For planar waves, the infinite boundaries contribute to $(\mathbf{F}_F)_S$ $(\mathbf{C}_{FF})_S$ and for spherical and cylindrical waves, the infinite boundaries contribute to $(\mathbf{F}_F)_S$, $(\mathbf{C}_{FF})_S$ and $(\mathbf{K}_{FF})_S$.
- It is instructive to compare the above approach with the doubly-asymptotic approach in Olson and Bathe.

ref. L.G. Olson and K.J. Bathe, "An infinite element for analysis of transient fluid-structure interactions", *Engineering Computations*, Vol 2, pp 319-329, 1985.

In the DAA, the infinite fluid region is modeled as the superposition of two effects, an added mass approximation and a plane wave approximation. The added mass approximation dominates for low frequencies and the plane wave approximation dominates for high frequencies. The plane wave approximation gives exactly the results given above for planar waves (equation 2.11-25). But the added mass approximation in the DAA requires a volume integral and hence volume infinite elements are required for modeling non-planar waves. Provided that the nodes in the volume infinite elements “at infinity” are placed at the correct locations, the volume infinite elements give exactly the same results as the spherical infinite elements (equation (2.11-29)).

Since the surface integral formulation given in equations (2.11-20) to (2.11-30) gives the same results as the DAA approach, and since the surface integral formulation does not require volume elements, we feel that the surface integral formulation is preferable.

2.11.4 Theory: Ground motion loadings

- Ground motion loadings require special treatment. In this case,

$$\mathbf{U} = \mathbf{U}_r + \mathbf{U}_g \quad (2.11-31)$$

where \mathbf{U}_g is the vector of nodal point ground displacements and \mathbf{U}_r is the vector of nodal point displacements relative to the ground motion. The ground motions are expressed as

$$\mathbf{U}_g = \sum_k u_{gk} \mathbf{d}_k \quad (2.11-32)$$

where u_{gk} are the ground displacements in direction k and \mathbf{d}_k is the vector of nodal point values in which $\mathbf{d}_k(i) = 1$ if equation i corresponds to a translation in direction k and $\mathbf{d}_k(i) = 0$ otherwise.

Because the ground motions are known, the increment in displacements $\Delta\mathbf{U}$ is equal to the increment in the relative displacements $\Delta\mathbf{U}_r$. Hence the left hand sides of equations (2.11-12) and (2.11-19) are unchanged and the right-hand-sides of these

equations are updated only by additional internal forces. For example, the additional internal forces added to (2.11-19) from the fluid elements are

$$-\begin{bmatrix} (\mathbf{K}_{UU})_S \mathbf{d}_k \\ \mathbf{0} \end{bmatrix} u_{gk} - \begin{bmatrix} \mathbf{0} \\ -\mathbf{C}_{FU} \mathbf{d}_k \end{bmatrix} \dot{u}_{gk}$$

Note that ϕ still contains the nodal point values of ϕ and ϕ is the potential corresponding to the absolute (not relative) velocities.

Also note that ground motions are handled differently than physical body forces. This is quite different than in ordinary structural analysis, in which physical body forces and ground motions are both modeled using mass-proportional loads.

2.11.5 Theory: Static analysis

- The case of static analysis also requires special attention. It is not correct simply to set the velocities to zero in (2.11-1), because (2.11-1) is identically satisfied for zero velocities.
- We envision the process of obtaining a static solution as a quasi-static process. In this process, all fluid velocities are assumed negligible and therefore the fluid potential is constant (in space). Under these conditions,

$$\dot{h} = \Omega - \dot{\phi} \quad (2.11-32)$$

and the variational statement of continuity is

$$\delta F_\phi = \int_V \dot{\rho} \delta \phi dV - \int_S \rho \dot{\mathbf{u}} \cdot \mathbf{n} \delta \phi dS \quad (2.11-33)$$

Equation (2.11-33) is simply the integral equation of conservation of mass because $\delta \phi$ is constant in space.

- We now numerically integrate (2.11-33) in time. We use the Euler backwards approximations

$${}^{t+\Delta t} \dot{\rho} = \left({}^{t+\Delta t} \rho - {}^t \rho \right) / \Delta t, \quad {}^{t+\Delta t} \dot{\mathbf{u}} = \left({}^{t+\Delta t} \mathbf{u} - {}^t \mathbf{u} \right) / \Delta t \quad (2.11-34a,b)$$

and then obtain, at time $t + \Delta t$,

$$\delta F_\phi = \int_V \left({}^{t+\Delta t} \rho - {}^t \rho \right) \delta \dot{\phi} dV - \int_S {}^{t+\Delta t} \rho \left({}^{t+\Delta t} \mathbf{u} - {}^t \mathbf{u} \right) \cdot \mathbf{n} \delta \dot{\phi} dS \quad (2.11-35)$$

where we write $\delta \dot{\phi} = \delta \phi / \Delta t$.

The modification to the structural equations of motion is

$$\delta F_u = \int_{S_1} {}^{t+\Delta t} p \mathbf{n} \cdot \delta \mathbf{u} dS_1 \quad (2.11-36)$$

which is the same as equation (2.11-10), except that the $t + \Delta t$ is explicitly written.

We prefer the Euler backwards approximations because then the stiffness matrix remains symmetric when the displacement increments are infinitesimally small. Of course, the steps must be small enough so that the Euler backwards approximations apply, but in many problems, these approximations are quite good, especially when the fluid is almost incompressible.

Note that we assume that the domain of integration and its boundary remain (nearly) unchanged during the time integration. All integrations are performed over the original domain of integration.

Also note that the solution in the fluid is the single value $\dot{\phi}$. This value represents the Bernoulli constant.

The linearized equations of motion for the subsonic formulation become

$$\begin{bmatrix} (\tilde{\mathbf{K}}_{UU})_S & \tilde{\mathbf{K}}_{UF} \\ \tilde{\mathbf{K}}_{FU} & -(\tilde{\mathbf{K}}_{FF} + (\tilde{\mathbf{K}}_{FF})_S) \end{bmatrix} \begin{bmatrix} \Delta \mathbf{u} \\ \Delta \dot{\phi} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix} - \begin{bmatrix} \tilde{\mathbf{F}}_U \\ \tilde{\mathbf{F}}_F + (\tilde{\mathbf{F}}_F)_S \end{bmatrix} \quad (2.11-37)$$

where we use the \sim to emphasize that the vectors and matrices are different than the corresponding dynamic vectors and matrices, along with the constraints $\dot{\phi} = \text{constant}$ (if there is more than one fluid region in the problem, then $\dot{\phi} = \text{constant}$ in each fluid

region). As in the dynamic case (equation 2.11-12), equation (2.11-37) is in general non-symmetric, but (2.11-37) becomes symmetric when the displacement increment ${}^{t+\Delta t}\mathbf{u} - {}^t\mathbf{u}$ becomes infinitesimally small. So, as in the dynamic case, we symmetrize the system and use the usual symmetric equation solver.

In (2.11-37) the sum of $\tilde{\mathbf{F}}_F + (\tilde{\mathbf{F}}_F)_S$ can be interpreted as an “out-of-balance” mass vector. (2.11-37) is only satisfied if the total mass of each fluid region is conserved. Motions of the boundary that do not change the total mass of any fluid region are associated with zero pivots in the system matrices, unless there is structural stiffness associated with these motions. Note that the $(\tilde{\mathbf{K}}_{UU})_S$ matrix also provides structural stiffness, so if there are body forces, motions of the boundary that do not change the total mass of any fluid region are not associated with zero pivots.

The equations of motion for the infinitesimal velocity formulation can be obtained either from the above derivation, or can be formally derived from (2.11-19) by applying the Laplace transform to both sides of (2.11-19) and applying the final value theorem. The result is

$$\begin{bmatrix} (\mathbf{K}_{UU})_S & \mathbf{C}_{FU}^T \\ \mathbf{C}_{FU} & -\mathbf{M}_{FF} \end{bmatrix} \begin{bmatrix} \mathbf{U} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} (\mathbf{R}_{UB})_S \\ -\mathbf{R}_{FB} \end{bmatrix} \quad (2.11-38)$$

together with the condition

$$\mathbf{K}_{FF}\phi = \mathbf{0} \quad (2.11-39)$$

- There are a number of unusual characteristics of (2.11-37), (2.11-38) and (2.11-39):
 - ▶ The solution involves $\dot{\phi}$ instead of ϕ . This makes sense as (2.11-32) then implies that p is constant (in time) in a static solution.
 - ▶ The condition (2.11-39) must be satisfied. When there are no infinite boundaries, this condition is satisfied whenever

$\dot{\phi} = \text{constant}$ within each separate fluid region. Hence the number of unknown potential degrees of freedom in static analysis is equal to the number of separate fluid regions in the analysis.

- ▶ The condition $\dot{\phi} = \text{constant}$ within each separate fluid region implies that $p = \rho_0\Omega + C$ where C is a constant determined from the solution. Hence the variation of pressure within each separate fluid region is contained within C and any choice of constant of integration within $\rho_0\Omega$ (recall that Ω is a potential and therefore includes an arbitrary constant of integration) is balanced by an opposite change in C .
- ▶ If the ϕ degree of freedom is fixed at a node, or if infinite boundaries are included in the fluid region, then you must set $\phi = \mathbf{0}$ for all the nodes in the fluid region. This condition implies $p = \rho_0\Omega$ within the fluid region. Any choice of constant of integration within Ω then affects the solution.
- ▶ It is necessary to enter the density of the fluid in static analysis, even when the solution does not depend on the density.

2.11.6 Theory: Frequency analysis

- Frequency analysis is possible when there is no structural damping, when the infinitesimal velocity formulation is used, and when there are no infinite boundaries. The eigenvalue problem to be solved is

$$\left(-\omega_j^2 \begin{bmatrix} \mathbf{M} & \mathbf{0} \\ \mathbf{0} & \mathbf{M}_{FF} \end{bmatrix} - \omega_j \begin{bmatrix} \mathbf{0} & \mathbf{C}_{FU}^T \\ \mathbf{C}_{FU} & \mathbf{0} \end{bmatrix} + \begin{bmatrix} \mathbf{K} + (\mathbf{K}_{UU})_S & \mathbf{0} \\ \mathbf{0} & \mathbf{K}_{FF} \end{bmatrix} \right) \begin{bmatrix} \mathbf{U}^{(j)} \\ \mathbf{F}^{(j)} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix} \quad (2.11-40)$$

where $\mathbf{F}^{(j)} = -i\dot{\phi}^{(j)}$, $i = \sqrt{-1}$ and in which we also include the structural stiffness matrix \mathbf{K} and structural mass matrix \mathbf{M} .

(2.11-40) is derived from (2.11-19) by taking the Fourier transform of the left-hand-side.

(2.11-40) is a non-standard eigenvalue problem in which all of the eigenvalues are real and non-negative. It is solved using either the determinant search method or the Lanczos iteration method.

The eigenvectors are scaled according to the following orthogonality condition:

$$\begin{aligned} \mathbf{u}^{(i)} \mathbf{C}_{FU}^T \mathbf{F}^{(i)} + \mathbf{F}^{(i)} \mathbf{C}_{FU} \mathbf{u}^{(i)} + (\omega_i + \omega_j) (\mathbf{u}^{(i)} \mathbf{M}_{UU} \mathbf{u}^{(j)} + \mathbf{F}^{(i)} \mathbf{M}_{FF} \mathbf{F}^{(j)}) \\ = 2\sqrt{\omega_i \omega_j} \delta_{ij} \end{aligned} \quad (2.11-41)$$

where δ_{ij} is the Kronecker delta. Notice that when there is no fluid, (2.11-41) reduces to the usual orthogonality condition

$$\mathbf{u}^{(i)} \mathbf{M}_{UU} \mathbf{u}^{(j)} = \delta_{ij}.$$

(2.11-40) has one rigid body mode for each separate fluid region in which all of the ϕ degrees of freedom are free. Each rigid body mode is of the form $\mathbf{U}^{(j)} = 0, \mathbf{F}^{(j)} = \text{constant}$ in each separate fluid region. For future reference, we term these rigid body modes ϕ rigid body modes. The determinant search method can determine all ϕ rigid body modes.

The modal pressures and modal fluid particle displacements are computed using

$$p^{(j)} = \rho \omega_j f^{(j)}, \quad \mathbf{u}^{(j)} = \frac{1}{\omega_j} \nabla f^{(j)} \quad (2.11-42)$$

in which $p^{(j)}$ is the modal pressure at the point of interest for mode j , $f^{(j)}$ is the fluid potential eigenvector $\mathbf{F}^{(j)}$ interpolated to the point of interest and $\mathbf{u}^{(j)}$ is the modal displacement vector at the point of interest for mode j .

2.11.7 Theory: Mode superposition

- Mode superposition is possible when there is no structural

damping, the infinitesimal velocity formulation is used and when there are no infinite boundaries. The theory used to determine the modal equations of motion is given in Chapter 6 of the following reference:

ref. L. Meirovitch, *Computational Methods in Structural Dynamics*, Sijthoff & Noordhoff, 1980.

The derivation starts with the equations of motion (2.11-19) being put into the form

$$\begin{aligned} \begin{bmatrix} \mathbf{M} & \mathbf{0} \\ \mathbf{0} & \mathbf{M}_{FF} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{U}}_r \\ \ddot{\boldsymbol{\phi}} \end{bmatrix} + \begin{bmatrix} \mathbf{0} & \mathbf{C}_{FU}^T \\ -\mathbf{C}_{FU} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{U}}_r \\ \dot{\boldsymbol{\phi}} \end{bmatrix} + \begin{bmatrix} \mathbf{K} + (\mathbf{K}_{UU})_s & \mathbf{0} \\ \mathbf{0} & \mathbf{K}_{FF} \end{bmatrix} \begin{bmatrix} \mathbf{U}_r \\ \boldsymbol{\phi} \end{bmatrix} \\ = \begin{bmatrix} \mathbf{R} \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} (\mathbf{R}_{UB})_s \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \dot{\mathbf{R}}_{FB} \end{bmatrix} - \begin{bmatrix} (\mathbf{K}_{UU})_s \mathbf{d}_k \\ \mathbf{0} \end{bmatrix} u_{gk} + \begin{bmatrix} \mathbf{0} \\ \mathbf{C}_{FU} \mathbf{d}_k \end{bmatrix} \dot{u}_{gk} \end{aligned} \quad (2.11-43)$$

where we include the structural stiffness matrix \mathbf{K} and structural mass matrix \mathbf{M} . We also include the ground motion loading terms in (2.11-43) in anticipation of the response spectrum derivation below. (2.11-43) corresponds to an undamped gyroscopic system. The modal expansion used is

$$\mathbf{U}_r = \sum_{j=1}^n \frac{\xi_j}{\omega_j} \mathbf{U}^{(j)}, \quad \dot{\boldsymbol{\phi}} = \sum_{j=1}^n -\xi_j \mathbf{F}^{(j)} \quad (2.11-44a,b)$$

where ξ_j is the generalized coordinate for mode j . The modal equation is

$$\ddot{\xi}_j + \omega_j^2 \xi_j = I^{(j)} \quad (2.11-45)$$

where

$$I^{(j)} = -(\mathbf{F}^{(j)})^T \mathbf{R}_\phi + \omega_j (\mathbf{U}^{(j)})^T \mathbf{R}_u \quad (2.11-46)$$

and, from (2.11-43),

$$\begin{aligned}\mathbf{R}_u &= \mathbf{R} + (\mathbf{R}_{UB})_S - (\mathbf{K}_{UU})_S \mathbf{d}_k u_{gk} \\ \mathbf{R}_\phi &= \mathbf{C}_{FU} \mathbf{d}_k \dot{u}_{gk}\end{aligned}\quad (2.11-47a,b)$$

With modal damping (damping ratio ζ_j), (2.11-45) becomes

$$\ddot{\xi}_j + 2\omega_j \zeta_j \dot{\xi}_j + \omega_j^2 \xi_j = \Gamma^{(j)} \quad (2.11-48)$$

Then we use

$$\begin{aligned}\mathbf{U}_r &= \sum_{j=1}^n \frac{\xi_j}{\omega_j} \mathbf{U}^{(j)}, \quad \dot{\mathbf{U}}_r = \sum_{j=1}^n \frac{\dot{\xi}_j}{\omega_j} \mathbf{U}^{(j)}, \quad \ddot{\mathbf{U}}_r = \sum_{j=1}^n \frac{\ddot{\xi}_j}{\omega_j} \mathbf{U}^{(j)} \\ &\quad (2.11-49a,b,c)\end{aligned}$$

$$\begin{aligned}\boldsymbol{\phi} &= \sum_{j=1}^n \int_0^t -\xi_j dt \mathbf{F}^{(j)}, \quad \dot{\boldsymbol{\phi}} = \sum_{j=1}^n -\xi_j \mathbf{F}^{(j)}, \quad \ddot{\boldsymbol{\phi}} = \sum_{j=1}^n -\dot{\xi}_j \mathbf{F}^{(j)} \\ &\quad (2.11-50a,b,c)\end{aligned}$$

to obtain the structural and fluid response.

The initial modal coordinates are computed from the initial conditions (displacements, velocities, fluid potentials and time derivatives of fluid potentials). Initial accelerations and initial second time derivatives of fluid potentials are not used.

In the first solution step, we use the Newmark method with $\alpha = 1/2$, $\delta = 1$ because this choice of Newmark parameters does not require initial accelerations. In the successive solution steps, we use the Newmark method with the usual Newmark parameters $\alpha = 1/4$, $\delta = 1/2$.

2.11.8 Theory: Response spectrum, harmonic and random vibration analysis

- Modal participation factors are calculated for response spectrum analysis, harmonic vibration analysis and random vibration analysis. There are two cases: modal participation factors corresponding to applied forces (in which case we do not consider

response spectrum analysis) and modal participation factors corresponding to ground motions.

In the modal participation factor calculation, we only consider the dynamic solution (due to dynamically applied loads).

To begin the derivation, we introduce

$$x_j = \frac{\xi_j}{\omega_j} \quad (2.11-51)$$

as the new generalized coordinate. The modal expansion is then

$$\mathbf{U}_r = \sum_{j=1}^n x_j \mathbf{U}^{(j)}, \quad \dot{\phi} = \sum_{j=1}^n -\omega_j x_j \mathbf{F}^{(j)} \quad (2.11-52a,b)$$

and the modal equation of motion is

$$\ddot{x}_j + 2\omega_j \zeta_j \dot{x}_j + \omega_j^2 x_j = \bar{F}^{(j)} \quad (2.11-53)$$

where

$$\bar{F}^{(j)} = -\frac{1}{\omega_j} \left(\mathbf{F}^{(j)} \right)^T \mathbf{R}_\phi + \left(\mathbf{U}^{(j)} \right)^T \mathbf{R}_u \quad (2.11-54)$$

We will compute $\bar{F}^{(j)}$ rather than $F^{(j)}$ because the modal expansion for \mathbf{U}_r is then the commonly used one.

Applied forces: As it is assumed that the physical body forces are entirely static and that there are no ground motions, the modal participation factor is simply

$$\bar{F}^{(j)} = \left(\mathbf{U}^{(j)} \right)^T \mathbf{R} \quad (2.11-55)$$

As a consequence, the modal participation factor for any ϕ rigid body mode is zero.

Ground motions: The loads vectors are

$$\begin{aligned}\mathbf{R}_u &= -(\mathbf{K}_{UU})_S \mathbf{d}_k u_{gk} - \mathbf{M} \mathbf{d}_k \ddot{u}_{gk}, \\ \mathbf{R}_\phi &= \mathbf{C}_{FU} \mathbf{d}_k \dot{u}_{gk}\end{aligned}\quad (2.11-56a,b)$$

Hence

$$\begin{aligned}\bar{I}^{(j)} &= -\left(\frac{1}{\omega_j} (\mathbf{F}^{(j)})^T \mathbf{C}_{FU} \mathbf{d}_k + (\mathbf{U}^{(j)})^T \mathbf{M} \mathbf{d}_k \right) \ddot{u}_{gk} \\ &\quad - (\mathbf{U}^{(j)})^T (\mathbf{K}_{UU})_S \mathbf{d}_k u_{gk}\end{aligned}\quad (2.11-57)$$

We neglect the term $(\mathbf{U}^{(j)})^T (\mathbf{K}_{UU})_S \mathbf{d}_k u_{gk}$ (which is probably small anyway) and observe that the ground motion modal participation factor is then

$$\bar{I}^{(j)} = \frac{1}{\omega_j} (\mathbf{F}^{(j)})^T \mathbf{C}_{FU} \mathbf{d}_k + (\mathbf{U}^{(j)})^T \mathbf{M} \mathbf{d}_k \quad (2.11-58)$$

The modal participation factor for a ϕ rigid body mode can be zero or non-zero. It can be shown that the modal participation factor for a ϕ rigid body mode is zero if each fluid region is completely surrounded by interface elements.

Note: static corrections (residual calculations) are not implemented for the potential-based fluid element because residual displacement calculations are based upon non-constant (in space) body force loading. Such loading is not possible in general for the potential-based fluid elements.

2.11.9 Modeling: Formulation choice and potential master degree of freedom

The potential-based formulation discussed in this section is introduced in ADINA 7.4 and extensively modified in ADINA 8.0.

You can select the potential-based formulation in the AUI command-line input using the parameter FLUIDPOTENTIAL in the MASTER command:

FLUIDPOTENTIAL=AUTOMATIC: The potential-based formulation of ADINA 7.4 and 8.0 is employed. This is the default.

FLUIDPOTENTIAL=YES: The potential-based formulation of ADINA 7.3 and lower is employed.

FLUIDPOTENTIAL=NO: The potential-based formulation is not employed.

Note, it is not possible to select the potential-based formulation using the AUI user interfaces. The potential-based formulation of ADINA 8.0 is employed when using the AUI user interfaces.

When FLUIDPOTENTIAL=AUTOMATIC, the AUI automatically detects the presence of potential-based elements, and, if there are any potential-based elements, activates the potential master degree of freedom.

2.11.10 Modeling: Potential degree of freedom fixities

Subsonic formulation: Deleting the potential degree of freedom along part of the bounding surface has the effect of partially specifying the enthalpy along that surface, see equation (2.11-2). This has no physical meaning.

Infinitesimal velocity formulation: Deleting the potential degree of freedom along part of the bounding surface has the effect of setting the pressure equal to $\rho_0\Omega$ along that part of the bounding surface, see equation (2.11-15). If there are no body forces, then $\Omega = 0$ and the pressure is therefore set to zero along that part of the bounding surface. This boundary condition can be used if the displacements of the boundary are not of interest.

It is recommended that the potential degrees of freedom not be deleted anywhere in the fluid.

2.11.11 Modeling: Elements

The volume V of the fluid domain is modeled using two-dimensional or three-dimensional fluid elements. These elements

are analogous to the two-dimensional or three-dimensional solid elements and the nodal point numbering of the fluid elements is the same as the nodal point numbering of the solid elements.

The two-dimensional elements are either planar (unit thickness of fluid assumed) or axisymmetric (1 radian of fluid assumed).

These elements are defined within 2-D and 3-D fluid element groups (in ADINA). Set the formulation of the element group to either “Linear Potential-Based Element” (for the infinitesimal velocity formulation) or “Subsonic Potential-Based Element” (for the subsonic velocity formulation).

The bounding surface S of the fluid domain is modeled with potential-interfaces and/or interface elements, as discussed in detail below.

It is required that each separate fluid domain be modeled with separate fluid element groups. This is because the AUI constrains the potential degrees of freedom of each element group together in static analysis during phi model completion, step 7, see Section 2.11.15.

It is not permitted to have fluid regions of different densities sharing the same potential degrees of freedom. This is because the nodal pressure would be different as computed from the fluid regions connected to the node.

It is recommended that the fluid and adjacent structure *not* share the coincident nodes. This allows the AUI to construct appropriate constraint equations between the fluid and structural degrees of freedom that are most appropriate during phi model completion, see Section 2.11.15.

2.11.12 Modeling: Potential-interfaces

For ease of modeling, you can define potential-interfaces of various types along the surface of the fluid domain. When you generate a data file, the AUI places interface elements along the boundaries specified by the potential-interfaces, except as noted below. ADINA itself does not use the potential-interfaces.

There are several types of potential-interface:

Fluid-structure: Place a fluid-structure potential interface on the boundary between a potential-based fluid and the adjacent structure.

In many cases, the AUI can automatically generate fluid-

structure interface elements along the boundary between the fluid and structure during phi model completion, step 1, see Section 2.11.15. So you typically do not need to define fluid-structure potential-interfaces.

Free surface: Place a free surface potential interface on the boundary where the pressures are to be prescribed and the displacements are desired, for example, on the free surface of a fluid. Surface waves can be approximately modeled in this manner, but note that the displacement of the waves is assumed to be small.

ADINA-F: Place an ADINA-F potential interface on the boundary adjacent to an ADINA-F mesh.

Usually the AUI automatically generates ADINA-F interface elements along the boundary adjacent to an ADINA-F mesh during phi model completion, step 1. So you typically do not need to define ADINA-F potential-interfaces.

Infinite: Place an infinite potential-interface wherever infinite boundary conditions are desired.

There are three types of infinite potential-interface.

Planar: In the subsonic formulation, the pressure and velocity “at infinity” must be specified. In the infinitesimal velocity formulation, the pressure and velocity are assumed to be zero.

Spherical: The radius of the boundary must be specified. The pressure and velocity at infinity are assumed to be zero, and the velocities at the boundary are assumed to be small.

Cylindrical: The radius of the boundary must be specified. The pressure and velocity at infinity are assumed to be zero, and the velocities at the boundary are assumed to be small. This element cannot accurately model low-frequency waves, that is,

$$\text{waves with } \frac{\omega r}{c} < 1.$$

Inlet-outlet: Place an inlet-outlet potential-interface wherever the pressure of the boundary is specified, and where the displacements

of the boundary are not of interest. For example, the outlet of a pipe on which the pressure is known can be modeled using an inlet-outlet potential-interface.

If there is more than one inlet or outlet in the model, each inlet and outlet must have its own inlet-outlet potential-interface.

Fluid-fluid: Place a fluid-fluid potential-interface on the boundary between two potential-based fluid elements of two different element groups.

Note that only one fluid-fluid potential-interface need be defined for each boundary. The AUI generates a fluid-fluid interface element for each of the two elements that share a common boundary during phi model completion, step 1, see Section 2.11.15.

Rigid-wall: Place a rigid-wall potential interface wherever the fluid is not to flow through the boundary.

Note that this boundary condition is modeled in ADINA by the absence of any interface element. Therefore a rigid-wall potential-interface suppresses any automatic generation of interface elements along the boundary of the rigid-wall potential-interface. However the AUI uses the rigid-wall potential interface during phi model completion, step 2, in constructing structural normals, see Section 2.11.15.

2.11.13 Modeling: Interface elements

Interface elements are used on the surface of the fluid domain to specify a boundary condition. In many cases, you do not have to define interface elements. Rather you define potential-interfaces and the AUI then generates the interface elements. However we describe the interface elements here for completeness.

Interface elements are defined within the same element group as the fluid elements themselves. In 2-D analysis, the fluid-structure interface elements are 2 or 3 node line segments, in 3-D analysis, the fluid-structure interface elements are 3 to 9 node area segments. The nodal point numbering conventions are shown in Figure 2.11-4. The AUI automatically reorders all interface elements when generating the ADINA input data file so that the interface element normals point into the fluid.

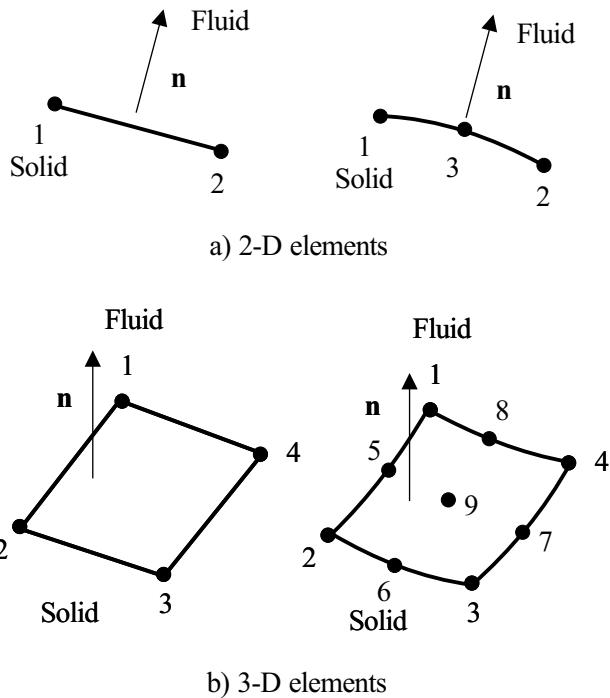


Figure 2.11-4: Fluid-structure interface elements, showing the local node numbering convention

There are several types of interface elements:

Fluid-structure interface element: This element connects the potential-based fluid element with an adjacent structural element (Figure 2.11-5). Each node of the element contains the potential degree of freedom and displacement degrees of freedom. It is assumed that the displacements of the nodes of the interface element are small.

It is assumed that the structure provides stiffness to all translational degrees of freedom, because the fluid-structure interface element does not provide stiffness, mass or damping to the tangential directions.

We emphasize that the potential-based fluid element, fluid-structure interface element and adjacent structural element must all be compatible.

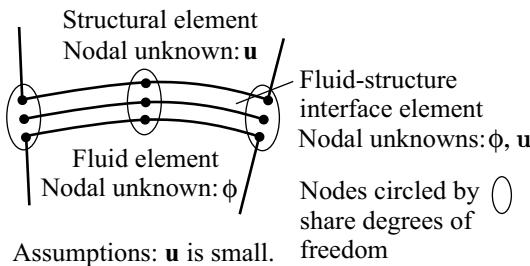


Figure 2.11-5: Fluid-structure interface element

Free surface interface element: This element is placed onto the boundary of a potential-based fluid element wherever the pressure is to be prescribed, and wherever the displacements of the fluid are required (Figure 2.11-6). For example, the free surface of a fluid in a basin can be modeled using free surface interface elements. Each node of the element contains the potential degree of freedom and displacement degrees of freedom. It is assumed that displacements and velocities of the nodes of the interface element are small.

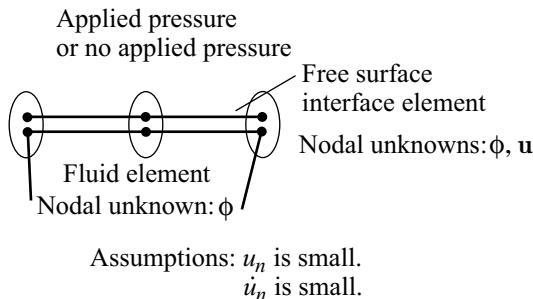


Figure 2.11-6: Free surface interface element

It is necessary to fix all displacements that are tangential to the free surface interface element, because the free surface interface element does not provide stiffness, mass or damping to the tangential directions.

In many cases, the AUI can generate skew systems and fixities corresponding to the tangential directions during phi model generation, see Section 2.11.15.

ADINA-F interface element: This element is placed onto the boundary of a potential-based fluid element wherever the boundary is adjacent to an ADINA-F mesh (Figure 2.11-7). Each node of the element contains the potential degree of freedom and displacement degrees of freedom. It is assumed that displacements and velocities of the nodes of the interface element are small.

Note that the coupling between ADINA and ADINA-F is through the displacements of the shared boundary. Therefore, because the displacements of the ADINA-F interface element are assumed to be small, the ADINA/ADINA-F coupling cannot be used for modeling actual fluid flow between the ADINA and ADINA-F models.

It is necessary to fix all displacements that are tangential to the ADINA-F interface element, because the ADINA-F interface element does not provide stiffness, mass or damping to the tangential directions. In many cases, the AUI can generate these fixities automatically during phi model completion, see Section 2.11.15.

The ADINA-F interface element need not be compatible with the elements from the adjacent ADINA-F mesh .

It is necessary to define an ADINA-F fluid-structure boundary in the ADINA model to connect the ADINA and ADINA-F models, just as in ordinary ADINA-F FSI analysis.

In most cases, the AUI automatically generates the ADINA-F interface elements during phi model completion, step 1, see Section 2.11.15.

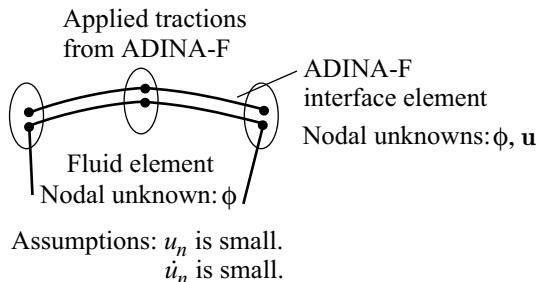


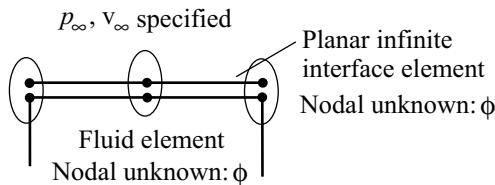
Figure 2.11-7: ADINA-F interface element

Infinite interface elements: This element is placed onto the boundary of a potential-based fluid element wherever infinite boundary conditions are desired. Each node of the element

contains only a potential degree of freedom.

There are three types of infinite interface element:

Planar infinite element: In the subsonic formulation, the pressure and velocity “at infinity” must be specified. In the infinitesimal velocity formulation, the pressure and velocity are assumed to be zero. A planar infinite element is shown in Figure 2.11-8.



Assumptions: ϕ constant along boundary.

Flow velocity is close to v_∞

Figure 2.11-8: Planar infinite interface element

Spherical infinite element: The radius of the boundary must be specified. The pressure and velocity at infinity are assumed to be zero, and the velocities at the boundary are assumed to be small.

Cylindrical infinite element: The radius of the boundary must be specified. The pressure and velocity at infinity are assumed to be zero, and the velocities at the boundary are assumed to be small. This element cannot accurately model low-frequency

waves, that is, waves with $\frac{\omega r}{c} < 1$.

When considering where to place the infinite interface elements, remember that the infinite elements are derived under the assumption that waves travel normal to the boundary. Hence the boundary must be placed so that any anticipated waves travel normal to the boundary (Figure 2.11-9).

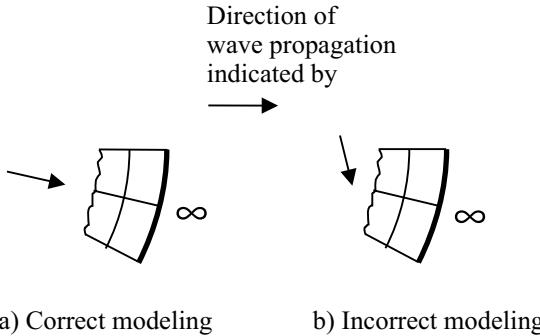
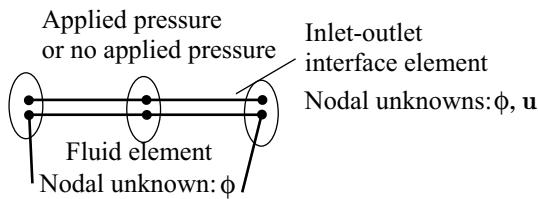


Figure 2.11-9: Direction of wave propagation must be normal to the infinite interface element

Inlet-outlet interface element: This element is placed onto the boundary of a potential-based fluid element wherever the pressure of the boundary is specified, and where the displacements of the boundary are not of interest (Figure 2.11-10). For example, the outlet of a pipe on which the pressure is known can be modeled using inlet-outlet interface elements. Each node of the element contains the potential degree of freedom and displacement degrees of freedom. The displacement degrees of freedom are only used to compute velocities and accelerations; the displacements themselves are not used.



Assumptions: \dot{u}_n constant along boundary.
 ϕ constant along boundary.

Figure 2.11-10: Inlet-outlet interface element

It is necessary to fix all displacements that are tangential to the inlet-outlet interface element, because the inlet-outlet interface element does not provide stiffness, mass or damping to the tangential directions. In addition, it is necessary to set the tangential fluid velocity to zero by constraining all of the potential

degrees of freedom to be equal. Finally it is necessary to set the normal velocity to be uniform along the element, by constraining the normal velocities to be equal. In many cases, the AUI can generate these fixities and constraints automatically during phi model completion, see Section 2.11.15.

Fluid-fluid interface element: This element is placed along the interface between two potential-based fluid elements of two different element groups (Figure 2.11-11). For example, the interface between air and water can be modeled using fluid-fluid interface elements.

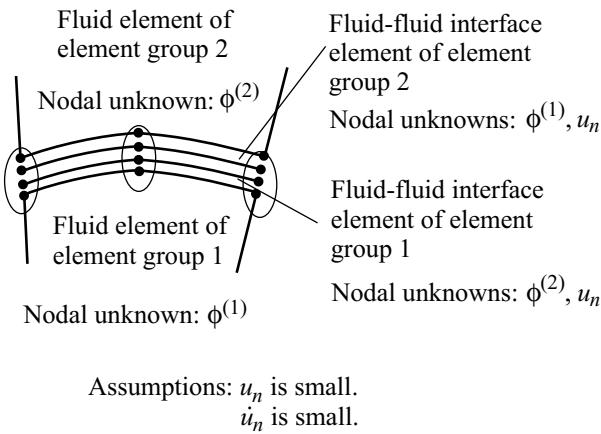


Figure 2.11-11: Fluid-fluid interface element

Each node of the interface element contains the potential degree of freedom and displacement degrees of freedom. The normal velocities and displacements are assumed to be small. It is necessary to fix all displacements that are tangential to the fluid-fluid interface element, because the fluid-fluid interface element does not provide stiffness, mass or damping to the tangential directions.

Each of the potential-based elements that share a common boundary requires a fluid-fluid interface element. The two fluids are connected by constraining the normal displacements of one of the interface elements to the corresponding normal displacements of the other interface element.

In many cases, the AUI can generate these fixities and constraints automatically during phi model completion, see Section 2.11.15.

2.11.14 Modeling: Loads

Concentrated forces, pressure loads, prescribed displacements

Concentrated forces, pressure loads and/or prescribed displacements can be applied directly to any part of the fluid boundary on which there are fluid-structure, free surface, inlet-outlet or fluid-fluid interface elements. However, when applying concentrated forces, remember that the AUI can apply skew systems to certain nodes on the fluid boundary during phi model completion, see Section 2.11.15. Therefore you should make sure that the nodes on which you apply concentrated forces have the degree of freedom directions that you anticipate.

Mass-proportional loads

Mass-proportional loads can be applied. However the AUI and ADINA make a distinction between those mass-proportional loads used to model physical body forces and those mass-proportional loads used to enter ground accelerations. For each mass-proportional load, you must specify its interpretation: body force or ground acceleration.

Body force: Mass-proportional loads interpreted as physical body forces must be constant in time when there are infinitesimal velocity potential-based fluid elements in the model. These loads are used in the construction of g_k and Ω as follows:

$$g_k = \text{MAGNITUDE} \times A(k) \times \text{TF}$$

where MAGNITUDE is the magnitude of the mass-proportional loading, $A(k)$ is the vector giving the direction of the mass-proportional loading and TF is the value of the time function. Then

$$\Omega = \sum_{k=1}^3 g_k (x_k - x_{k0}) \quad (2.11-59)$$

where x_{k0} is a datum value (entered as part of the fluid material description). Notice that the choice of the datum value affects the solution only when infinite interface elements are present, or if at least one potential degree of freedom is fixed in the fluid region.

Body forces can be applied in a static analysis as the only loads in the analysis. Then, if other time-varying loads are present, a restart to dynamic analysis can be performed. The body forces should be kept in the restart dynamic analysis.

Ground acceleration: Mass-proportional loads interpreted as ground motions can be time-varying, and are used in the construction of u_{gk} as follows: Suppose that you specify a mass-proportional load interpreted as ground motions of magnitude $g_k(t)$, where $g_k(t)$ is entered as for body force loads (but the time function need not be constant). Then ADINA computes the ground motions using

$$\ddot{u}_{gk} = -g_k, \quad \dot{u}_{gk} = \int_{tstart}^t \ddot{u}_{gk} dt, \quad u_{gk} = \int_{tstart}^t \dot{u}_{gk} dt \quad (2.11-60a,b,c)$$

Note that it is assumed that $\dot{u}_{gk}(tstart) = 0$, $u_{gk}(tstart) = 0$.

The choice of interpretation for the mass-proportional loads does not affect the structural elements used in the model.

Centrifugal loads

Centrifugal load effects are not included in the potential-based fluid elements.

Mass flux loads (phiflux loads)

Mass flux loads (also referred to as phiflux loads) can be prescribed directly onto potential-based fluid elements. The mass

fluxes can be distributed or concentrated. The dimensions of distributed mass flux are $\frac{[\text{mass}]}{[\text{time}] \times [\text{area}]}$ in dynamic analysis and are $\frac{[\text{mass}]}{[\text{area}]}$ in static analysis). The dimensions of concentrated mass flux are $\frac{[\text{mass}]}{[\text{time}]}$ in dynamic analysis and are $[\text{mass}]$ in static analysis.

Positive mass flux is assumed to represent mass flowing into the fluid domain.

No interface elements or potential-interfaces should be defined on boundaries with distributed or concentrated mass fluxes.

2.11.15 Modeling: Phi model completion

As can be seen above, there are many restrictions and conditions that must be considered when specifying boundary conditions on potential-based fluid elements. Many of these conditions have been automated in the AUI in the following way. The AUI performs “phi model completion” whenever generating a data file in which potential-based fluid elements are used. The steps in phi model completion are:

- 1) The AUI loops over all fluid element sides on the boundary of each fluid element region. If the fluid element side already has an interface element, the side is skipped. If the fluid element side has a potential-interface, an interface element of the appropriate type is generated. Otherwise, the side is checked to see if it is attached to a structure (shares structural degrees of freedom with structural elements), close to a structure (nodes coincident with nodes of a structural element) or on a ADINA-F fluid-structure boundary; and, if any of the above conditions are met, an interface element of the appropriate type is generated. The intent of this step is to cover as much of the fluid boundary as possible with interface elements.

Note that phi model completion relies on determining the nodes that are “close” together. The tolerance used for this determination is the same tolerance as is used in coincident node checking

(parameter COINCIDENCE of command TOLERANCES GEOMETRIC).

2) The AUI loops over all nodes attached to interface elements. If the node is attached to structural elements, the node is skipped. Otherwise the types of the attached interface elements are determined. Then

a) If the node is attached only to a free surface interface, ADINA-F interface, inlet-outlet interface or fluid-fluid interface, then the node has a free normal direction (normal to the interface) and zero stiffness directions that are tangential to the free normal. The free normal and zero stiffness directions are identified, and if they are not aligned with the global directions, a skew system is generated that is aligned with the free normal and zero stiffness directions.

b) If the node is attached to a free surface interface, ADINA-F interface, inlet-outlet interface or fluid-fluid interface, and is also attached to a fluid-structure interface or rigid-wall interface, the AUI proceeds as follows. The node has a free normal direction (determined from the free surface interface, ADINA-F interface, inlet-outlet interface or fluid-fluid interface), a structural normal direction (determined from the fluid-structure interface or rigid-wall interface), and, in 3D, another direction orthogonal to the free normal and structural normal directions, which may be a zero stiffness direction or another structural normal direction. The free normal direction is modified to be orthogonal to the structural normal directions. The free normal, structural normal and zero stiffness directions are identified, and, if they are not aligned with the global directions, a skew system is generated that is aligned with the free normal, structural normal and zero stiffness directions.

This process relies on an angle tolerance in 3D analysis to determine if the structure is “smooth”. This tolerance is parameter PHI-ANGLE in command TOLERANCES GEOMETRIC. See Example 2 below.

The intent of step 2 is to identify the zero stiffness and free normal directions of the nodes.

3) The AUI loops over all nodes attached to interface elements. If the node is attached to a structural element, the node is skipped. If the node (node A) is attached to a fluid-structure interface element and is close to a structural node B, node A is constrained to node B as follows. Each displacement degree of freedom for node A is constrained to the corresponding degrees of freedom for node B, accounting for differences in skew systems between A and B, accounting for B possibly being a slave node in a constraint equation (but not accounting for B possibly being a slave in a rigid link), accounting for B possibly being fixed. But a displacement degree of freedom for node A is not constrained if the degree of freedom is a free normal direction (see 2 above).

The intent of step 3) is to connect the fluid mesh with the structural mesh, when different nodes are used for the fluid and structure. The connection still allows the fluid nodes to slip relative to the structural nodes on intersections between free surfaces and the structure.

4) In static analysis, when there are no body force loads, the AUI loops over all nodes on a free surface, ADINA-F interface, inlet-outlet or fluid-fluid interface. If the node is attached to a structural element, the node is skipped. Otherwise, constraint equations are defined for all nodes so that the displacements in the direction of the free normal are equal.

The intent is to remove the zero pivots in the system matrices that are otherwise present (see the discussion after equation (2.11-37)).

5) In dynamic analysis, or in static analysis when there are body force loads, the AUI loops over all nodes on a fluid-fluid interface. If the node is attached to a structural element, the node is skipped. Otherwise constraint equations are defined for all pairs of nodes, so that the displacements in the direction of the free normal are compatible.

The intent is to enforce displacement compatibility between the fluids.

6) The AUI then loops over all nodes with zero stiffness degrees of freedom and defines fixities for each zero stiffness degree of freedom.

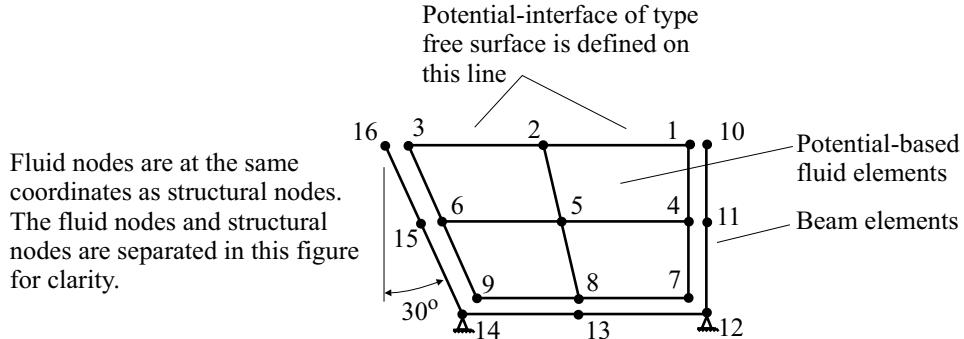
- 7) In static analysis, the AUI constrains all of the potential degrees of freedom for an element group together.
- 8) In dynamic analysis, the AUI constrains all of the potential degrees of freedom on an inlet-outlet together. The intent is to remove the tangential flows on an inlet-outlet.

Note that the model that results from the phi model completion process is not stored in the AUI, but is immediately written to the ADINA data file. Therefore the model that results from phi model completion cannot be displayed during model definition. Phi model completion does write some messages to the log file or user interface, which you may find helpful. However, we recommend that you also check the model by running the model through ADINA, for example, for one load step; then use ADINA-PLOT to display the model.

Example 1: We now present a detailed example for a 2-D fluid filled basin with flexible walls.

Figure 2.11-12(a) shows the model before phi model completion. The model is defined with separate nodes for the fluid and the structure. One way to guarantee that the nodes on the fluid and structure are separate is to generate the structural elements as usual, but to set coincidence checking to “group” when generating the fluid elements.

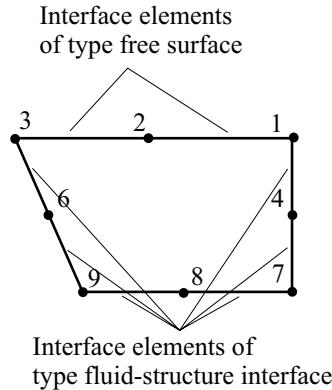
A potential-interface of type free surface is defined on the geometry line corresponding to the free surface.



a) Finite element model before phi model completion

Figure 2.11-12: Example 1 of phi model completion

In step 1 of phi model completion, the AUI generates fluid-structure interface elements where the fluid is adjacent to the structure, and free surface interface elements corresponding to the potential-interface (Figure 2.11-12(b)).

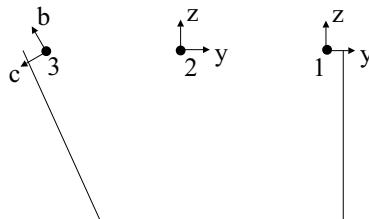


b) Step 1 of phi model completion; interface elements are created

Figure 2.11-12: (continued)

In step 2 of phi model completion, the AUI classifies the displacement directions on the free surface (Figure 2.11-12(c)). Notice that the free normal for node 2 is taken from the free

surface, but the free normal for node 3 is modified by the presence of the adjacent structure. A skew system is defined for node 3 because the free normal and structural normal are not aligned with the global coordinate directions. The zero stiffness direction of node 2 will be fixed in step 6 below.



Node 1: y = structural normal direction,
 z = free normal direction

Node 2: y = zero stiffness direction,
 z = free normal direction

Node 3: b = free normal direction,
 c = structural normal direction

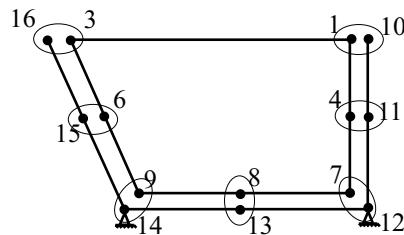
c) Step 2 of phi model completion: classification of displacement directions on free surface

Figure 2.11-12: (continued)

In step 3 of phi model completion, the AUI constrains the fluid displacement directions to the structure (Figure 2.11-12(d)). At node 1, only the y displacement direction is constrained; the z displacement is left free so that the free surface can slip along the wall. At node 3, only the c displacement direction is constrained; the b displacement is left free so that the free surface can slip along the wall.

Notice that at node 4, both the y and z displacements are constrained to the structure. The fluid still slips in the z direction because only the normal displacement (the y displacement in this case) is used by the fluid equations. Similar statements hold for nodes 6 and 8.

Nodes 7 and 9 are fixed because corresponding nodes 12 and 14 are fixed.



$$\text{Node 1: } u_y = u_y^{10}$$

$$\text{Node 3: } u_c = -\cos 30^\circ u_y^{16} - \sin 30^\circ u_z^{16}$$

$$\text{Node 4: } u_y = u_y^{11}, u_z = u_z^{11} \quad \text{Node 6: } u_y = u_y^{15}, u_z = u_z^{15}$$

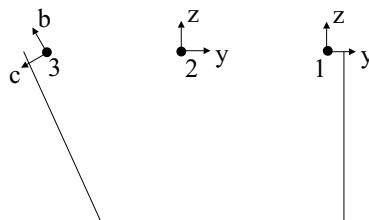
$$\text{Node 7: } u_y = u_z = \text{fixed} \quad \text{Node 8: } u_y = u_y^{13}, u_z = u_z^{13}$$

$$\text{Node 9: } u_y = u_z = \text{fixed}$$

d) Step 3: Creation of constraint equations and fixities

Figure 2.11-12: (continued)

If the analysis is static without body forces, then the AUI performs step 4 of phi model completion (Figure 2.11-12(e)). The free surface can only translate vertically as a rigid body.



$$\text{Node 2: } u_z = u_z^1$$

$$\text{Node 3: } u_b = u_z^1 / \cos 30^\circ$$

Step 4 is only performed in static analysis when there are no body forces.

e) Step 4 of phi model completion: defining constraint equations to set normal displacements equal on free surface

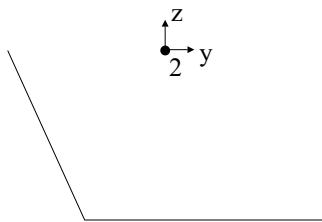
Figure 2.11-12: (continued)

This motion affects the total mass of the fluid region, so that there is no zero pivot in the system matrices.

If there are body forces, then step 4 is not necessary because all boundary motions are given stiffness by the matrix $(\mathbf{K}_{UU})_S$.

Step 5 of phi model completion is skipped because there are no fluid-fluid interfaces.

In step 6 of phi model completion, the zero stiffness direction at node 2 is fixed (Figure 2.11-12(f)). Vertical motions of the nodes attached only to free surface interface elements are allowed, but horizontal motions of these nodes are not allowed (because the fluid does not provide stiffness, damping or mass to horizontal motions).

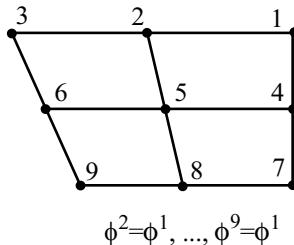


Node 2: $u_y = \text{fixed}$

f) Step 6 of phi model completion: defining fixities to eliminate zero stiffness degrees of freedom

Figure 2.11-12: (continued)

If the analysis is static, then the AUI performs step 7 of phi model completion (Figure 2.11-12g). Only constant (in space) potentials are allowed in static analysis.



Step 7 is performed only in static analysis

- g) Step 7: defining constraint equations to set all potential degrees of freedom equal

Figure 2.11-12: (continued)

Example 2: An alternative way to model the problem of Example 1 is shown in Figure 2.11-13. Here the free surface is shifted downwards slightly so that the nodal coincidence checking feature of the AUI mesh generation algorithms will not reuse any existing nodes on the free surface boundary. Depending on how much you shift the fluid free surface, you may need to adjust the tolerance used in the coincidence checking algorithms.

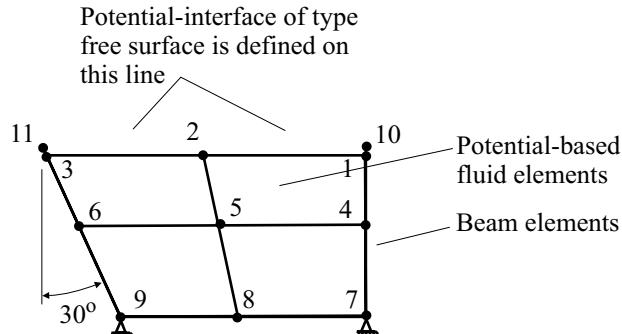


Figure 2.11-13: Alternative modeling of problem in Figure 2.11-12

During phi model completion, it is necessary to set the coincidence tolerance to be loose enough so that node 1 is considered “close” to node 10 and node 3 is considered to be close to node 11. Then steps 1, 2, 4, 6, 7 of phi model completion proceed as in Example 1, and step 3 only processes nodes 1 and 3. The solution results will be almost exactly the same as in Example

1 (the solutions will be slightly different because the geometry is slightly different).

We think that the user input for Example 2 is more difficult than the user input for Example 1 because Example 2 requires a good working knowledge of the nodal coincidence checking features of the AUI mesh generation commands.

Example 3: In 3-D analysis of a fluid-filled basin, there is one additional consideration. Consider the model shown in Figure 2.11-14, in which only the free surface and the adjacent structural nodes are shown.

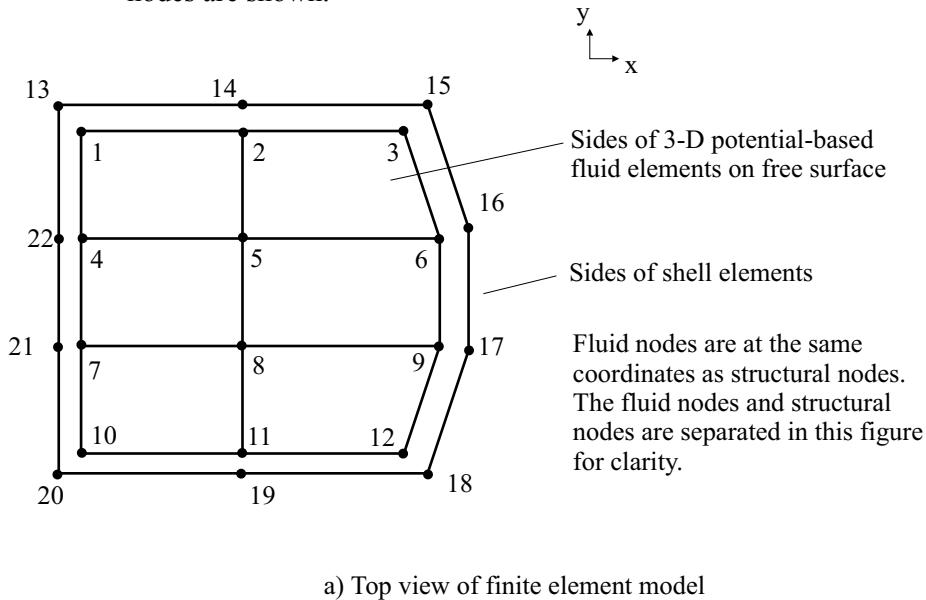
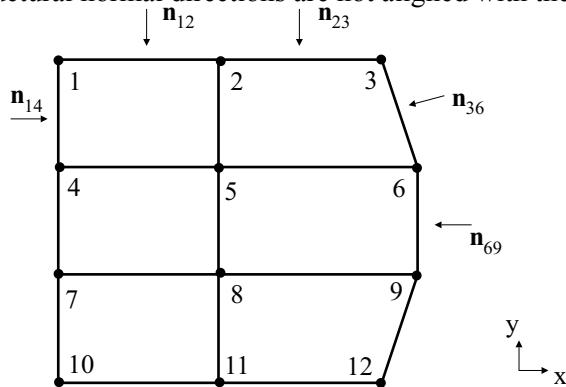


Figure 2.11-14: Example 3 of phi model completion

During step 2 of phi model completion, the AUI determines the structural normal direction(s), zero stiffness direction(s) and free normal direction for the nodes of the free surface (Figure 2.11-14(b)). Notice that node 3 has two structural normals, but node 6 has only one structural normal. That is because the angle between the two structural normals for node 3 is greater than PHI-ANGLE, but the angle between the two structural normals for node 6 is less than PHI-ANGLE.

Nodes 3, 6, 9 and 12 are assigned skew systems because the

structural normal directions are not aligned with the global system.



- Node 1: Structural normal 1 = \mathbf{n}_{14}
 Structural normal 2 = \mathbf{n}_{12}
 Free normal = \mathbf{z}
- Node 2: Structural normal = $\mathbf{n}_{12} = \mathbf{n}_{23}$
 Zero stiffness direction = \mathbf{x}
 Free normal = \mathbf{z}
- Node 3: Structural normal 1 = \mathbf{n}_{23}
 Structural normal 2 = \mathbf{n}_{36}
 Free normal = \mathbf{z}
- Node 5: Zero stiffness direction 1 = \mathbf{x}
 Zero stiffness direction 2 = \mathbf{y}
 Free normal = \mathbf{z}
- Node 6: Structural normal = average of \mathbf{n}_{36} and \mathbf{n}_{69}
 Free normal = \mathbf{z}
 Zero stiffness direction = remaining orthogonal direction

- b) Classification of structural normals and zero stiffness directions for some nodes on the free surface

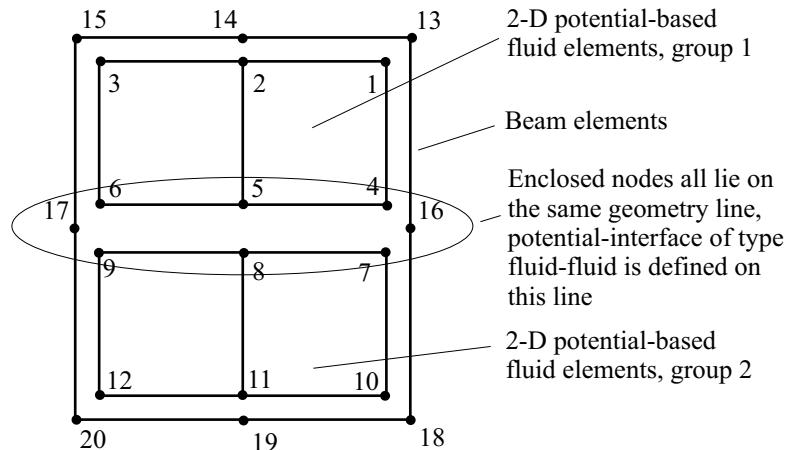
Figure 2.11-14: (continued)

During step 3 of phi model completion, the AUI creates constraint equations for the fluid nodes adjacent to the structural nodes. For example, node 1 is constrained in both the x and y directions to node 13, because both directions are structural normal directions. Node 2 is also constrained in both the x and y directions to node 14, here because the y direction is a structural normal direction and the x direction is a zero stiffness direction. (It is assumed that the structure provides stiffness in the x direction.)

During step 6 of phi model completion, the AUI fixes the x and y directions for nodes 5 and 8, because these directions are zero stiffness directions, and there is no adjacent structure.

Example 4: We now present some of the steps for phi model completion of an enclosure with two distinct fluid regions. Figure 2.11-15(a) shows the model before phi model completion.

Fluid nodes are at the same coordinates as structural nodes.
The fluid nodes and structural nodes are separated in this figure
for clarity.



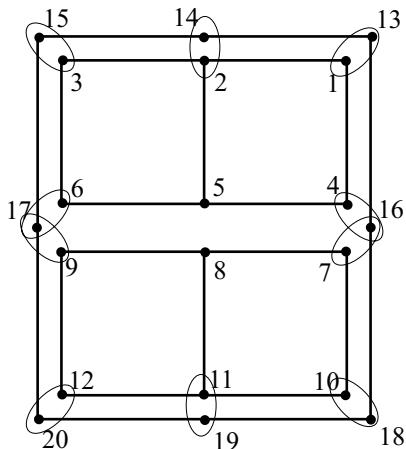
a) Finite element model before phi model completion

Figure 2.11-15: Example 4 of phi model completion

In step 1 of phi model completion, the AUI generates fluid-structure interface elements where the fluid is adjacent to the structure, and fluid-fluid interface elements corresponding to the potential-interface. Four fluid-fluid interface elements are generated, two for each shared element side.

In step 2 of phi model completion, the AUI classifies the displacement directions on the fluid-fluid interface. Here the free normal is always in the z direction and the zero stiffness directions for nodes 5 and 8 are in the y direction.

In step 3 of phi model completion, the fluid nodes are constrained to the adjacent structural nodes (Figure 2.11-15(b)). Notice that the nodes on the fluid-fluid interface are allowed to slip relative to the structure.



Node 1: $u_y=u_y^{13}, u_z=u_z^{13}$ Node 2: $u_y=u_y^{14}, u_z=u_z^{14}$

Node 3: $u_y=u_y^{15}, u_z=u_z^{15}$ Node 4: $u_y=u_y^{16}$

Node 6: $u_y=u_y^{17}$ Node 7: $u_y=u_y^{16}$

Node 9: $u_y=u_y^{17}$ Node 10: $u_y=u_y^{18}, u_z=u_z^{18}$

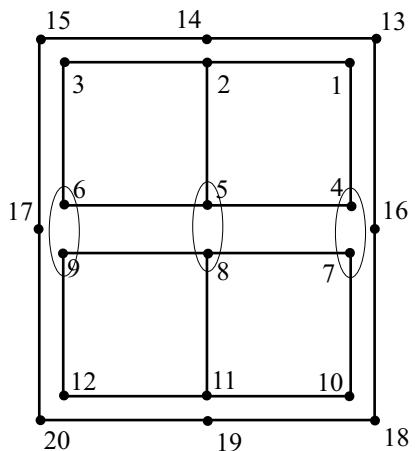
Node 11: $u_y=u_y^{19}, u_z=u_z^{19}$ Node 12: $u_y=u_y^{20}, u_z=u_z^{20}$

b) Step 3: constraining fluid nodes to adjacent structural nodes

Figure 2.11-15: (continued)

If the analysis is static without body forces, then the AUI performs step 4 of phi model completion. In this case, the z displacements of nodes 5 to 9 are constrained to be equal to the z displacement of node 4. The free surface can only translate vertically as a rigid body.

In step 5 of phi model completion, the fluid nodes on the fluid-fluid interface are constrained to each other (Figure 2.11-15(c)). Step 5 is not performed if step 4 was performed. Notice that the potential degrees of freedom are not constrained.



Node 4: $u_z = u_z^7$

Node 6: $u_z = u_z^9$

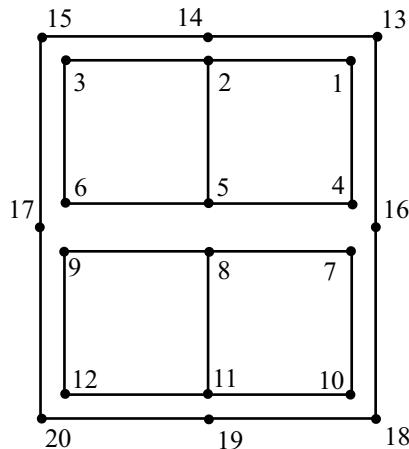
Node 5: $u_z = u_z^8$

c) Step 5: constraining adjacent nodes of the two fluid groups together

Figure 2.11-15: (continued)

In step 6 of phi model completion, the zero stiffness direction of nodes 5 and 8 are fixed. Vertical motions of the nodes attached only to fluid-fluid interface elements are allowed, but horizontal motions are not allowed (because the fluid does not provide stiffness, damping or mass to horizontal motions).

If the analysis is static, then the AUI performs step 7 of phi model completion (Figure 2.11-15(d)). Only constant (in space) potentials are allowed in static analysis, but the potential can be different for the two fluid element groups.



Step 7 is performed only in static analysis

- d) Step 7: defining constraint equations to set the potential degrees of freedom of each element group together

Figure 2.11-15: (continued)

Example 5: We demonstrate how to model an initial pressure in a confined fluid region. This can be done in a static analysis as follows: Define an auxiliary fluid element, physically unattached to the fluid region but belonging to the same fluid element group. On the auxiliary fluid element, place a potential-interface of type inlet-outlet, and apply the specified initial pressure to the inlet-outlet.

The constraint equations that the AUI defines in static analysis connects the auxiliary fluid element to the confined fluid region.

The result is that the value of $\dot{\phi}$ in the confined fluid region is the same as the value of $\dot{\phi}$ in the auxiliary fluid element; the value of $\dot{\phi}$ in the auxiliary fluid element is determined by the pressure that you apply to the inlet-outlet.

If you restart to a dynamic or frequency analysis, fix all of the degrees of freedom of the auxiliary fluid element. Since, in dynamic or frequency analysis, the AUI does not define constraint equations connecting the potential degrees of freedom, the auxiliary fluid element is no longer connected to the confined fluid

region. The initial values of ϕ in the confined fluid region are that from the static analysis, so the confined fluid region has the same initial pressure as in static analysis.

The modeling process is schematically illustrated in Figure 2.11-16.

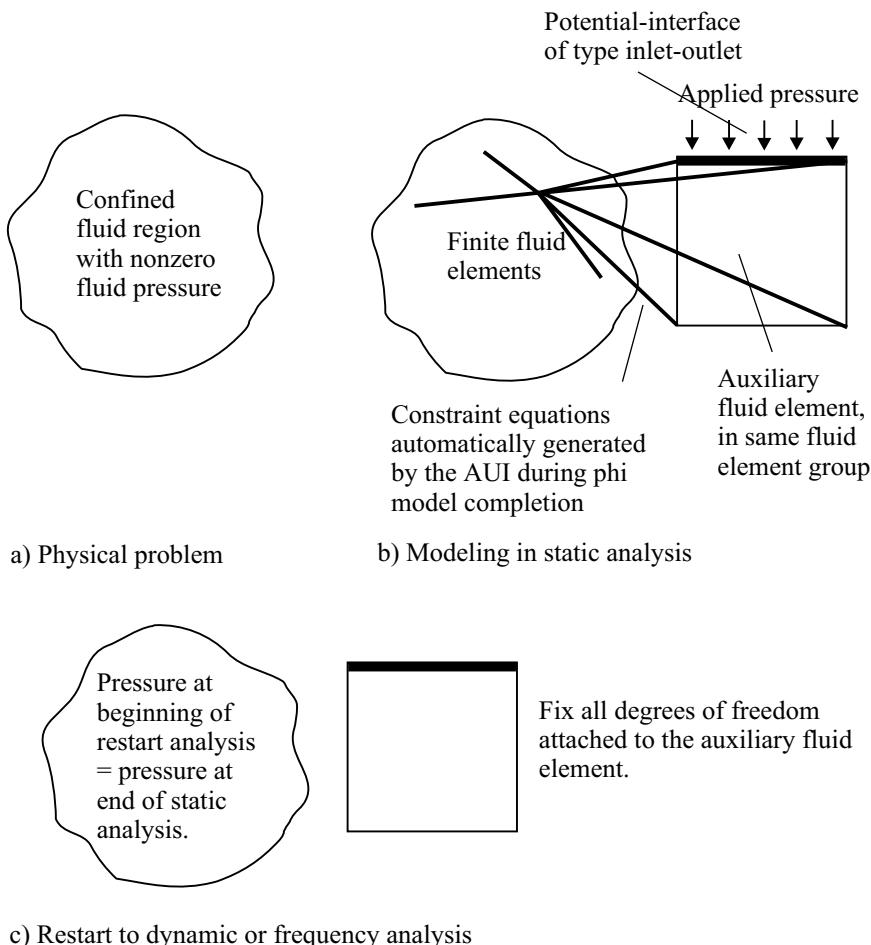


Figure 2.11-16: Modeling an internal pressure in a confined fluid region

2.11.16 Modeling: Considerations for static analysis

In static analysis, the potential degree of freedom output should

be interpreted as $\dot{\phi}$, not ϕ .

2.11.17 Modeling: Considerations for dynamic analysis

Any of the direct time integration methods available in ADINA, except for the central difference method, can be used in direct time integration. Rayleigh damping can be specified in the structure.

Please remember that there is no damping within the fluid. Therefore the choice of initial conditions is extremely important, because poorly chosen initial conditions will cause transient responses that are not damped out.

2.11.18 Modeling: Considerations for frequency analysis and modal participation factor analysis

Frequency analysis (and all analyses that depend upon frequency data) cannot be performed if infinite interface elements are present in the analysis.

For ground motion modal participation factor calculations, it is recommended that each fluid region be completely surrounded by interface elements. This sets the modal participation factors for the ϕ rigid body modes to zero.

Figure 2.11-17 shows a one-dimensional example in which there is a ϕ rigid body mode and the ground motion modal participation factor is nonzero. The motion excited by the nonzero ground motion modal participation factor corresponds to a constant ground acceleration. Notice that this ground motion causes the unbounded expansion of the fluid region.

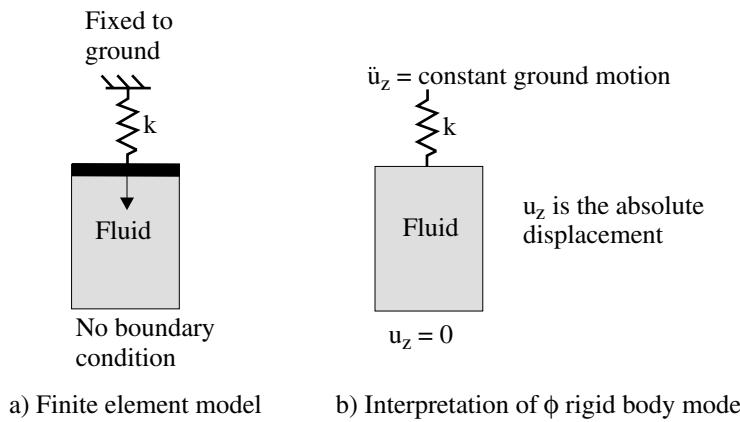


Figure 2.11-17: Problem in which the ground motion modal participation factor for the ϕ rigid body mode is nonzero

2.11.19 Modeling: Element output

Each fluid element outputs, at its integration points, the following information to the porthole file (the integration point numbering is the same as the numbering convention used for the solid elements): `FE_PRESSURE`, `FLUID_REFERENCE_PRESSURE`, `ELEMENT_X-VELOCITY`, `ELEMENT_Y-VELOCITY`, `ELEMENT_Z-VELOCITY` (`ELEMENT_X-VELOCITY` is applicable only for 3-D elements). `FE_PRESSURE` is the value of p and `FLUID_REFERENCE_PRESSURE` is the value of $\rho\Omega$ (see Equation (2.11-59)).

In the output of modal quantities, `ELEMENT_X-VELOCITY`, `ELEMENT_Y-VELOCITY`, `ELEMENT_Z-VELOCITY` should be interpreted as modal particle displacements, not velocities, in accordance with equation (2.11-42). However in all other types of analysis (static, direct time integration, modal superposition, response spectrum, harmonic and random), these quantities are velocities. The velocities are absolute velocities, even when ground motions are entered using mass-proportional loads.

Interface elements do not have any output.

This page intentionally left blank.

3. Material models and formulations

The objective of this chapter is to summarize the theoretical basis and practical use of the material models and formulations available in ADINA.

3.1 Stress and strain measures in ADINA

3.1.1 Summary

- It is important to recognize which stress and strain measures are employed in the use of a material model:
 - ▶ In the preparation of the input data in which the material parameters are defined with respect to these stress and strain measures
 - ▶ In the interpretation of the analysis results in which the type of stresses and strains output must be considered
- The practical use of the material models available in ADINA regarding the stress and strain measures used for the input data and analysis results is summarized in the following. These stress/strain measures are described in detail in ref. KJB, Section 6.2.

Small displacement/small strain formulation

Input of material parameters: All elements and material models use the engineering stress-engineering strain relationship.

Output: All elements and material models output Cauchy stresses and engineering strains.

Large displacement/small strain formulation

Input of material parameters: 2nd Piola-Kirchhoff stresses and Green-Lagrange strains. Note that under small strain conditions, 2nd Piola-Kirchhoff stresses are nearly equal to engineering stresses, and Green-Lagrange strains are nearly equal to engineering strains.

Output:

- (1) 2-D, 3-D elements: all material models output Cauchy stresses and Green-Lagrange strains.
- (2) Beam, iso-beam, pipe, shell elements: all material models output 2nd Piola-Kirchhoff stresses and Green-Lagrange strains.

Large displacement/large strain formulation, 2D and 3D elements

For the two- and three-dimensional solid elements, the following material models can be used:

- (1) Plastic-bilinear, plastic-multilinear, Drucker-Prager, Mroz bilinear, plastic-orthotropic, thermo-plastic, creep, plastic-creep, multilinear-plastic-creep, plastic-creep-variable, multilinear-plastic-creep-variable, viscoelastic, user-supplied. In these cases the updated Lagrangian Hencky formulation is used.

Input of material parameters: Cauchy (true) stresses and logarithmic (true) strains

Output: Cauchy stresses and deformation gradients.

- ref. A.L. Eterovic and K.J. Bathe, "A Hyperelastic-Based Large Strain Elasto-Plastic Constitutive Formulation with Combined Isotropic-Kinematic Hardening Using the Logarithmic Stress and Strain Measures", *Int. J. Numerical Methods in Engineering*, Vol. 30, pp. 1099-1114, 1990.

- (2) Mooney-Rivlin, Ogden, Arruda-Boyce, hyper-foam. In this case the total Lagrangian formulation is used.

Input of material parameters: Mooney-Rivlin, Ogden, Arruda-Boyce or hyper-foam constants.

Output: Cauchy stresses and deformation gradients.

- ref. T. Sussman and K.J. Bathe, "A Finite Element Formulation for Nonlinear Incompressible Elastic and Inelastic Analysis," *J. Computers and Structures*, Vol. 26, 1987.

Large displacement/large strain formulation, shell elements

For the shell elements, the following material models can be used, provided that the elements are single layer elements described using midsurface nodes:

- (1) Plastic-bilinear, plastic-multilinear. Either the updated Lagrangian Jaumann (ULJ) formulation or the updated Lagrangian Hencky (ULH formulation), can be used if the shell elements are 3-node, 4-node, 9-node or 16-node shell elements. The default is the ULH formulation, except in the following cases:
 - a) the rigid-target contact algorithm is employed, or
 - b) explicit time integration is employed
- (2) Plastic-orthotropic. The shell elements must be 3-node, 4-node, 9-node or 16-node elements. The updated Lagrangian Jaumann (ULJ) formulation is employed.

When the ULJ formulation is employed for shell elements:

Input of material parameters: Cauchy (true) stresses and logarithmic (true) strains.

Output: Cauchy stresses and logarithmic strains.

When the ULH formulation is employed for shell elements:

Input of material parameters: Kirchhoff stresses and logarithmic (true) strains.

Output: Kirchhoff stresses and left Hencky strains.

- Small strains are strains less than about 2%.

Strain measures: The strain measures used in ADINA are illustrated hereafter in the simplified case of a rod under uniaxial tension (see Fig. 3.1-1).

$$\text{Engineering strain: } e_0 = \frac{\ell - \ell_0}{\ell_0}$$

$$\text{Green-Lagrange strain: } \varepsilon = \frac{1}{2} \frac{\ell^2 - \ell_0^2}{\ell_0^2}$$

$$\text{Almansi strain: } \varepsilon_a = \frac{1}{2} \frac{\ell^2 - \ell_0^2}{\ell^2}$$

$$\text{Logarithmic strain and Hencky strains: } e = \ln\left(\frac{\ell}{\ell_0}\right) \left[= \int_{\ell_0}^{\ell} \frac{d\ell}{\ell} \right]$$

$$\text{Stretch: } \lambda = \frac{\ell}{\ell_0}$$

ref. KJB
Sec. 6.2.2

- Green-Lagrange strains are used in the large displacement/small strain formulations. This is because large rotations do not affect the Green-Lagrange strains (Green-Lagrange strains are invariant with respect to rigid-body rotations), and for small strains, small rotations, Green-Lagrange strains and engineering strains are equivalent.
- In the small strain formulations, the current area is always assumed to be equal to the initial undeformed area.

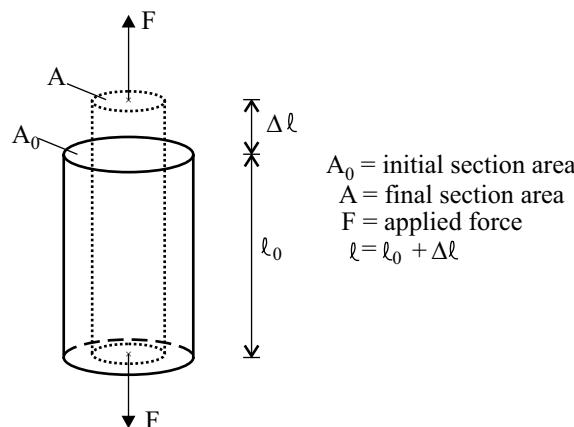


Figure 3.1-1: Rod under uniaxial tension

- Engineering strains are also called nominal strains in the literature.
- Logarithmic strains are also known as true strains.

Stress measures: The stress measures used in ADINA are illustrated hereafter in the simplified case of a rod under uniaxial tension (see Fig. 3.1-1).

Engineering stress: $\sigma = \frac{F}{A_0}$

Cauchy stress: $\tau = \frac{F}{A} = \frac{\sigma A_0}{A}$

2nd Piola-Kirchhoff stress: $S = \frac{F\ell_0}{A_0\ell} = \frac{\sigma\ell_0}{\ell}$

Kirchhoff stress: $J\tau = \frac{F\ell}{A_0\ell_0} = \frac{\sigma\ell}{\ell_0}$

- Cauchy stresses are also known as true stresses.
- For the case in which the material is incompressible, $\tau = J\tau = \frac{\sigma\ell}{\ell_0}$ can be used to compute the Cauchy stress and the Kirchhoff stress from the engineering stress.
- When the strains are small, the 2nd Piola-Kirchhoff stresses are nearly equal to the Cauchy stresses from which the rigid body rotations have been removed.
- 2nd Piola-Kirchhoff stresses are input only for the large displacement/small strain formulation. Because the strains are assumed to be small, the engineering stresses can be entered as the stress input quantities.
- 2nd Piola-Kirchhoff stresses are output only for the large

displacement/small strain formulation and for element types in which the stresses expressed in the element coordinate system of the deformed element are of physical significance. If the element does not undergo rigid body rotations, the 2nd Piola-Kirchhoff stresses are nearly equal to the Cauchy stresses, because the strains are assumed to be small. If the element also undergoes rigid body rotations, the Cauchy stresses expressed in the element coordinate system do not change (because the element coordinate system also rotates). Since the 2nd Piola-Kirchhoff stresses, as expressed in the undeformed element coordinate system, also do not change during a rigid body rotation, these 2nd Piola-Kirchhoff stresses remain nearly equal to the Cauchy stresses. The 2nd Piola-Kirchhoff stresses here provide a convenient way of removing the rigid body rotations from the output stresses.

- When the material is nearly incompressible, the Kirchhoff stresses are nearly equal to the Cauchy stresses.
- Since Kirchhoff stresses are input/output only for materials that are nearly incompressible, practically speaking, the differences between Kirchhoff and Cauchy stresses are negligible.

3.1.2 Large strain thermo-plasticity and creep analysis

- This section discusses large strain analysis (ULH formulation) with the following material models: plastic-bilinear, plastic-multilinear, Drucker-Prager, Mroz bilinear, plastic-orthotropic, thermo-plastic, creep, plastic-creep, multilinear-plastic-creep, creep-variable, plastic-creep-variable, multilinear-plastic-creep-variable, user-supplied.
- The following is a quick summary of the theory of large strain inelastic analysis. For further information, see ref KJB, Section 6.6.4 and also the following references:

ref. F.J. Montáns and K.J. Bathe, "Computational issues in large strain elasto-plasticity: an algorithm for mixed hardening and plastic spin", *Int. J. Numer. Meth. Engng*, 2005; 63;159-196.

ref. M. Kojić and K.J. Bathe, *Inelastic Analysis of Solids and Structures*, Springer-Verlag, 2003.

Total deformation gradient tensor: Let \mathbf{X} be the total deformation gradient tensor at time t with respect to an initial configuration taken at time 0. For ease of writing, we do not include the usual left superscripts and subscripts.

Polar decomposition into rotation and right stretch tensor: The total deformation gradient tensor \mathbf{X} can be decomposed into a material rigid-body rotation tensor \mathbf{R} and a symmetric positive-definite (right) stretch tensor \mathbf{U} (polar decomposition):

$$\mathbf{X} = \mathbf{R} \mathbf{U} \quad (3.1-1)$$

Principal directions of right stretch tensor: The right stretch tensor \mathbf{U} can be represented in its principal directions by a diagonal tensor $\mathbf{\Lambda}$, such that

$$\mathbf{U} = \mathbf{R}_L \mathbf{\Lambda} \mathbf{R}_L^T \quad (3.1-2)$$

where \mathbf{R}_L is a rotation tensor with respect to the fixed global axes (see Figure 3.1-2).

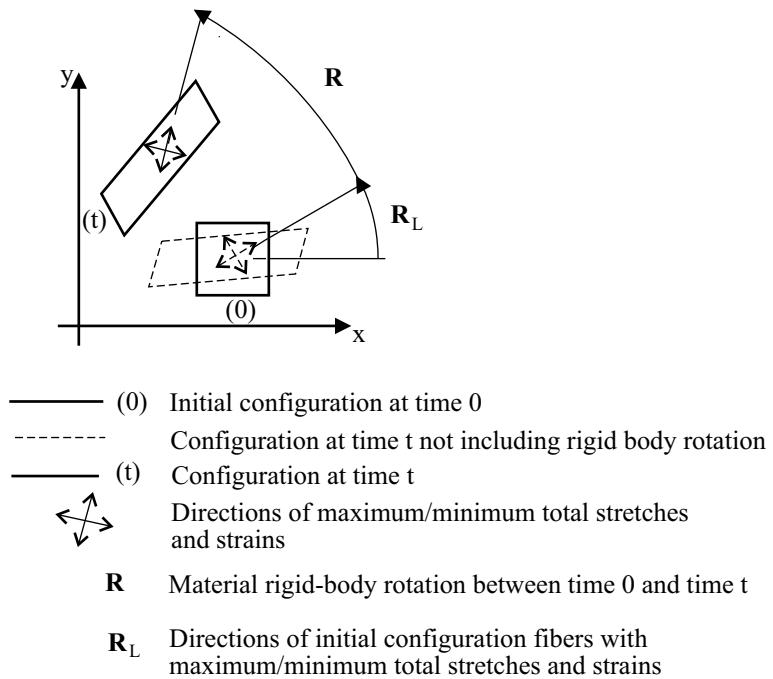


Figure 3.1-2: Directions of maximum/minimum total stretches and strains

(Note that the rotation \mathbf{R}_L does not correspond to a material rigid-body rotation, but to a rotation of the coordinate system: \mathbf{U} and $\mathbf{\Lambda}$ are two representations of the same deformed state, respectively in the global coordinate system and in the \mathbf{U} principal directions coordinate system.)

Right Hencky strain tensor: The Hencky strain tensor (computed in the right basis) is given by

$$\mathbf{E}^R = \ln \mathbf{U} = \mathbf{R}_L \ln \mathbf{\Lambda} \mathbf{R}_L^T \quad (3.1-3)$$

The superscript “R” symbolizes the right basis.

Polar decomposition into rotation and left stretch tensor: The total deformation gradient tensor \mathbf{X} can also be decomposed into a material rigid-body rotation \mathbf{R} and a symmetric positive-definite (left) stretch tensor \mathbf{V} (polar decomposition):

$$\mathbf{X} = \mathbf{V} \mathbf{R} \quad (3.1-4)$$

\mathbf{R} in (3.1-3) is the same as \mathbf{R} in (3.1-1).

Principal directions of left stretch tensor: The left stretch tensor \mathbf{V} can be represented in its principal directions by a diagonal tensor $\mathbf{\Lambda}$, such that

$$\mathbf{V} = \mathbf{R}_E \mathbf{\Lambda} \mathbf{R}_E^T \quad (3.1-5)$$

where \mathbf{R}_E is a rotation tensor with respect to the fixed global axes.

Note that $\mathbf{R}_E = \mathbf{R} \mathbf{R}_L$.

Left Hencky strain tensor: The Hencky strain tensor (computed in the left basis) is given by

$$\mathbf{E}^L = \ln \mathbf{V} = \mathbf{R}_E \ln \mathbf{\Lambda} \mathbf{R}_E^T \quad (3.1-6)$$

The superscript “L” symbolizes the left basis.

Comparison of left and right Hencky strain tensors: The principal values of the left and right Hencky strain tensors are identical, and equal to the logarithms of the principal stretches. Hence both of these strain tensors can be considered to be logarithmic strain tensors. However, the principal directions of the left and right Hencky strain tensors are different. The principal directions of the right Hencky strain tensor do not contain the rigid body rotations of the material, but the principal directions of the left Hencky strain tensor contain the rigid body rotations of the material.

Therefore, for a material undergoing rigid body rotations, the principal directions of the right Hencky strain tensor do not rotate, however the principal directions of the left Hencky strain tensor rotate with the material. Hence, the left Hencky strain tensor is preferred for output and visualization of the strain state.

Multiplicative decomposition of deformation gradient in inelastic analysis: In inelastic analysis, the following multiplicative decomposition of the total deformation gradient into

an elastic deformation gradient \mathbf{X}^E and an inelastic deformation gradient \mathbf{X}^P is assumed:

$$\mathbf{X} = \mathbf{X}^E \mathbf{X}^P \quad (3.1-7)$$

To understand (3.1-7), consider a small region of material under a given stress state with deformation gradient \mathbf{X} . If this region of material is separated from the rest of the model and subjected to the same stress state, the deformation gradient is still \mathbf{X} . Now if the stress state is removed, (3.1-7) implies that the deformation gradient of the unloaded material is \mathbf{X}^P . The stresses are due entirely to the strains associated with the elastic deformation gradient \mathbf{X}^E .

It can be shown (see Montáns and Bathe), that (3.1-7) is equivalent to the additive decomposition of the displacements into elastic displacements and plastic displacements.

For the materials considered here, $\det \mathbf{X}^P = 1$.

Polar decomposition of elastic deformation gradient: The elastic deformation gradient can be decomposed into an elastic rotation tensor \mathbf{R}^E and elastic right and left stretch tensors \mathbf{U}^E , \mathbf{V}^E :

$$\mathbf{X}^E = \mathbf{R}^E \mathbf{U}^E = \mathbf{V}^E \mathbf{R}^E \quad (3.18-a,b)$$

Elastic Hencky strain tensors: The elastic Hencky strain tensors in the right and left bases are given by

$$\mathbf{E}^{ER} = \ln \mathbf{U}^E, \quad \mathbf{E}^{EL} = \ln \mathbf{V}^E \quad (3.1-9a,b)$$

Stress-strain relationships: The stresses are computed from the elastic Hencky strain tensors using the usual stress-strain law of isotropic elasticity. However, the stress measures used depend upon the strain measures used. When the right Hencky strain measure is used, the stress measure used is the rotated Kirchhoff stress

$$\bar{\boldsymbol{\tau}} = (\mathbf{R}^E)^T J \boldsymbol{\tau} \mathbf{R}^E \quad (3.1-10)$$

and when the left Hencky strain measure is used, the stress measure is the (unrotated) Kirchhoff stress $J\boldsymbol{\tau}$. $J = \det \mathbf{X}$ is the volume change of the material, and, using $\det \mathbf{X}^P = 1$, $J = \det \mathbf{X}^E$.

With these choices of stress and strain measures, the stresses and strains are work-conjugate.

The choice of right Hencky strain and rotated Kirchhoff stresses gives the same numerical results as the choice of left Hencky strain and (unrotated) Kirchhoff stresses).

Implementation notes: For 2-D and 3-D solid elements, the difference between the Cauchy and Kirchhoff stresses is neglected. The stress measure used with the right Hencky strains is

$\bar{\boldsymbol{\tau}} = (\mathbf{R}^E)^T \boldsymbol{\tau} \mathbf{R}^E$. The input of material properties is assumed to be in terms of Cauchy stresses, and the output of stresses is in terms of Cauchy stresses.

For shell elements, Kirchhoff stresses are used throughout. The input of material properties is assumed to be in terms of Kirchhoff stresses, and the output of stresses is in terms of Kirchhoff stresses.

These assumptions are justified because they are used with material models in which the plastic deformations are incompressible and the plastic deformations are generally much larger than the elastic deformations.

3.2 Linear elastic material models

- The following material models are discussed in this section:

Elastic-isotropic: isotropic linear elastic

Elastic-orthotropic: orthotropic linear elastic

In each model, the total stress is uniquely determined by the total strain.

- These models can be employed using the **small displacement** or **large displacement** formulations. In all cases, the strains are assumed small.

When the elastic-isotropic and elastic-orthotropic materials are used with the small displacement formulation, the formulation is linear.

- If the material models are employed in a large displacement analysis, the total or the updated Lagrangian formulation is automatically selected by the program depending on which formulation is numerically more effective.

- In the small displacement formulation, the stress-strain relationship is

$${}^t_0\sigma = \mathbf{C} {}^t_0\mathbf{e}$$

in which ${}^t_0\sigma$ = engineering stresses and ${}^t_0\mathbf{e}$ = engineering strains.

- In the total Lagrangian formulation, the stress-strain relationship is

$${}^t_0\mathbf{S} = \mathbf{C} {}^t_0\mathbf{\varepsilon}$$

in which ${}^t_0\mathbf{S}$ = second Piola-Kirchhoff stresses and ${}^t_0\mathbf{\varepsilon}$ = Green-Lagrange strains.

- In the updated Lagrangian formulation, the stress-strain relationship is

$${}^t\tau = \mathbf{C} {}^t_t\mathbf{\varepsilon}^a$$

in which ${}^t\tau$ = Cauchy stresses and ${}^t_t\mathbf{\varepsilon}^a$ = Almansi strains.

- The same matrix \mathbf{C} is employed in all of these formulations. As long as the strains are small (with large displacements), the difference in the response predictions obtained with the total and

updated Lagrangian formulations is negligible.

- However, if the strains are large, the difference in the response predictions is very significant (see ref. KJB, pp 589-590). If the strains are large, it is recommended that the linear elastic material model not be used.

3.2.1 Elastic-isotropic material model

- This material model is available for the **truss**, **2-D solid**, **3-D solid**, **beam**, **iso-beam**, **plate**, **shell** and **pipe** elements.
- The two material constants used to define the constitutive relation (the matrix **C**) are

$$E = \text{Young's modulus}, \nu = \text{Poisson's ratio}$$

*ref. KJB
Table 4.3,
p. 194*

- The same constants are employed in the small and large displacement formulations, and hence the matrices **C** are identically the same in all formulations.
- You can specify the coefficient of thermal expansion as part of the elastic-isotropic material description. The coefficient of thermal expansion is assumed to be temperature-independent.

3.2.2 Elastic-orthotropic material model

- The elastic-orthotropic material model is available for the **2-D solid**, **3-D solid**, **plate** and **shell** elements.

2-D solid elements: Figure 3.2-1 shows a typical two-dimensional element, for which the in-plane orthogonal material axes are "a" and "b". The third orthogonal material direction is "c" and is perpendicular to the plane defined by "a" and "b". The material constants are defined in the principal material directions (a,b,c).

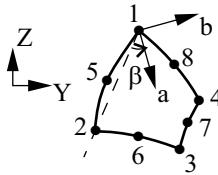


Figure 3.2-1: Principal in-plane material axes orientation for the orthotropic material model for 2-D solid elements

The stress-strain matrices are as follows for the various 2-D element types:

Plane stress:

$$\begin{bmatrix} e_a \\ e_b \\ e_c \\ \gamma_{ab} \end{bmatrix} = \begin{bmatrix} \frac{1}{E_a} & -\frac{\nu_{ab}}{E_b} & -\frac{\nu_{ac}}{E_c} & 0 \\ -\frac{\nu_{ba}}{E_a} & \frac{1}{E_b} & -\frac{\nu_{bc}}{E_c} & 0 \\ -\frac{\nu_{ca}}{E_a} & -\frac{\nu_{cb}}{E_b} & \frac{1}{E_c} & 0 \\ 0 & 0 & 0 & \frac{1}{G_{ab}} \end{bmatrix} \begin{bmatrix} \sigma_a \\ \sigma_b \\ \sigma_c = 0 \\ \sigma_{ab} \end{bmatrix}$$

Plane strain:

$$\begin{bmatrix} e_a \\ e_b \\ e_c = 0 \\ \gamma_{ab} \end{bmatrix} = \begin{bmatrix} \frac{1}{E_a} & -\frac{\nu_{ab}}{E_b} & -\frac{\nu_{ac}}{E_c} & 0 \\ -\frac{\nu_{ba}}{E_a} & \frac{1}{E_b} & -\frac{\nu_{bc}}{E_c} & 0 \\ -\frac{\nu_{ca}}{E_a} & -\frac{\nu_{cb}}{E_b} & \frac{1}{E_c} & 0 \\ 0 & 0 & 0 & \frac{1}{G_{ab}} \end{bmatrix} \begin{bmatrix} \sigma_a \\ \sigma_b \\ \sigma_c \\ \sigma_{ab} \end{bmatrix}$$

Axisymmetric:

$$\begin{bmatrix} e_a \\ e_b \\ e_c \\ \gamma_{ab} \end{bmatrix} = \begin{bmatrix} \frac{1}{E_a} & -\frac{\nu_{ab}}{E_b} & -\frac{\nu_{ac}}{E_c} & 0 \\ -\frac{\nu_{ba}}{E_a} & \frac{1}{E_b} & -\frac{\nu_{bc}}{E_c} & 0 \\ -\frac{\nu_{ca}}{E_a} & -\frac{\nu_{cb}}{E_b} & \frac{1}{E_c} & 0 \\ 0 & 0 & 0 & \frac{1}{G_{ab}} \end{bmatrix} \begin{bmatrix} \sigma_a \\ \sigma_b \\ \sigma_c \\ \sigma_{ab} \end{bmatrix}$$

The seven material constants (E_a , E_b , E_c , ν_{ab} , ν_{ac} , ν_{bc} and G_{ab}) define the symmetric compliance matrix \mathbf{C}_ℓ^{-1} .

$$\mathbf{C}_\ell^{-1} = \begin{bmatrix} \frac{1}{E_a} & -\frac{\nu_{ab}}{E_b} & 0 & -\frac{\nu_{ac}}{E_c} \\ 0 & \frac{1}{E_b} & 0 & -\frac{\nu_{bc}}{E_c} \\ 0 & 0 & \frac{1}{G_{ab}} & 0 \\ \text{symmetric} & & & \frac{1}{E_c} \end{bmatrix}$$

where the subscript " ℓ " in \mathbf{C}_ℓ^{-1} indicates that the material law is given in the local system of the material axes. Note that the determinant of \mathbf{C}_ℓ^{-1} must not be zero in order to be able to calculate the inverse \mathbf{C}_ℓ . This imposes the following restrictions on the constants. The material constants must be defined so that the stress-strain constitutive matrix is positive-definite; i.e.:

$$\left| \nu_{ji} \right| < \left(\frac{E_i}{E_j} \right)^{\frac{1}{2}}, \quad i, j = a, b, c$$

$$\nu_{ab}\nu_{bc}\nu_{ca} < 0.5 \left(1 - \nu_{ab}^2 \frac{E_a}{E_b} - \nu_{bc}^2 \frac{E_b}{E_c} - \nu_{ca}^2 \frac{E_c}{E_a} \right) \leq 0.5$$

Based on the input values for ν_{ij} , ADINA calculates ν_{ji} so as to have a symmetric constitutive matrix; i.e.,

$$\frac{\nu_{ji}}{E_i} = \frac{\nu_{ij}}{E_j}$$

See also for example the following reference:

ref. Jones, R.M., *Mechanics of Composite Materials*, McGraw-Hill p. 42, 1975.

In ADINA, the Poisson's ratio ν_{ij} is defined differently from Jones, although it is consistent with Jones' nomenclature. In Figure 3.2-2, a uniaxial tensile test is illustrated in which the material under test is loaded in the a -direction. The Poisson's ratio is defined in ADINA by

$$\nu_{ba} = -\frac{\varepsilon_b}{\varepsilon_a}$$

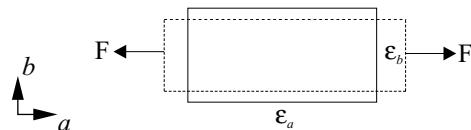


Figure 3.2-2: Definition of ν_{ba}

In general,

$$\nu_{ij} = -\frac{\varepsilon_i}{\varepsilon_j} \quad (i, j = a, b, c)$$

In Jones, however, the same Poisson's ratio is notated

$$^J \nu_{ij} = -\frac{\varepsilon_j}{\varepsilon_i}.$$

Using the correspondence between the Jones and ADINA nomenclature for the Poisson's ratio,

$$\nu_{ji} = ^J \nu_{ij}$$

and the relationship

$$\frac{\nu_{ji}}{E_i} = \frac{\nu_{ij}}{E_j}$$

it is possible to calculate the value of ν_{ji} given a value for $^J \nu_{ij}$ from

$$\nu_{ij} = \left(\frac{E_j}{E_i} \right) ^J \nu_{ij}$$

When material data is available from sources that follow Jones' notation of Poisson's ratio, the above conversion allows the equivalent ADINA Poisson's ratio to be calculated from the Jones Poisson's ratio for input to ADINA.

To obtain the stress-strain matrix \mathbf{C} corresponding to the global material axes, first \mathbf{C}_ℓ is calculated and then

$$\mathbf{C} = \mathbf{Q}^T \mathbf{C}_\ell \mathbf{Q}$$

where

$$\mathbf{Q} = \begin{vmatrix} \ell_1^2 & m_1^2 & \ell_1\ell_2 & 0 \\ \ell_2^2 & m_2^2 & m_1m_2 & 0 \\ 2m_1m_2 & 2\ell_1\ell_2 & \ell_1m_2 + \ell_2m_1 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

and ℓ_1, ℓ_2 and m_1, m_2 are the direction cosines of the a, b material axes to the global axes.

Note that in a large displacement analysis it is assumed that the angle β remains constant throughout the incremental solution. Hence, the use of the total Lagrangian formulation is usually most appropriate.

3-D solid elements: In three-dimensional analysis we have

$$\mathbf{C}_\ell^{-1} = \begin{bmatrix} 1/E_a & -v_{ab}/E_b & -v_{ac}/E_c & 0 & 0 & 0 \\ & 1/E_b & -v_{bc}/E_c & 0 & 0 & 0 \\ & & 1/E_c & 0 & 0 & 0 \\ & & & 1/G_{ab} & 0 & 0 \\ & & & & 1/G_{ac} & 0 \\ & & & & & 1/G_{bc} \end{bmatrix}$$

symmetric

and, as in two-dimensional analysis, this matrix is inverted and then transformed to the global coordinate axes.

Plate elements: Figure 3.2-2 shows a typical plate element for which the in-plane orthogonal material axes are "a" and "b". The material constants are defined in the principal material directions (a,b), for which we have

$$\begin{bmatrix} e_a \\ e_b \\ e_c \\ \gamma_{ab} \end{bmatrix} = \begin{bmatrix} 1/E_a & -v_{ab}/E_b & -v_{ac}/E_c & 0 \\ -v_{ba}/E_a & 1/E_b & -v_{bc}/E_c & 0 \\ -v_{ca}/E_a & -v_{cb}/E_b & 1/E_c & 0 \\ 0 & 0 & 0 & 1/G_{ab} \end{bmatrix} \begin{bmatrix} \sigma_a \\ \sigma_b \\ \sigma_c = 0 \\ \sigma_{ab} \end{bmatrix}$$

Note that the transverse normal strain e_c is not calculated for the plate elements.

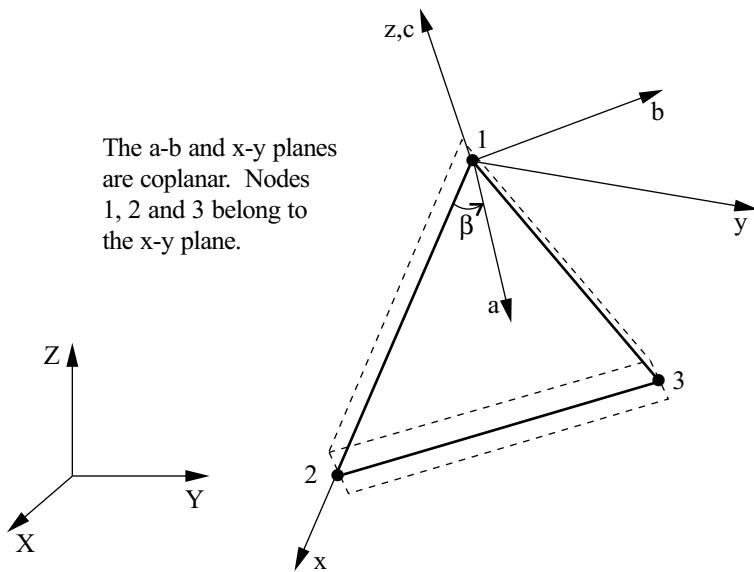


Figure 3.2-3: Principal in-plane material axes orientation for the orthotropic material model for plate elements

In order to obtain the stress-strain matrix \mathbf{C} corresponding to the local x y z directions of the element, we use

$$\mathbf{C} = \mathbf{Q}^T \mathbf{C}_\ell \mathbf{Q}^T$$

$$\text{where } \mathbf{Q}^T = \begin{bmatrix} \cos\beta \cos\beta & \sin\beta \sin\beta & \cos\beta \sin\beta \\ \sin\beta \sin\beta & \cos\beta \cos\beta & -\cos\beta \sin\beta \\ -2\cos\beta \sin\beta & 2\cos\beta \sin\beta & \cos^2\beta - \sin^2\beta \end{bmatrix}$$

β is the angle between the a and x axes, see Fig. 3.2-2.

Shell elements: Figure 3.2-3 shows typical shell elements for which the orthogonal material axes are "a", "b" and "c".

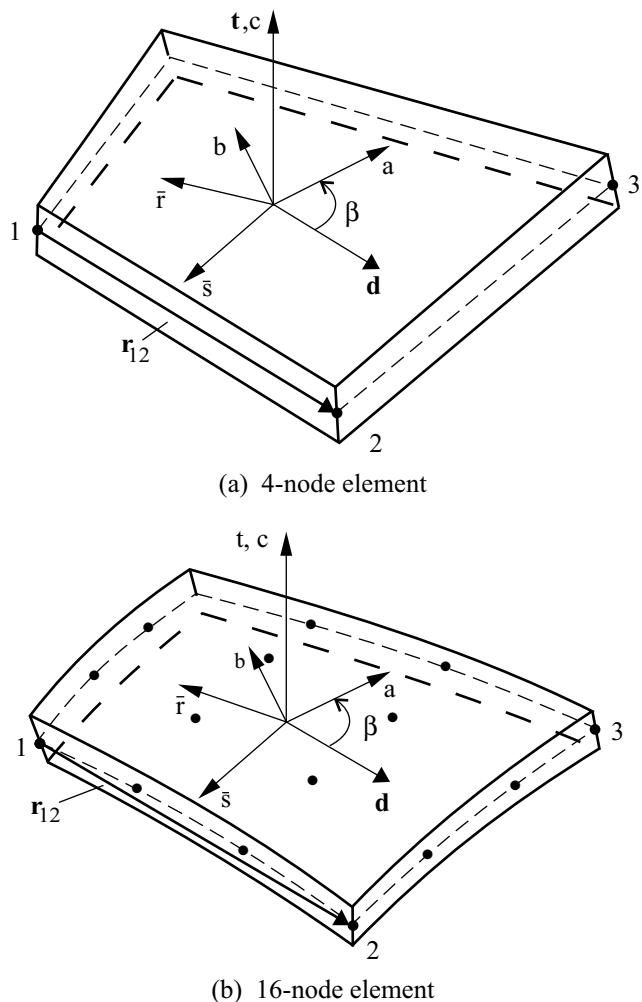
The constitutive relation defined in the a,b,c system is

$$\begin{bmatrix} e_a \\ e_b \\ e_c \\ \gamma_{ab} \\ \gamma_{ac} \\ \gamma_{bc} \end{bmatrix} = \begin{bmatrix} 1/E_a & -v_{ab}/E_b & -v_{ac}/E_c & 0 & 0 & 0 \\ -v_{ba}/E_a & 1/E_b & -v_{bc}/E_c & 0 & 0 & 0 \\ -v_{ca}/E_a & -v_{cb}/E_b & 1/E_c & 0 & 0 & 0 \\ 0 & 0 & 0 & 1/G_{ab} & 0 & 0 \\ 0 & 0 & 0 & 0 & 1/G_{ac} & 0 \\ 0 & 0 & 0 & 0 & 0 & 1/G_{bc} \end{bmatrix} \begin{bmatrix} \sigma_a \\ \sigma_b \\ \sigma_c = 0 \\ \sigma_{ab} \\ \sigma_{ac} \\ \sigma_{bc} \end{bmatrix}$$

- Table 3.2-1 summarizes which material constants are necessary, depending on the type of element used in the analysis.

Table 3.2-1: Required material constants for the element types available in ADINA

Element type	Required material constants (these constants must be nonzero except for Poisson's ratios)
2-D solid	$E_a, E_b, E_c, v_{ab}, v_{ac}, v_{bc}, G_{ab}$
3-D solid	$E_a, E_b, E_c, v_{ab}, v_{ac}, v_{bc}, G_{ab}, G_{ac}, G_{bc}$
Plate	E_a, E_b, v_{ab}, G_{ab}
Shell	$E_a, E_b, E_c, v_{ab}, v_{ac}, v_{bc}, G_{ab}, G_{ac}, G_{bc}$



Note: (\bar{r}, \bar{s}, t) is the local Cartesian system.

The a-b and \bar{r} - \bar{s} planes are coplanar.

d is the unit projection of \mathbf{r}_{12} onto the \bar{r} - \bar{s} plane.

β is the input material angle.

Figure 3.2-4: Definition of axes of orthotropy for shell elements

3.3 Nonlinear elastic material model (for truss elements)

- For the truss element, an elastic material model is available for which the stress-strain relationship is defined as piecewise linear. Fig. 3.3-1 illustrates the definition of the stress-strain law.

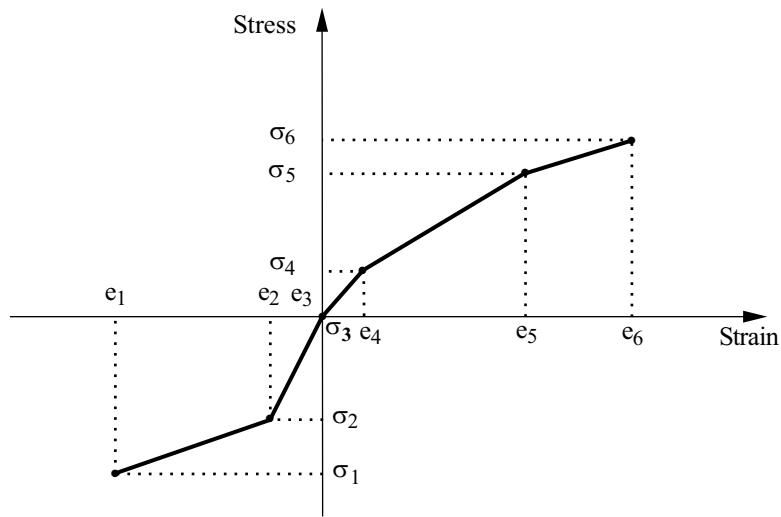


Figure 3.3-1: Nonlinear elastic material for truss element

Note that the stress is uniquely defined as a function of the strain only; hence for a specific strain ' e ', reached in loading or unloading, a unique stress is obtained from the curve in Fig. 3.3-1.

- A typical example of the truss nonlinear elastic model is shown in Fig. 3.3-2. Note that the stress-strain relation shown corresponds to a cable-like behavior in which the truss supports tensile but no compressive loading.

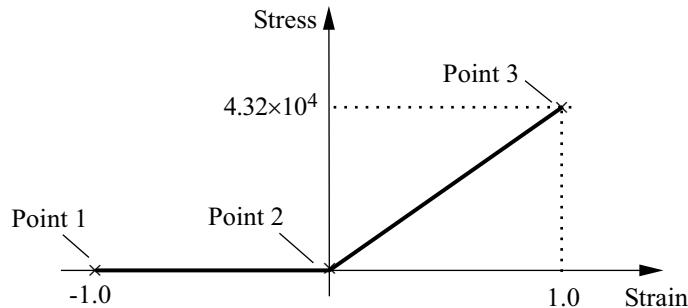


Figure 3.3-2: Example of nonlinear elastic stress-strain material model for truss

A sufficient range (in terms of the strain) must be used in the definition of the stress-strain relation so that the element strain evaluated in the solution lies within that range; i.e., referring to Fig. 3.3-1, we must have $e_1 \leq {}^t e \leq e_6$ for all t .

- The truss element with this material model is particularly useful in modeling gaps between structures. This modeling feature is illustrated in Fig. 3.3-3. Note that to use the gap element, it is necessary to know which node of one body will come into contact with which node of the other body.

ref. S.M. Ma and K.J. Bathe, "On Finite Element Analysis of Pipe Whip Problems," *Nuclear Engineering and Design*, Vol. 37, pp. 413-430, 1976.

ref. K.J. Bathe and S. Gracewski, "On Nonlinear Dynamic Analysis using Substructuring and Mode Superposition," *Computers and Structures*, Vol. 13, pp. 699-707, 1981.

Note that an alternate gap element can be obtained by the use of the 2-node truss element employing the plastic-bilinear material model, see Section 3.4.1.

A more general way of modeling contact between bodies is the use of contact surfaces, see Chapter 4.

- Other modeling features available with the truss element and this material model are shown in Figure 3.3-4.

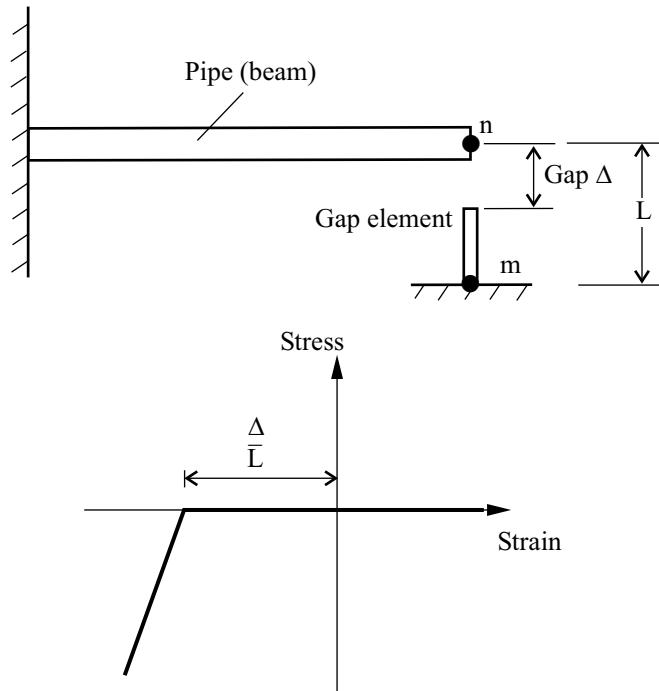


Figure 3.3-3: Modeling of gaps (pipe whip problem)

- ▶ The model shown in Fig. 3.3-4(a) corresponds to a compression-only behavior.
- ▶ The model shown in Fig. 3.3-4(b) corresponds to a tension cut-off behavior, the snapping of a cable, for example.
- ▶ The model shown in Fig. 3.3-4(c) corresponds to a behavior exhibiting both tension and compression cut-offs.

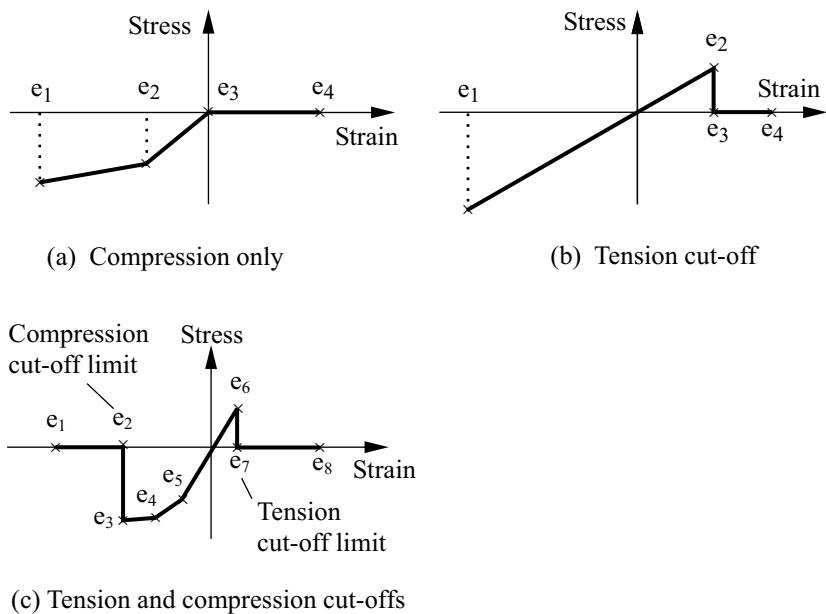


Figure 3.3-4: Various modeling features available with the nonlinear elastic truss model

3.4 Isothermal plasticity material models

*ref. KJB
Section 6.6.3*

- This section describes the following material models:

Plastic-bilinear, plastic-multilinear: von Mises model with isotropic, kinematic hardening or mixed hardening

Mroz-bilinear: von Mises model with Mroz hardening

Plastic-orthotropic: Hill yielding with bilinear proportional hardening

Ilyushin: Ilyushin model for plates

Gurson: Gurson plastic model for fracture/damage analysis

- All elastic-plasticity models use the flow theory to describe the elastic-plastic response; the basic formulations for the von Mises models are summarized on pp. 596-604, ref. KJB, and for the Mroz model in:

ref. Y.F. Dafalias and E.P. Popov, "Plastic Internal Variables Formalism of Cyclic Plasticity," *J. Appl. Mech., Trans. ASME*, Vol. 43, pp. 645-651, 1976.

The formulation for the Ilyushin model is given in the following reference:

ref. K.J. Bathe, E. Dvorkin and L.W. Ho, "Our Discrete-Kirchhoff and Isoparametric Shell Elements – An Assessment," *J. Computers and Structures*, Vol. 16, pp. 89-98, 1983.

- The Drucker-Prager material model is described in Section 3.9.2.

3.4.1 Plastic-bilinear and plastic-multilinear material models

- These material models are based on
 - ▶ The von Mises yield condition (see p. 597, ref. KJB)
 - ▶ An associated flow rule using the von Mises yield function
 - ▶ An isotropic or kinematic, bilinear or multilinear, hardening rule

Figs. 3.4-1 to 3.4-3 summarize some important features of these material models.

- These models can be used with the **truss**, **2-D solid**, **3-D solid**, **beam** (plastic-bilinear only), **iso-beam**, **shell** and **pipe** elements.

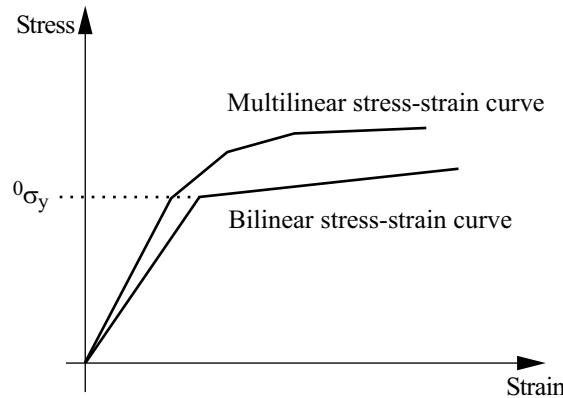


Figure 3.4-1: von Mises model

- These models can be used with the **small displacement/small strain**, **large displacement/small strain** and **large displacement/large strain** formulations. The large displacement/large strain formulation can only be used with the 2-D solid, 3-D solid and shell elements (the shell elements must be 3-node, 4-node, 9-node or 16-node single layer shell elements described entirely by midsurface nodes).

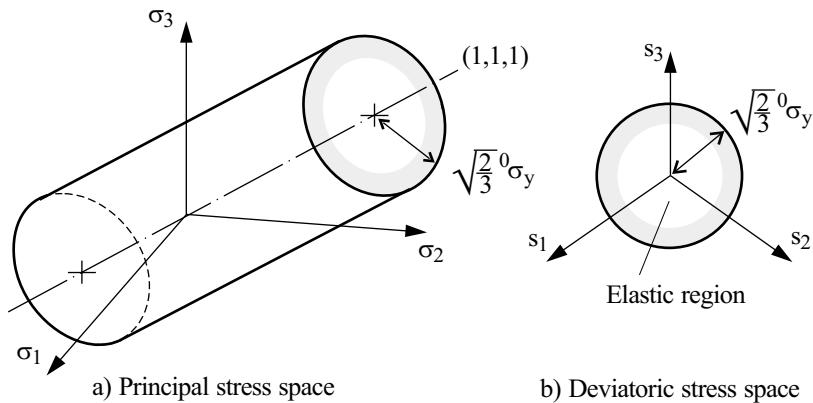


Figure 3.4-2: von Mises yield surface

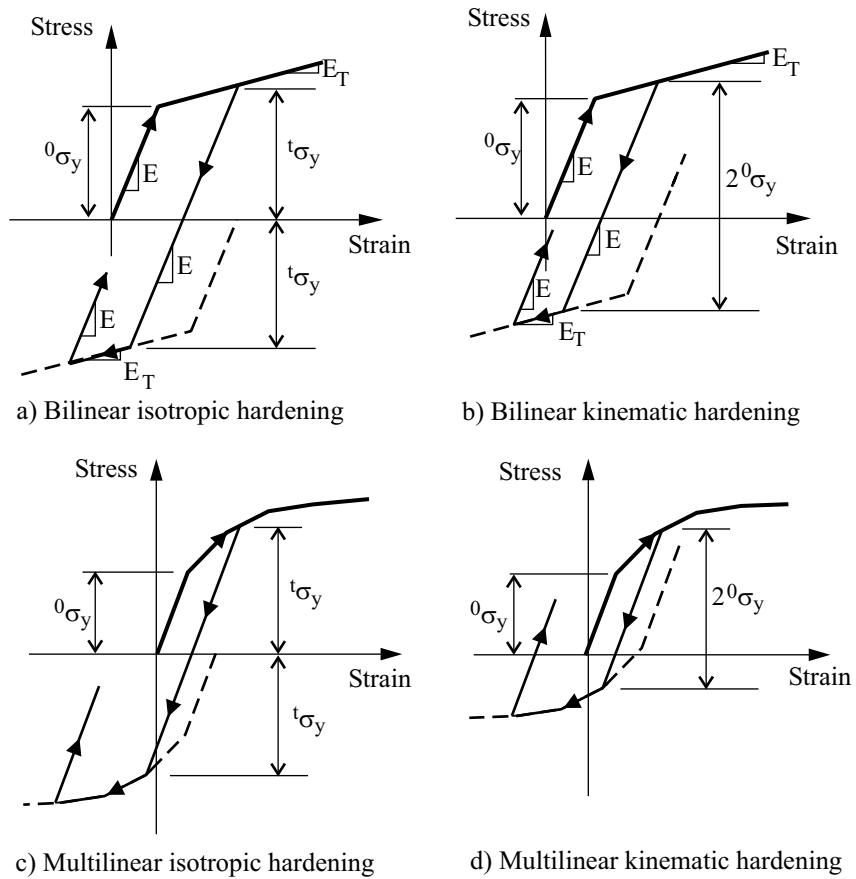


Figure 3.4-3: Isotropic and kinematic hardening

When used with the small displacement/small strain formulation, a materially-nonlinear-only formulation is employed, when used with the large displacement/small strain formulation, either a TL or a UL formulation is employed, and when used with the large displacement/large strain formulation, a ULH formulation (solid elements) or either a ULH or ULJ formulation (shell element) is employed.

- For the truss element, multilinear plasticity, you can only enter up to 7 stress-strain points in the stress-strain curve. For other elements, multilinear plasticity, there is no restriction on the number of stress-strain points in the stress-strain curve.

- Plane strain, axisymmetric or 3-D solid elements that reference these material models should also employ the mixed displacement-pressure (u/p) element formulation.
- In the von Mises model with isotropic hardening, the following yield surface equation is used:

$${}^t f_y = \frac{1}{2} ({}^t \mathbf{s} \cdot {}^t \mathbf{s}) - \frac{1}{3} {}^t \sigma_y^2 = 0$$

where ${}^t \mathbf{s}$ is the deviatoric stress tensor and ${}^0 \sigma_y^2$ the updated yield stress at time t .

In the von Mises model with kinematic hardening, the following yield surface equation is used:

$${}^t f_y = \frac{1}{2} ({}^t \mathbf{s} - {}^t \mathbf{a}) \cdot ({}^t \mathbf{s} - {}^t \mathbf{a}) - \frac{1}{3} {}^0 \sigma_y^2 = 0$$

where ${}^t \mathbf{a}$ is the shift of the center of the yield surface (back stress tensor) and ${}^0 \sigma_y^2$ is the virgin, or initial, yield stress.

In the von Mises model with mixed hardening, the following yield surface equation is used:

$${}^t f_y = \frac{1}{2} ({}^t \mathbf{s} - {}^t \mathbf{a}) \cdot ({}^t \mathbf{s} - {}^t \mathbf{a}) - \frac{1}{3} {}^t \sigma_y^2 = 0$$

where

$${}^t \sigma_y = {}^0 \sigma_y + M E_p \mathbf{e}^p$$

The back stress ${}^t \mathbf{a}$ is evolved by

$$d\mathbf{a} = C_p (1 - M) d\mathbf{e}^p$$

C_p is Prager's hardening parameter, related to the plastic modulus E_p by

$$C_p = \frac{2}{3} E_p$$

and M is the factor used in general mixed hardening ($0 < M < 1$) which can be a variable, expressed as

$$M = M_\infty + (M_0 - M_\infty) \exp(-\eta e^p)$$

The formulation for the von Mises model with mixed hardening is given in the following reference:

ref K.J. Bathe and F.J. Montáns, “On Modeling Mixed Hardening in Computational Plasticity”, *Computers and Structures*, Vol. 82, No. 6, pp. 535 - 539, 2004.

Note that the convergence might not be good when the mixed hardening parameter $\eta \neq 0$. Therefore, $\eta = 0$ is preferred.

The yield stress is a function of the effective plastic strain, which defines the hardening of the material. The effective plastic strain is defined as

$${}^t \bar{e}^{-P} = \int_0^t \sqrt{\frac{2}{3} d\mathbf{e}^p \cdot d\mathbf{e}^p}$$

in which $d\mathbf{e}^p$ is the tensor of differential plastic strain increments.

In finite element analysis, we approximate ${}^t \bar{e}^{-P}$ as the sum of all of the plastic strain increments up to the current solution time:

$${}^t \bar{e}^{-P} = \sum_{\text{all solution steps}} \Delta \mathbf{e}^p$$

where $\Delta \bar{e}^{-P} = \sqrt{\frac{2}{3} d\mathbf{e}^p \cdot d\mathbf{e}^p}$ and $\Delta \mathbf{e}^p$ is the tensor of plastic strain increments in a solution step. Because of the summation over the solution steps, we refer to the calculated value of ${}^t \bar{e}^{-P}$ as the

accumulated effective plastic strain.

- If a thermal load is applied to the structure, the thermal strains are taken into account but the material characteristics are considered to be temperature independent.

Modeling of rupture: Rupture conditions can also be modeled.

For the bilinear stress-strain curve, a maximum allowable effective plastic strain \bar{e}_A^P can be specified for the rupture condition. For the multilinear stress-strain curve, the rupture plastic strain corresponds to the effective plastic strain at the last point input for the stress-strain curve.

When rupture is reached at a given element integration point, the corresponding element is removed from the model (see Section 11.5).

There is also the option of user-supplied rupture. You code the rupture condition into one of the CURUP subroutines: CURUP2 for 2-D solid elements, CURUP3 for 3-D solid elements, CURUP4 for beam elements, CURUP5 for iso-beam elements, CURUP7 for shell elements and CURUP8 for pipe elements. These subroutines are in file ovlusr.f. ADINA provides the calculated latest stresses, total strains, thermal strains, plastic strains, creep strains, yield stress and accumulated effective plastic strain to these subroutines and these subroutines return the current rupture state.

Rate-dependent plasticity for the truss element: The elastic-plastic material models in the truss element can include strain-rate effects in dynamic analysis. In this case, the yield stress including strain-rate effects (σ_y^D) is interpolated based on the current plastic

strain rate, from the $\left(\frac{\sigma_y^D}{\sigma_y}\right)$ versus plastic strain rate relation that

you define (σ_y is the yield stress without strain rate effects).

Rate-dependent plasticity for truss, 2-D, 3-D and shell

elements: The rate dependent model in ADINA is used to simulate the increase in the yield stress with an increase in strain rate.

The rate-dependent model only applies to the isotropic plasticity models with isotropic hardening (bilinear or multilinear).

The rate-dependent model is implemented for truss, 2-D solid, 3-D solid and shell (either single-layered or multilayered) elements.

The effective yield stress including strain rate effects is

$$\sigma_y = \sigma_y^0 \left[1 + b \ln \left(1 + \frac{\dot{\varepsilon}^P}{\dot{\varepsilon}_0} \right) \right]$$

where σ_y^0 is the static yield stress, $\dot{\varepsilon}_0$ is the transition strain rate and b is the strain rate hardening parameter.

The material parameter $\dot{\varepsilon}_0$ must be directly specified. The material parameter b can either be directly specified or can be calculated using an additional user-input stress-strain curve for the stress-strain behavior at a given strain rate.

For more information, see the following reference:

ref. W.H. Drysdale and A.R. Zak, "Mechanics of Materials and Structural Theories — A Theory for Rate Dependent Plasticity", *Computers and Structures*, Vol. 20, pp. 259-264, 1985.

Modeling of gaps using the truss element: When used in conjunction with the 2-node truss element the elastic-plastic material models can be used to model gaps in the elements (the nonlinear elastic model can also be used to model gaps with the truss elements; see Section 3.3). In this case the gap elements can only resist compressive loads, i.e., gap elements have no tensile stiffness. The gap width input for each element is used to determine a strain e_{gap} (compressive strain is defined as negative). The compressive stiffness of a gap element is zero if the strain in the element is greater than or equal to e_{gap} and is nonzero when the strain is less than e_{gap} .

3.4.2 Mroz-bilinear material model

- This material model is based on
 - The von Mises yield condition
 - An associated flow rule
 - Bilinear Mroz hardening, including a rule for displacement

of the yield surface and a rule for displacement of the bounding surface

- The Mroz-bilinear model can be used with the **2-D solid** and **3-D solid** elements.
- The Mroz-bilinear model can be used with the **small displacement/small strain, large displacement/small strain and large displacement/large strain** formulations.

When used with the small displacement/small strain formulation, a materially-nonlinear-only formulation is employed, when used with the large displacement/small strain formulation, the TL formulation is employed and when used with the large displacement/large strain formulation, the ULH formulation is employed.

- Plane strain, axisymmetric or 3-D solid elements that reference this material model should also employ the mixed displacement-pressure (u/p) element formulation.
- The material behavior is characterized by a uniaxial bilinear stress-strain curve plus a bounding line.

A representation of the stress-strain curve is shown in Fig. 3.4-4. Under uniaxial loading the plastic deformation starts at stress $\sigma = {}^0\sigma_y$ and continues along the line AB until the bounding stress σ_{yB} is reached. With further stress increase, plastic deformation continues and the stress-strain relation follows the bounding line (segment BC in the figure). The material response corresponding to deformation ABC is the same as the plastic-bilinear model with kinematic or isotropic hardening (Section 3.4.1). If the tangent modulus E_{TB} of the bounding surface is equal to zero, the deformation on the segment BC corresponds to a perfectly plastic von Mises material.

The uniaxial plastic deformation is represented in the deviatoric stress space in Fig. 3.4-5(a). The stress path A'B'C' corresponds to the stress-strain path ABC. After the yield surface touches the bounding surface at point B', the two surfaces translate in the stress space with the common stress point moving along the line B'C'.

- When the load is reversed, plastic deformation starts at point D and follows the segments DF and FG, see Fig. 3.4-4.

In the reverse loading, the material response differs from the response corresponding to the plastic-bilinear model with isotropic or kinematic hardening. For example, in the case of kinematic hardening, the stress-strain relation in the reverse loading follows the line $D\bar{F}$ with the slope E_{TB} while in the Mroz hardening, the reverse plastic deformation starts with the slope E_T and continues with that slope until the bounding line at point F is reached. Also, in all subsequent reverse loadings, yielding starts with the tangent modulus E_T (as at point H in Fig. 3.4-4). The stress path $F'G'$ corresponding to segment FG, and the yield and bounding surfaces are shown in Fig. 3.4-5(b).

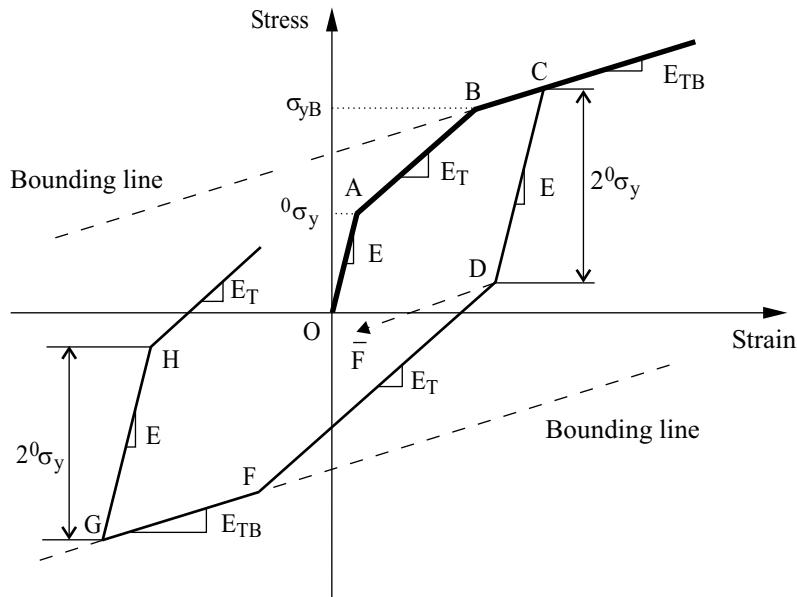


Figure 3.4-4: Mroz model; bilinear stress strain curve

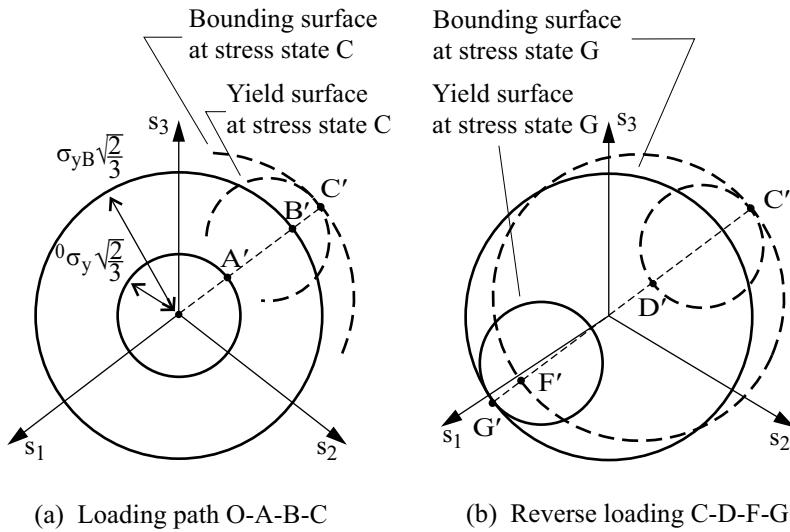


Figure 3.4-5: Mroz model behavior in deviatoric stress space

- In the case of a general nonradial loading, some additional assumptions are used. The model stress-strain behavior is described as follows:

- The yield surface

$${}^t f_y = \frac{1}{2} ({}^t \mathbf{s} - {}^t \mathbf{a}) \cdot ({}^t \mathbf{s} - {}^t \mathbf{a}) - \frac{1}{3} {}^0 \sigma_y^2 = 0$$

and the bounding surface

$${}^t f_{yB} = \frac{1}{2} ({}^t \mathbf{s}_B - {}^t \mathbf{b}) \cdot ({}^t \mathbf{s}_B - {}^t \mathbf{b}) - \frac{1}{3} {}^0 \sigma_{yB}^2 = 0$$

are given in Fig. 3.4-6 where ${}^t \mathbf{a}$ is the back stress and ${}^t \mathbf{b}$ defines the position of the bounding surface.

- The plastic flow rule is

$${}^t \dot{\mathbf{e}}^P = {}^t \dot{\Lambda} ({}^t \mathbf{s} - {}^t \mathbf{a})$$

where ${}^t\Lambda$ is the plastic multiplier (positive scalar).

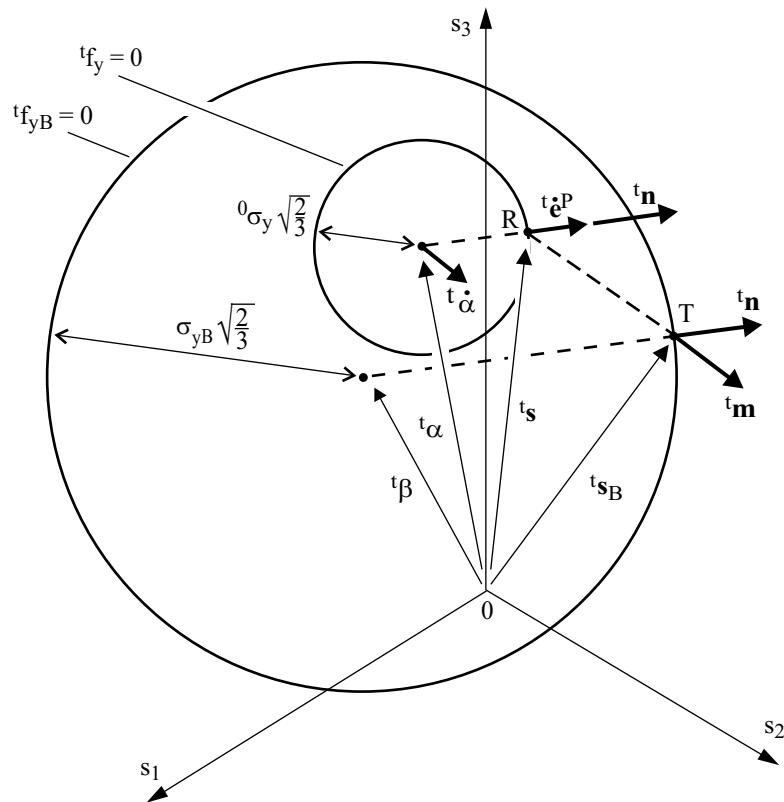


Figure 3.4-6: Mroz model in case of general loading conditions

- The constitutive relation governing the change of the back stress '**a**' is given by

$${}^t\ddot{\mathbf{a}} = {}^t\bar{C}_p \sqrt{\frac{2}{3}} \frac{d^t \bar{e}^P}{dt} {}^t\mathbf{m}$$

with

$${}^t\bar{C}_p = \frac{2}{3} \frac{{}^tE_p}{{}^t\mathbf{m} \cdot {}^t\mathbf{n}}$$

$${}^tE_p = \frac{E {}^tE_T}{E - E_T}$$

Here $\frac{d'{}^t\bar{e}^P}{dt}$ represents the rate of the effective plastic strain,

$'E_p$ is the plastic modulus, E is the elastic modulus and $'E_T$ is the hardening modulus. The variables ' \mathbf{m} ' and ' \mathbf{n} ' are unit normals shown in Fig. 3.4-6. The above definition of the modulus $'\bar{C}_p$ is based on the assumption that the work per unit time of the back stress corresponding to the plastic straining in general loading conditions is equal to the same quantity in the radial loading conditions. The direction of ' $\dot{\alpha}$ ' is such that the yield and bounding surfaces have the same normal at the contact point.

- ▶ The bounding surface moves in the stress space only when the yield surface is in contact with the bounding surface and the loading continues in the direction of the then common normal (see Figs. 3.4-5(b) and 3.4-6, i.e. $'\dot{\beta} = {}^t\dot{\alpha}$).
- ▶ Note that when the tangent modulus of the bounding surface E_{TB} is equal to zero, the bounding surface does not move in the stress space. In this case, the effective stress $'\bar{\sigma}$ remains smaller than σ_{yB} for any loading conditions.
- If a thermal loading is applied to the structure, thermal strains are taken into account, but the material characteristics are considered to be temperature independent.
- Rupture conditions can be modeled. When a positive value \bar{e}_A^P is input for the maximum allowable effective plastic strain, the program compares the accumulated effective plastic strain $'\bar{e}^P$ with \bar{e}_A^P for each material point used in the analysis and over the

whole history of deformation. It is considered that the rupture at a material point occurs when ' \bar{e}^P ' is greater than \bar{e}_A^P . When rupture is reached at a given element integration point, the corresponding element is removed from the model. (see Section 11.5).

The user-supplied rupture option is also available, see Section 3.4.1.

3.4.3 Plastic-orthotropic material model

- The plastic-orthotropic model is based on:
 - ▶ The Hill yield condition
 - ▶ An associated flow rule
 - ▶ A bilinear proportional hardening rule
- This model can be used with the **2-D solid**, **3-D solid** and **shell** elements.
- The plastic-orthotropic model can be used with the **small displacement/small strain**, **large displacement/small strain** and **large displacement/large strain** formulations. Small strains are always assumed with the 2-D solid and 3-D solid elements. The large displacement/large strain formulation can be used with the shell elements (which must be 3-node, 4-node, 9-node or 16-node single layer elements described entirely by midsurface nodes).

When used with the small displacement/small strain formulation, a materially-nonlinear-only formulation is employed and when used with the large displacement/small strain formulation, the TL formulation is employed. When used with the large displacement/large strain formulation, the ULJ formulation is employed.

- The Hill yield condition is given by:

$$\begin{aligned} F(\sigma_{bb} - \sigma_{cc})^2 + G(\sigma_{cc} - \sigma_{aa})^2 + H(\sigma_{aa} - \sigma_{bb})^2 \\ + 2L\sigma_{ab}^2 + 2M\sigma_{bc}^2 + 2N\sigma_{ac}^2 - 1 = 0 \end{aligned}$$

where (a, b, c) are the material principal axes, and F, G, H, L, M, N are material constants. These constants are given by

$$F = \frac{1}{2} \left(\frac{1}{Y^2} + \frac{1}{Z^2} - \frac{1}{X^2} \right)$$

$$G = \frac{1}{2} \left(\frac{1}{Z^2} + \frac{1}{X^2} - \frac{1}{Y^2} \right)$$

$$H = \frac{1}{2} \left(\frac{1}{X^2} + \frac{1}{Y^2} - \frac{1}{Z^2} \right)$$

$$L = \frac{1}{2Y_{ab}^2}; \quad M = \frac{1}{2Y_{bc}^2}; \quad N = \frac{1}{2Y_{ac}^2}$$

where X, Y, Z , are the yield stresses in the material directions a, b, c , and Y_{ab}, Y_{bc}, Y_{ac} are the yield stresses for pure shear in the planes $(a,b), (b,c)$, and (a,c) (see Figure 3.4-7).

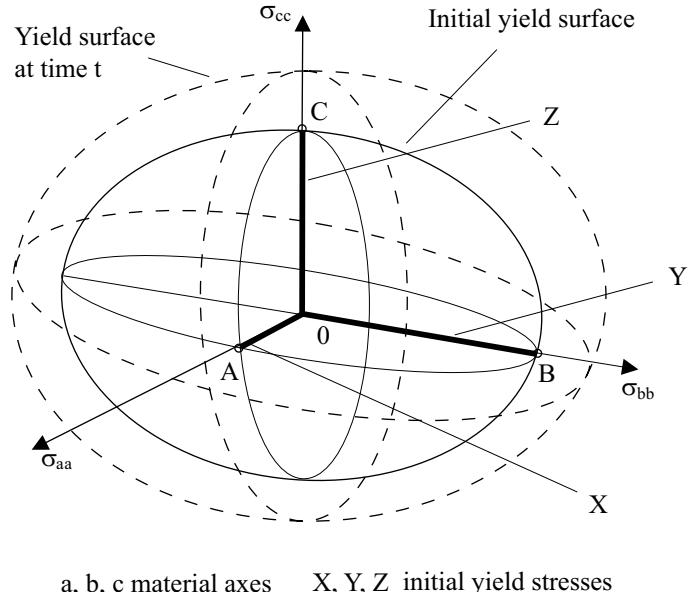


Figure 3.4-7: Orthotropic yield surface

There are three ways to specify the material constants F, G, H, L, M, N :

- ▶ Enter the yield stresses, then ADINA computes the material constants using the equations given above.
- ▶ Enter the Lankford coefficients r_0, r_{45}, r_{90} ; then

$$\begin{aligned} F &= \frac{r_0}{r_{90} \cdot (r_0 + 1)} & 2L &= \frac{(r_0 + r_{90}) \cdot (2r_{45} + 1)}{r_{90} \cdot (r_0 + 1)} \\ G &= \frac{1}{(r_0 + 1)} & M &= 1.5 \\ H &= \frac{r_0}{(r_0 + 1)} & N &= 1.5 \end{aligned}$$

in which r_0 is the Lankford coefficient for the (rolling) direction, r_{45} is the Lankford coefficient for the direction 45° to the rolling direction and r_{90} is the Lankford coefficient for the direction 90° to the rolling direction (the transverse direction).

- ▶ Directly specify F, G, H, L, M, N .
- Please note that F, G, H, L, M, N must all be positive. In addition, for 2D axisymmetric elements, the coefficients must satisfy the conditions $F = G$, $M = N$, $L - F = 2H$.
- Hardening can be specified in three ways:
 - ▶ Enter the universal plastic modulus E_u^p . In this case the plastic moduli for directions a, b, c and planes $(a,b), (a,c), (b,c)$ are given by $E_a^p = E_b^p = E_c^p = E_u^p$ and $E_{ab}^p = E_{ac}^p = E_{bc}^p = \frac{E_u^p}{\sqrt{3}}$.
 - ▶ Enter the independent moduli E_a^T, E_b^T, E_c^T for the a, b, c directions and for the $(a,b), (a,c), (b,c)$ planes. In this case, E_u^p is equal to

$$\sqrt{\frac{1}{2} \left(\frac{1}{3} \left(E_a^{p^2} + E_b^{p^2} + E_c^{p^2} \right) + E_{ab}^{p^2} + E_{ac}^{p^2} + E_{bc}^{p^2} \right)}$$

where

$$E_i^p = \frac{E_i E_i^T}{E_i - E_i^T} \quad i = a, b, c$$

and

$$E_{ij}^p = \frac{E_{ij} E_{ij}^T}{E_{ij} - E_{ij}^T} \quad ij = ab, ac, bc$$

- ▶ Enter the parameters of a stress-strain curve which is used to calculate the universal plastic modulus. The stress-strain curve is of the form

$$\sigma = C \cdot (\varepsilon_0 + \varepsilon)^n$$

in which σ is the yield stress, ε is the plastic strain, ε_0 is the initial plastic strain, n is the strain hardening coefficient and C is a constant.

- When the 2-D solid element (plane stress or plane strain) or shell element is used, material axes a and b must lie in the plane of the element. When the 2-D solid element (axisymmetric) is used, material axis a must be aligned with the Y axis and material axis c must be aligned with the Z axis.
- If a thermal load is applied to the structure, the thermal strains are taken into account but the material characteristics are considered to be temperature independent.
- Rupture conditions can also be modeled: a maximum allowable effective plastic strain $\bar{\varepsilon}_A^P$ can be specified for the rupture condition. When rupture is reached at a given element integration point, the corresponding element is removed from the model (see Section 11.5).

The user-supplied rupture option can also be used (see Section 3.4.1).

- The effective-stress-function algorithm is used to calculate stresses and plastic strains when plasticity occurs.
- Note that orthotropic proportional hardening reduces to isotropic hardening for the appropriate values of the elastic and tangent moduli.

3.4.4 Ilyushin material model

- The Ilyushin model is based on
 - ▶ The Ilyushin yield condition (see Fig. 3.4-8)
 - ▶ An associated flow rule using the Ilyushin function
 - ▶ Bilinear isotropic hardening

The Ilyushin model is only available for the **plate** element.

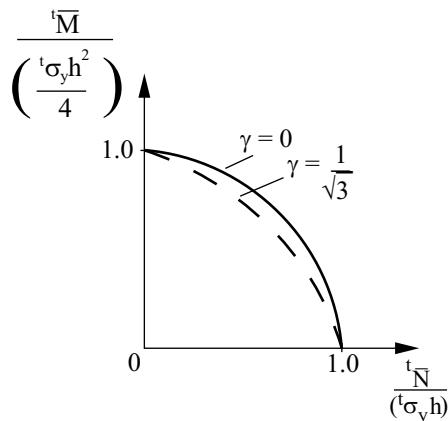
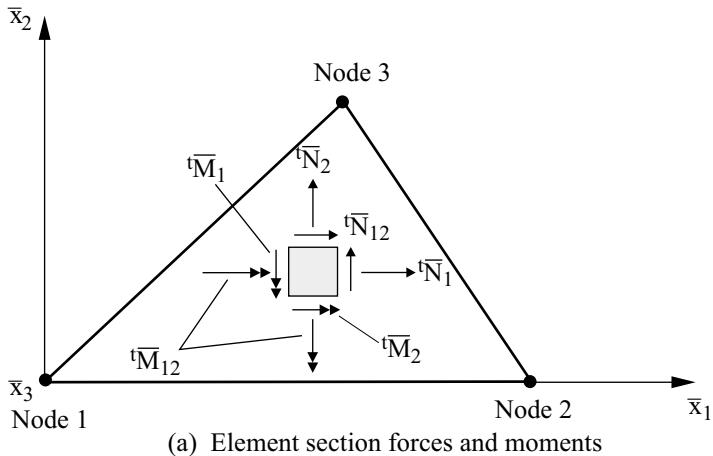
- The Ilyushin model can be used with the **small displacement** or **large displacement** formulations. In all cases the strains are assumed to be small.

When used with the small displacement formulation, a materially-nonlinear-only formulation is employed, and when used with the large displacement formulation, the UL formulation is employed.

- The Ilyushin model is formulated in terms of the stress resultants, the membrane forces, and bending moments.
- The Ilyushin yield surface equation is given by

$${}^t f_y = {}^t \bar{Q}_M + {}^t \bar{Q}_B + \left| {}^t \bar{Q}_{MB} \right| \gamma - {}^t \sigma_y^2 = 0$$

where γ is an input parameter, ${}^t \sigma_y$ is the yield stress, and



(b) Uniaxial yield condition

Figure 3.4-8: Ilyushin model for use with the plate element

$${}^t\bar{Q}_M = \frac{1}{h^2} \left\{ \left({}^t\bar{N}_1 \right)^2 + \left({}^t\bar{N}_2 \right)^2 - {}^t\bar{N}_1 \cdot {}^t\bar{N}_2 + 3 \left({}^t\bar{N}_{12} \right)^2 \right\}$$

$${}^t\bar{Q}_B = \frac{16}{h^4} \left\{ \left({}^t\bar{M}_1 \right)^2 + \left({}^t\bar{M}_2 \right)^2 - {}^t\bar{M}_1 \cdot {}^t\bar{M}_2 + 3 \left({}^t\bar{M}_{12} \right)^2 \right\}$$

$${}^t \bar{Q}_{MB} = \frac{12}{h^3} \left\{ \frac{1}{3} \left({}^t \bar{N}_1 \cdot {}^t \bar{M}_1 + {}^t \bar{N}_2 \cdot {}^t \bar{M}_2 \right) + {}^t \bar{N}_{12} \cdot {}^t \bar{M}_{12} - \frac{1}{6} \left({}^t \bar{N}_1 \cdot {}^t \bar{M}_2 + {}^t \bar{N}_2 \cdot {}^t \bar{M}_1 \right) \right\}$$

- Since the Ilyushin model is formulated using stress resultants, no integration through the element thickness is performed in the calculation of element matrices.

Thus, considering the yielding of a plate element at an integration point, the element is elastic until the entire section becomes plastic. This delay in the change of state from elastic to plastic can be observed when comparing the response predicted by the plate element and the shell element; see Fig. 6 of the following reference:

ref. K.J. Bathe, E. Dvorkin and L.W. Ho, "Our Discrete-Kirchhoff and Isoparametric Shell Elements – An Assessment," *J. Computers and Structures*, Vol. 16, pp. 89-98, 1983.

- In an elasto-plastic plate/shell analysis using the Ilyushin model, the membrane forces and bending moments are, in general, coupled. Hence, even though the element constant strain conditions still hold in the membrane representation, constant stress conditions may not be predicted.

3.4.5 Gurson material model

- The Gurson model is intended for use in fracture/damage analysis. The purpose of the Gurson plastic model is to predict ductile crack growth by void growth and coalescence. This is a micromechanical model based approach.
- The Gurson model is available for the **2-D solid** and **3-D solid** elements.
- The Gurson model can be used with the **small displacement/small strain, large displacement/small strain** and **large displacement/large strain** formulations.

When used with the small displacement/small strain formulation, a materially-nonlinear-only formulation is employed, when used with the large displacement/small strain formulation, the

TL formulation is employed and when used with the large displacement/large strain formulation, the ULH formulation is employed.

- In the Gurson model, the yield function has the form

$$\phi = \left(\frac{q}{\sigma_0} \right)^2 + 2q_1 f^* \cosh \left(-\frac{3}{2} \frac{q_2 p}{\sigma_0} \right) - \left(1 + q_3 f^{*2} \right) = 0$$

in which ϕ is the yield function, q is the von Mises stress, σ_0 is the equivalent tensile flow stress, p is the pressure (positive in compression), f^* is based on the void volume fraction and q_1, q_2, q_3 are the Tvergaard constants. Note that this reduces to the von Mises yield function when f^* equals zero. f^* is defined as

$$f^* = \begin{cases} f & \text{for } f \leq f_c \\ f_c + \frac{f_u^* - f_c}{f_F - f_c} (f - f_c) & \text{for } f > f_c \end{cases}$$

in which f is the void volume fraction, f_c is the critical void volume fraction, f_F is the void volume fraction at final failure and $f_u^* = 1/q_1$.

- Initially, the void volume fraction is f_0 , which must be input. The increase of void volume fraction is controlled by

$$df = (1-f) d\varepsilon_{kk}^p + A d\bar{\varepsilon}^p$$

in which $d\varepsilon_{kk}^p$ is the sum of the normal plastic strain components, $\bar{\varepsilon}^p$ is the equivalent plastic strain and A is the void nucleation intensity. A is defined as

$$A = \frac{f_N}{S_N \sqrt{2\pi}} \exp \left[-\frac{1}{2} \left(\frac{\bar{\varepsilon}^p - \varepsilon_N}{S_N} \right)^2 \right]$$

in which f_N is the volume fraction of void nucleating particles, ε_N is the mean void nucleation burst strain and S_N is the corresponding standard deviation.

- The uniaxial stress-strain curve is

$$\frac{\sigma_0}{\sigma_y} = \left(\frac{\sigma_0}{\sigma_y} + \frac{3G}{\sigma_y} \bar{\varepsilon}^p \right)^N$$

where σ_y is the tensile yield stress of the matrix, G is the shear modulus and N is a material constant (typically on the order of 0.1).

- For more information, see the following reference:

ref. N. Aravas, "On the numerical integration of a class of pressure-dependent plasticity models", *Int. J. Num. Meth. in Engng.*, Vol. 24, 1395-1416 (1987).

3.5 Thermo-elastic material models

- The thermo-isotropic and thermo-orthotropic material models are used to calculate stress distributions due to imposed temperatures. For thermal stress analysis, the material moduli can be constant or temperature-dependent.
- The thermo-isotropic model is available for the **truss**, **2-D solid**, **3-D solid**, **iso-beam**, **shell** and **pipe** elements.
- The thermo-orthotropic model is available for the **2-D solid**, **3-D solid** and **shell** elements.
- Both models can be used with the **small displacement** and **large displacement** formulations. In all cases the strains are assumed to be small.

When used with the small displacement formulation, a

materially-nonlinear-only formulation is employed, and when used with the large displacement formulation, either the TL or UL formulation is employed.

- In the data input for the analysis, the nodal point temperatures must be defined for all time steps (reading these temperatures from a file created by ADINA-T, reading these temperatures from a mapping file or defining them by time functions). If the model includes axisymmetric shell or shell elements, the nodal point temperature gradients must also be defined for all time steps. See Section 5.6 for more information on prescribing temperatures and temperature gradients.
- For these models, the elastic moduli, the shear moduli, the Poisson's ratios and the coefficients of thermal expansion are input as piecewise linear functions of the temperature, as illustrated in Fig. 3.5-1. Linear interpolation is used to calculate the material properties between input points.

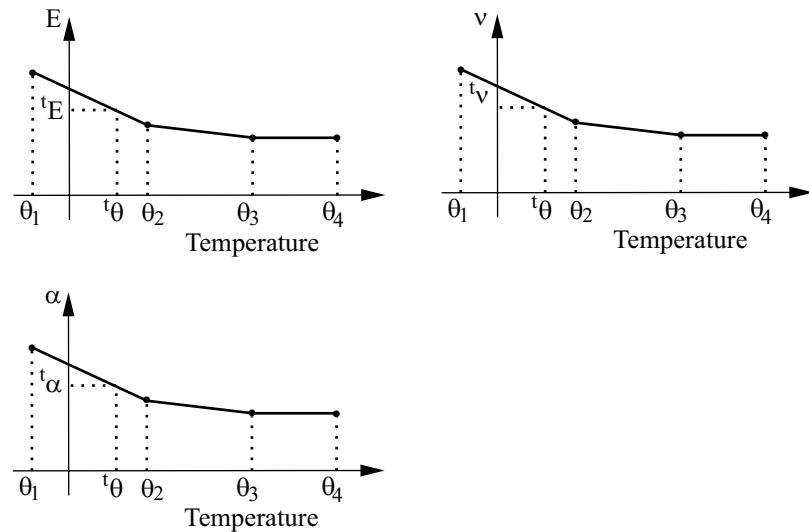


Figure 3.5-1: Variation of material properties for thermo-elastic model

- The thermal strains at a given element integration point are calculated at time t using

$${}^t e_{ij}^{TH} = {}^t \bar{\alpha} \left({}^t \theta - {}^0 \theta \right) \delta_{ij} \quad (3.5-1)$$

where

$${}^t \bar{\alpha} = \frac{1}{({}^t \theta - {}^0 \theta)} \left(\alpha({}^t \theta)({}^t \theta - \theta_{REF}) - \alpha({}^0 \theta)({}^0 \theta - \theta_{REF}) \right) \quad (3.5-2)$$

and where the temperatures ${}^t \theta$ and ${}^0 \theta$ are obtained by interpolation from the nodal point temperatures (and temperature gradients), ${}^t \alpha$ is obtained as a function of ${}^t \theta$ or ${}^0 \theta$ from the input definition of the material parameters, ${}^0 \theta$ is the initial temperature corresponding to zero thermal strains at the integration point, θ_{REF} is the material reference temperature, and δ_{ij} is the Kronecker delta ($\delta_{ij} = 1$ for $i = j$ and $\delta_{ij} = 0$ for $i \neq j$).

- Equations (3.5-1) and (3.5-2) are derived as follows: Suppose that, from experimental data, the dependence of the length of a bar as a function of temperature is obtained, as shown in Figure (3.5-2).

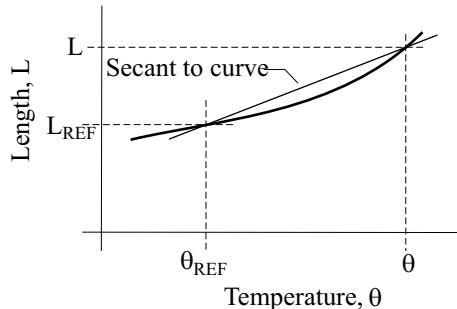


Figure 3.5-2: Length of bar vs. temperature

The thermal strain with respect to the reference length may be calculated as

$$\varepsilon_{th} = \frac{L - L_{REF}}{L_{REF}}$$

Then we define the mean coefficient of thermal expansion for a given temperature as follows:

$$\alpha(\theta) = \frac{\varepsilon_{th}(\theta)}{\theta - \theta_{REF}}$$

With this definition, the secant slope in Figure (3.5-2) is $L_{REF} \alpha(\theta)$.

Now, in ADINA, we assume that the thermal strains are initially zero. To do this, we subtract off the thermal strain corresponding to ${}^0\theta$ to obtain

$${}^t\varepsilon_{th} = \alpha({}^t\theta)({}^t\theta - \theta_{REF}) - \alpha({}^0\theta)({}^0\theta - \theta_{REF})$$

and this may be rewritten as

$${}^t\varepsilon_{th} = {}^t\bar{\alpha}({}^t\theta - {}^0\theta)$$

in which

$${}^t\bar{\alpha} = \frac{1}{{}^t\theta - {}^0\theta} (\alpha({}^t\theta)({}^t\theta - \theta_{REF}) - \alpha({}^0\theta)({}^0\theta - \theta_{REF}))$$

Notice that if the mean coefficient of thermal expansion is constant, then θ_{REF} no longer enters into the definition of ${}^t\bar{\alpha}$. In general, when the mean coefficient of thermal expansion is not constant, θ_{REF} must be chosen based on knowledge of the experiment used to determine $\alpha(\theta)$ (for the same material curve, different choices of θ_{REF} yield different values of $\alpha(\theta)$).

The materials that accept mean coefficients of thermal expansion all accept the reference temperature as an input parameter.

- For the evaluation of the temperatures ${}^t\theta$ and ${}^0\theta$ at the integration point considered, the isoparametric interpolation functions h_i are used; e.g., in two-dimensional analysis we have

$$\theta = \sum_{i=1}^q h_i \theta_i$$

where θ_i is the temperature at element nodal point i (see Fig. 3.5-3). Note that when higher-order elements are used the temperatures at the integration point can be significantly different from the values at the nodal point (for example negative although all nodal point temperatures are greater than or equal to zero).

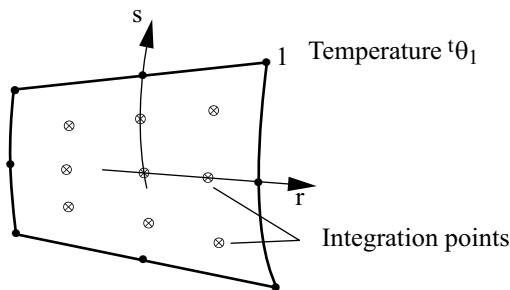


Figure 3.5-3: Interpolation of temperature at integration points

- For axisymmetric shell and shell elements, the temperature at an integration point is established using the temperatures at the midsurface nodes and the temperature gradients if they are specified through input data. Note that the temperature gradient at a shell midsurface node is defined in the direction of the mid-surface director vector at that node, see Section 2.7.3.

For transition shell elements, no temperature gradients are associated with the top/bottom nodes and only the temperatures at the top/bottom nodes are used to evaluate the temperatures at the integration points.

3.6 Thermo-elasto-plasticity and creep material models

- This section describes the **thermo-plastic**, **creep**, **plastic-creep**, **multilinear-plastic-creep**, **creep-variable**, **plastic-creep-variable**, **multilinear-plastic-creep-variable** and **irradiation creep** material models.

ref. KJB
Section 6.6.3

- The thermo-elasto-plasticity and creep models include the effects of
 - ▶ Thermal strains, ${}^t e_{rs}^{TH}$
 - ▶ Time-independent plastic strains, ${}^t e_{rs}^P$
 - ▶ Time-dependent creep strains, ${}^t e_{rs}^C$

and the constitutive relation used is

$${}^t \sigma_{ij} = {}^t C_{ijrs}^E \left({}^t e_{rs}^P - {}^t e_{rs}^C - {}^t e_{rs}^{TH} \right)$$

where ${}^t \sigma_{ij}$ is the stress tensor at time t and ${}^t C_{ijrs}^E$ is the elasticity tensor at the temperature corresponding to time t . The tensor ${}^t C_{ijrs}^E$ can be expressed in terms of Young's modulus $'E$ and Poisson's ratio $'v$ which are both temperature-dependent.

- The irradiation creep model includes the effects of
 - ▶ Thermal strains, ${}^\phi e_{rs}^{TH}$
 - ▶ Irradiation strains, ${}^\phi e_{rs}^W$
 - ▶ Neutron fluence-dependent creep strains, ${}^\phi e_{rs}^C$

and the constitutive relation used is

$${}^\phi \sigma_{ij} = {}^\phi C_{ijrs}^E \left({}^\phi e_{rs}^C - {}^\phi e_{rs}^W - {}^\phi e_{rs}^{TH} \right)$$

where ${}^\phi \sigma_{ij}$ is the stress tensor corresponding to the neutron fluence ϕ at time t and ${}^\phi C_{ijrs}^E$ is the elasticity tensor corresponding to the neutron fluence ϕ at time t . The tensor ${}^\phi C_{ijrs}^E$ can be expressed in terms of Young's modulus ${}^\phi E$ which also corresponds to the neutron fluence ϕ at time t .

- These material models can be used with the **truss**, **2-D solid**, **3-D solid**, **iso-beam**, **shell** and **pipe** elements, with the exception of the creep-irradiation model which can only be used with **2-D solid** and **3-D solid** elements.
- These models can be used with the **small displacement/small strain**, **large displacement/small strain** and **large displacement/large strain** formulations. The large displacement/large strain formulation can only be used with the 2-D solid and 3-D solid elements.

When used with the small displacement/small strain formulation, a materially-nonlinear-only formulation is employed, when used with the large displacement/small strain formulation, either a TL or a UL formulation is employed and when used with the large displacement/large strain formulation, the ULH formulation is employed.

- Plane strain, axisymmetric or 3-D solid elements that reference these material models should also employ the mixed displacement-pressure (u/p) element formulation.
- Note that the constitutive relations for the thermal, plastic and creep strains are independent of each other; hence the only interaction between the strains comes from the fact that all strains affect the stresses. Figure 3.6-1 summarizes the constitutive description for a one-dimensional stress situation and a bilinear stress-strain curve.

Since there is no direct coupling in the evaluation of the different strain components, we can discuss the calculation of each strain component independently.

- ref. M.D. Snyder and K.J. Bathe, "A Solution Procedure for Thermo-Elastic-Plastic and Creep Problems," *J. Nuclear Eng. and Design*, Vol. 64, pp. 49-80, 1981.
- ref. M. Kojić and K.J. Bathe, "The Effective-Stress-Function Algorithm for Thermo-Elasto-Plasticity and Creep," *Int. J. Numer. Meth. Engng.*, Vol. 24, No. 8, pp. 1509-1532, 1987.

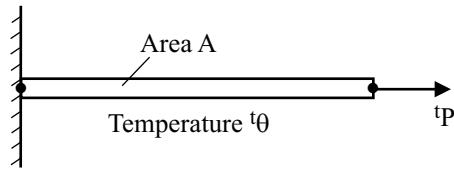
- Rupture of the material based on the following three-invariant criterion can be specified:

$$\alpha J_3(' \sigma) + \beta J_1(' \sigma) + (1 - \alpha - \beta) J_2(' \sigma) \leq \bar{\sigma}_u$$

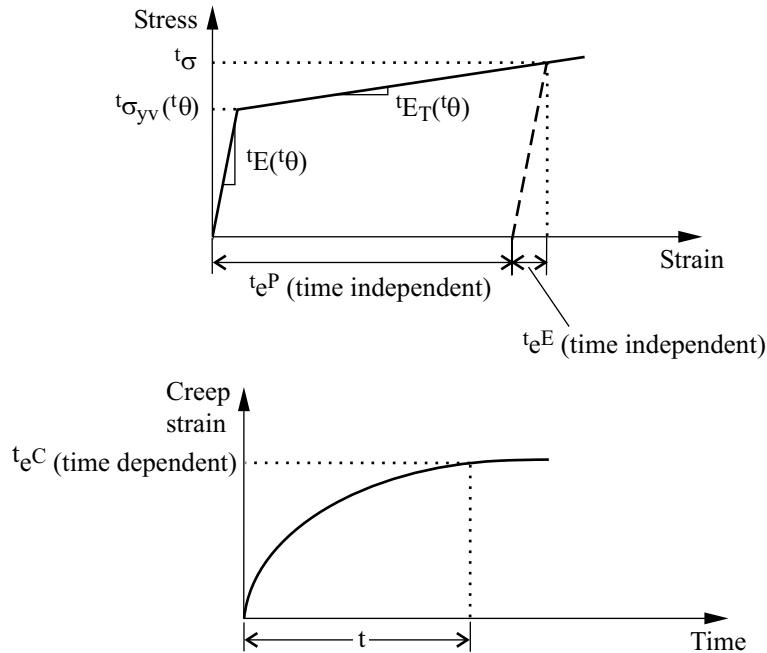
in which $' \sigma$ is the stress tensor, J_1 , J_2 , J_3 are the first, second and third stress tensor invariants, α , β are material properties (input data) and $\bar{\sigma}_u$ is the ultimate effective stress (input data).

This rupture criteria can be used in combination with other rupture criteria described later in this section.

When rupture is reached at a given element integration point, the corresponding element is removed from the model (see Section 11.5).



(a) Model problem of truss element under constant load



(b) Strains considered in the model

Figure 3.6-1: Thermo-elasto-plasticity and creep constitutive description in one-dimensional analysis

- The nodal point temperatures (and possibly temperature gradients) are input to ADINA as described in Section 5.6.

3.6.1 Evaluation of thermal strains

- The thermal strains are calculated from the prescribed nodal point temperatures with the coefficient of thermal expansion being temperature dependent. The procedure used is the same as for the thermo-isotropic material model, see Section 3.5.
- The thermal strains of the irradiation creep model are dependent on temperature and neutron fluence. The value of thermal strain is determined using the Shifting rule described in Section 3.6.5.

3.6.2 Evaluation of plastic strains

- Plasticity effects are included in the thermo-plastic, plastic-creep, multilinear-plastic-creep, plastic-creep-variable and multilinear-plastic-creep-variable material models. For the material models that have the word "multilinear" in their names, the stress-strain curve is multilinear, otherwise the stress-strain curve is bilinear.
- The plastic strains are calculated using the von Mises plasticity model (see Section 3.4.1) with temperature-dependent material parameters (Young's modulus, strain hardening modulus, Poisson's ratio, yield stress, etc...). Figure 3.6-2 shows the assumptions used for the dependence of the material parameters on temperature (in the case of a bilinear stress-strain law). (Note that the plastic modulus

$$E_P = E \cdot E_T / (E - E_T)$$

is interpolated as a function of temperature, not the tangent modulus E_T .) Hence, the yield function is in isotropic hardening

$${}^t f_y = \frac{1}{2} {}^t \mathbf{s} \cdot {}^t \mathbf{s} - \frac{1}{3} {}^t \sigma_{yv}^2$$

and in kinematic hardening

$${}^t f_y = \frac{1}{2} ({}^t \mathbf{s} - {}^t \mathbf{a}) \cdot ({}^t \mathbf{s} - {}^t \mathbf{a}) - \frac{1}{3} {}^t \sigma_{yv}^2$$

where $'\mathbf{s}$ is the deviatoric stress tensor, $'\sigma_{yv}$ is the virgin yield stress corresponding to temperature $'\theta$ (see Fig. 3.6-2) and α is the shift of the stress tensor due to kinematic hardening. In the case of multilinear stress-strain curves, the yield curves are interpolated during the solution as shown in Fig. 3.6-3.

- The expressions for plastic strain increments resulting from the flow theory are $de_{ij}^P = d\lambda 's_{ij}$ for isotropic hardening and $de_{ij}^P = d\lambda ('s_{ij} - '\alpha_{ij})$ for kinematic hardening, in which $d\lambda$ is the plastic multiplier (positive scalar) which can be determined from the yield condition $'f_y = 0$. In the case of kinematic hardening, we express the change of the yield surface position in the form

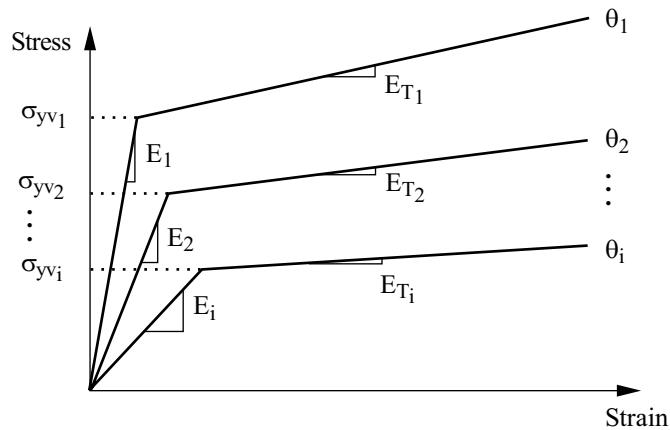
$$d\alpha_{ij} = 'C de_{ij}^P$$

where $'C$ is the modulus

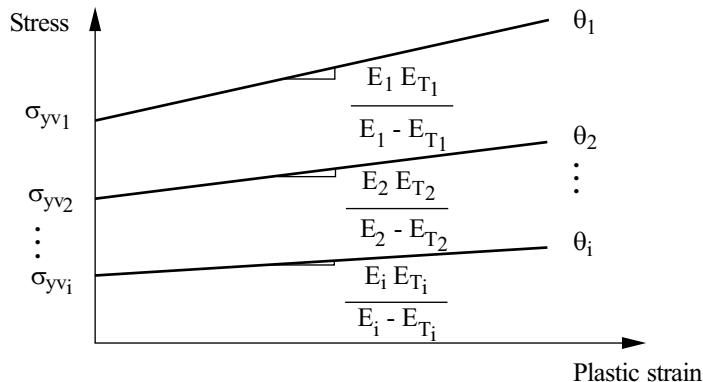
$$'C = \frac{2}{3} \frac{'E 'E_T}{'E - 'E_T}$$

In the case of multilinear yield curves, $'E_T$ represents the tangent modulus of the segment on the yield curve corresponding to the accumulated effective plastic strain $'\bar{e}^P$.

- Care should be exercised in the use of this model in kinematic hardening conditions. Namely, nonphysical effects can result when the variation in E_T as a function of temperature is large. Hence, it is recommended to use this model only when the variation in E_T as a function of temperature is small.
- Rupture due to plasticity can be modeled using this material model. When using a bilinear stress-strain curve, a maximum allowable effective plastic strain \bar{e}_A^P can be input. When using a multilinear stress-strain curve, the maximum allowable effective plastic strain at a given temperature corresponds to the last point on the yield curve at that temperature (see Fig. 3.6-3).

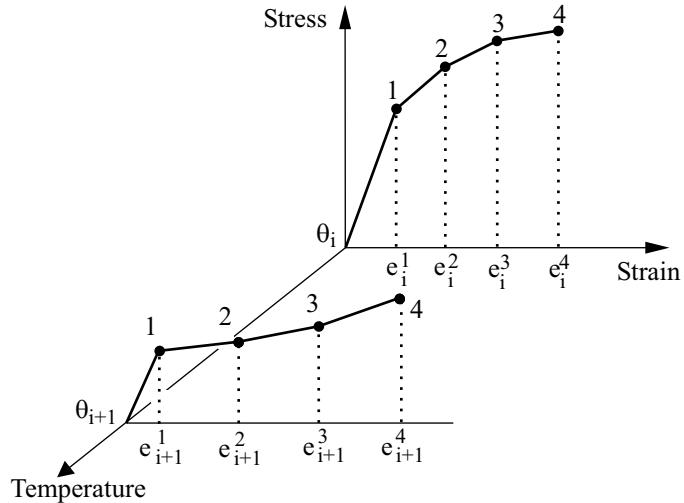


(a) Uniaxial relation of stress versus total strain

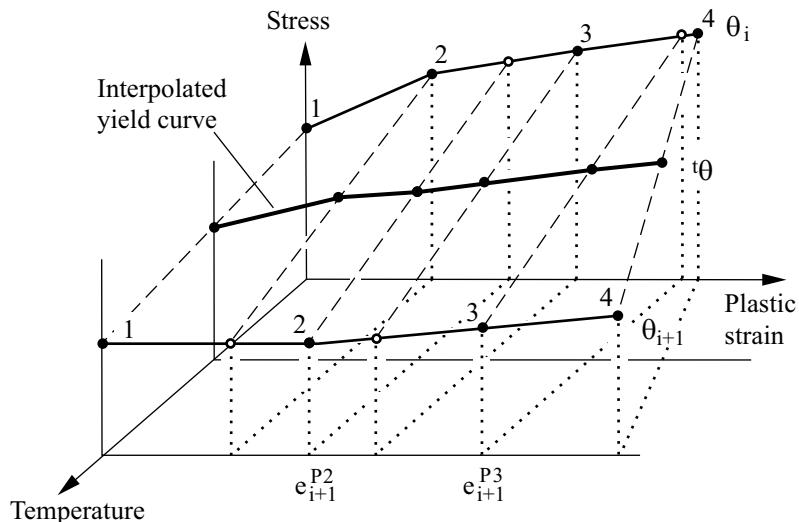


(b) Uniaxial relation of stress versus plastic strain

Figure 3.6-2: Variation of material properties with temperature for the thermo-elasto-plasticity and creep models; bilinear stress-strain curves



a) Stress-strain curves input data



b) Yield curves

Figure 3.6-3: Interpolation of multilinear yield curves with temperature

3.6.3 Evaluation of creep strains

Creep, plastic-creep, multilinear-plastic-creep material models

- The effective creep strain is calculated using one of the following creep laws:

Power creep law (creep law 1) :

$${}^t\bar{e}^C = a_0 {}^t\sigma^{a_1} t^{a_2}$$

in which a_0 , a_1 , a_2 are material constants.

Exponential creep law (creep law 2) :

$${}^t\bar{e}^C = F \left(1 - e^{-Rt} \right) + Gt$$

with

$$F = a_0 e^{a_1 \cdot {}^t\sigma}; \quad R = a_2 \left(\frac{{}^t\sigma}{a_3} \right)^{a_4}; \quad G = a_5 e^{a_6 \cdot {}^t\sigma}$$

in which a_0 to a_6 are material constants.

Eight-parameter creep law (creep law 3) :

$${}^t\bar{e}^C = S \cdot T \cdot e^{-H}$$

with

$$S = a_0 {}^t\sigma^{a_1}; \quad T = t^{a_2} + a_3 t^{a_4} + a_5 t^{a_6}, \quad H = \frac{a_7}{{}^t\theta + 273.16}$$

in which a_0 to a_7 are material constants.

In the above equations, ${}^t\bar{e}^C$, ${}^t\sigma$ and ${}^t\theta$ denote the effective creep strain, stress and temperature at time t . The unit of the temperature can either be °K or °C.

LUBBY2 creep law: The LUBBY2 creep law can be employed to predict long-term behavior of rock salt mass.

The LUBBY2 creep law is implemented for 2-D solid and 3-D solid elements. It can be used in conjunction with strain-hardening or time-hardening.

The LUBBY2 creep law is specified by the following equations:

$$\frac{d^t\bar{\epsilon}^C}{dt} = \left[\frac{1}{\bar{\eta}_K(t\bar{\sigma})} \exp\left(-\frac{\bar{G}_K(t\bar{\sigma})}{\bar{\eta}_K(t\bar{\sigma})} t\right) + \frac{1}{\bar{\eta}_M(t\bar{\sigma})} \right] t\bar{\sigma}$$

or

$$^t\bar{\epsilon}^C = \left[\frac{-1}{\bar{G}_K(t\bar{\sigma})} \exp\left(-\frac{\bar{G}_K(t\bar{\sigma})}{\bar{\eta}_K(t\bar{\sigma})} t\right) + \frac{t}{\bar{\eta}_M(t\bar{\sigma})} + \frac{1}{\bar{G}_K(t\bar{\sigma})} \right] t\bar{\sigma}$$

where $\bar{\eta}_M(t\bar{\sigma}) = \bar{\eta}_M^* e^{m t\bar{\sigma}}$, $\bar{G}_K(t\bar{\sigma}) = \bar{G}_K^* e^{k_1 t\bar{\sigma}}$ and $\bar{\eta}_K(t\bar{\sigma}) = \bar{\eta}_K^* e^{k_2 t\bar{\sigma}}$. Also, as usual, $^t\bar{\epsilon}^C$ is the effective creep strain, $\frac{d^t\bar{\epsilon}^C}{dt}$ is the effective creep strain rate and $t\bar{\sigma}$ is the effective creep stress.

In these equations, m , k_1 , k_2 , $\bar{\eta}_M^*$, \bar{G}_K^* and $\bar{\eta}_K^*$ are six material parameters. m , k_1 , k_2 account for the stress effect on parameters $\bar{\eta}_M^*$, \bar{G}_K^* , $\bar{\eta}_K^*$. The six material parameters are identified in ADINA as a0, a1, acrp2, a3, a4 and a5 respectively.

Unloading and cyclic loading effects are considered in the LUBBY2 creep law.

Analytical solutions with the LUBBY2 creep law are very sensitive to the three parameters m , k_1 , k_2 , so these parameters must be very carefully specified.

For more information about the LUBBY2 creep law, see the following reference:

ref. K.H. Lux and S. Heusermann, "Creep Tests on Rock Salt with Changing Load as a Basis for the Verification of Theoretical Material Laws", *Proc. 6th Symp. on Salt*, Toronto, Vol. 1, pp. 417-435, 1983.

Blackburn creep law:

$${}^t\bar{e}^C = C_1 \{1 - \exp(-r_1 t)\} + C_2 \{1 - \exp(-r_2 t)\} + Gt$$

with

$$C_1 = \frac{A_0 G^{A_1}}{r_1}, \quad C_2 = \frac{A_2 G^{A_3}}{r_2}$$

$$r_1 = A_4 t_R^{-A_5}, \quad r_2 = A_6 t_R^{-A_7}, \quad G = F t_R^{-A_8}$$

$$\log_{10}(t_R) = B_0 + B_1 \log_{10} {}^t\sigma + B_2 (\log_{10} {}^t\sigma)^2 + A_9$$

$$F = A_{10} \exp\left(\frac{A_{11}}{{}^t\theta + 273.16}\right)$$

$$B_0 = \frac{A_{12}}{{}^t\theta + 273.16} + A_{13}, \quad B_1 = \frac{A_{14}}{{}^t\theta + 273.16}, \quad B_2 = \frac{A_{15}}{{}^t\theta + 273.16}$$

$A_0, A_1, A_2, \dots, A_{15}$ are material constants.

Creep-variable, plastic-creep-variable, multilinear-plastic-creep-variable material models

The following creep laws can be employed:

Eight-parameter creep law (creep law 3) with variable coefficients: The same functional form as in creep law 3 described above is employed, but coefficients a_0 to a_7 are considered to be functions of temperature and effective stress.

The values of a_0 to a_7 can also be evaluated within user-supplied subroutines. In the user-supplied subroutines, these

parameters can be functions of the temperature and the effective stress.

The user-supplied subroutines are:

- UCOEF2 for 2-D solid elements (file ovl30u.f)
- UCOEF3 for 3-D solid elements (file ovl40u.f)
- UCOEFB for iso-beam elements (file ovl60u.f)
- UCOEFS for shell elements (file ovl100u.f)
- UCOEFP for pipe elements (file ovl110u.f)

These subroutines are called by ADINA every time the stress-strain law is updated.

Six-parameter LUBBY2 creep law with variable coefficients: The same functional form as in the LUBBY2 creep law described above is employed, but the six parameters are considered to be functions of the temperature (but not the stress, because the stress effects on parameters $\bar{\eta}_M$, \bar{G}_K , $\bar{\eta}_K$ have already been taken into account using the three parameters m , k_1 , k_2). The six parameters can also be evaluated using the user-supplied subroutines mentioned above.

Irradiation creep model

The irradiation creep strain is calculated from

$$\phi \bar{e}^C = [1 - \exp(-a_1\phi)] \frac{a_2 \phi \sigma}{E_0} + M \phi \sigma \phi,$$

$$M = a_3 \exp(a_4\theta + a_5)$$

where ϕ is the neutron fluence at time t , a_1, \dots, a_5 are material constants, θ is the Kelvin temperature at time t , and E_0 is the initial Young's modulus. The subsequent Young's modulus can be dependent on neutron fluence and temperature. In this case, its value is determined by using the Shifting rule described in Section 3.6.5.

All creep material models

- The creep strains are evaluated using the strain hardening procedure for load and temperature variations, and the O.R.N.L. rules for cyclic loading conditions.

ref. C.E. Pugh, J.M. Corum, K.C. Liu and W.L. Greenstreet, "Currently Recommended Constitutive Equations for Inelastic Design of FFTF Components," *Report No. TM-3602, Oak Ridge National Laboratory*, Oak Ridge, Tennessee, 1972.

Hence, the incremental creep strains are calculated using

$$e_{ij}^C = \Delta t {}^t\gamma {}^t s_{ij}$$

where the creep multiplier ${}^t\gamma$ varies with stresses and creep strains, and the ${}^t s_{ij}$ are the deviatoric stresses (see Section 3.6.2 and ref. KJB, p. 607).

The procedure used to evaluate the incremental creep strains is summarized in the following: Given the total creep strains ${}^t e_{ij}^C$ and the deviatoric stresses ${}^t s_{ij}$ at time t ,

- 1) Calculate the effective stress

$${}^t \bar{\sigma} = \left[\frac{3}{2} {}^t s_{ij} {}^t s_{ij} \right]^{\frac{1}{2}}$$

- 2) Calculate the pseudo-effective creep strain

$${}^t \bar{e}^C = \left[\frac{2}{3} \left({}^t \bar{e}_{ij}^C - e_{ij}^{orig} \right) \left({}^t \bar{e}_{ij}^C - e_{ij}^{orig} \right) \right]^{\frac{1}{2}}$$

- 3) Substitute ${}^t \bar{\sigma}$, ${}^t \bar{e}^C$ and ${}^t \theta$ into the generalized uniaxial creep law

$${}^t\bar{e}^C = F_c({}^t\bar{\sigma}, {}^t\theta, \bar{t})$$

- 4) Solve for the pseudo-time \bar{t} .
- 5) Calculate the effective creep strain rate from the generalized uniaxial creep law

$$\frac{\partial {}^t\bar{e}^C}{\partial t} = \frac{\partial F_c}{\partial \bar{t}}({}^t\bar{\sigma}, {}^t\theta, \bar{t})$$

- 6) Calculate ${}^t\gamma$

$${}^t\gamma = \frac{3}{2} \frac{\left(\frac{\partial {}^t\bar{e}^C}{\partial t} \right)}{{}^t\bar{\sigma}}$$

and the increment

$$\Delta {}^t\bar{e}^C = \Delta t {}^t\gamma {}^t s_{ij}$$

Steps 1) through 4) correspond to a strain hardening procedure. There is also the option of using a time hardening procedure, in which steps 1) through 4) are replaced by the formula $\bar{t} = t$. See ref. KJB, pp 607-608 for a discussion of the two hardening procedures. The strain hardening procedure is recommended for general use (and is the default).

- Rupture of the material due to excessive creep strain can be specified. In this case, a uniaxial creep strain failure envelope must be defined as a function of temperature and effective stress (Fig. 3.6-4).

The following criterion is then checked at each integration point during the analysis:

$${}^t\bar{e}^C \leq \frac{e_{u,r}^C}{TF}$$

in which $e_{u,r}^C$ is the uniaxial creep strain at failure, ${}^t\bar{e}^C$ is the

effective creep strain, TF is the triaxiality factor, and

$${}^t\sigma_m = {}^t\sigma_{11} + {}^t\sigma_{22} + {}^t\sigma_{33}$$
 is the volumetric stress.

You can choose to use the triaxiality factor $TF = \frac{|{}^t\sigma_m|}{{}^t\bar{\sigma}}$ (the default) or to use no triaxiality factor (that is, $TF = 1$).

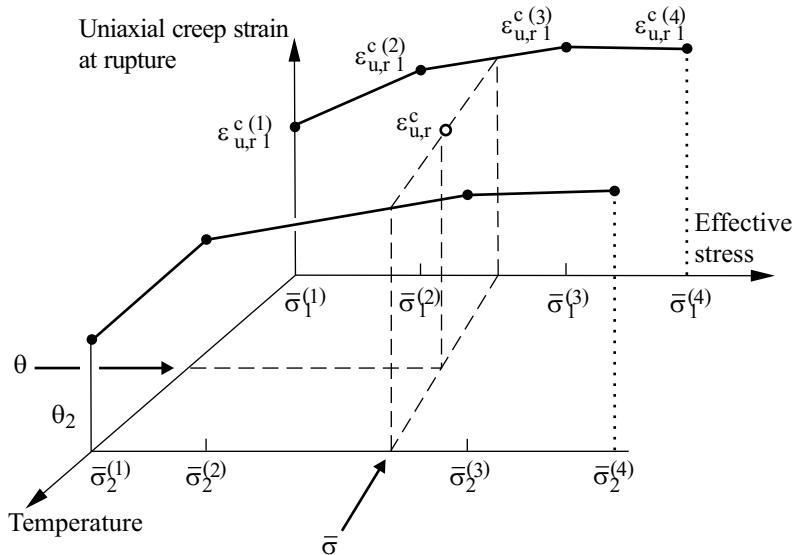


Figure 3.6-4: Multilinear interpolation of uniaxial creep strain at rupture

You can choose to apply the rupture model only under tensile conditions (${}^t\sigma_m > 0$) or under both tensile and compressive conditions. The default is to apply the rupture model only under tensile conditions.

Rupture can be allowed only in tension, based on the sign of ${}^t\sigma_m$.

- You can apply a time offset to any of the creep material models (for example, creep law 1 becomes ${}^t\bar{e}^C = a_0 {}^t\sigma^{a_1} (t - t_0)^{a_2}$, where t_0 is the time offset, which can be positive or negative). The time offset can be modified in a restart run.

- The accumulated effective creep strain is calculated as

$${}^t\bar{\epsilon}^C = \sum_{\text{all solution steps}} \Delta\bar{\epsilon}^C$$

where $\Delta\bar{\epsilon}^C = \sqrt{\frac{2}{3}\Delta\mathbf{e}^C \cdot \Delta\mathbf{e}^C}$ and $\Delta\mathbf{e}^C$ is the tensor of creep strain increments in a solution step. Because of the summation over the solution steps, we refer to the calculated value of ${}^t\bar{\epsilon}^C$ as the accumulated effective creep strain.

3.6.4 Computational procedures

*ref. KJB
Section 6.6.3*

- The stresses at the integration points are evaluated using the effective-stress-function algorithm.

ref. M. Kojić and K.J. Bathe, "The Effective-Stress-Function Algorithm for Thermo-Elasto-Plasticity and Creep," *Int. J. Numer. Meth. Engng.*, Vol. 24, No. 8, pp. 1509-1532, 1987.

Briefly, the procedure used consists of the following calculations. The general constitutive equation

$${}^{t+\Delta t}\sigma^{(i)} = {}^{t+\Delta t}\mathbf{C}^E \left({}^{t+\Delta t}\mathbf{e}^{(i)} - {}^{t+\Delta t}\mathbf{e}^P - {}^{t+\Delta t}\mathbf{e}^C - {}^{t+\Delta t}\mathbf{e}^{TH} \right) \quad (3.6-1)$$

is solved separately for the mean stress and for the deviatoric stresses. In this equation the index (i) denotes the iteration counter in the iteration for nodal point equilibrium. For easier writing this index will be dropped in the discussion to follow. The mean stress is calculated as

$${}^{t+\Delta t}\sigma_m = \frac{{}^{t+\Delta t}E}{1 - 2{}^{t+\Delta t}\nu} \left({}^{t+\Delta t}e_m - {}^{t+\Delta t}e^{TH} \right) \quad (3.6-2)$$

The deviatoric stresses ${}^{t+\Delta t}\mathbf{s}$ depend on the inelastic strains and they can be expressed as

$${}^{t+\Delta t} \mathbf{s} = \frac{1}{{}^{t+\Delta t} a_E + \alpha \Delta t {}^{\tau} \gamma + \Delta \lambda} \left[{}^{t+\Delta t} \mathbf{e}'' - (1-\alpha) \Delta t {}^{\tau} \gamma {}^t \mathbf{s} \right] \quad (3.6-3)$$

where ${}^{t+\Delta t} a_E = \frac{{}^{t+\Delta t} E}{1 + {}^{t+\Delta t} v}$, ${}^t \mathbf{s}$ = deviatoric stress at the start of the time step and α is the integration parameter used for stress evaluation ($0 \leq \alpha < 1$) The creep and plastic multipliers ${}^{\tau} \gamma$ and $\Delta \lambda$ are functions of the effective stress ${}^{t+\Delta t} \bar{\sigma}$ only, and they account for creep and plasticity; also

$${}^{t+\Delta t} \mathbf{e}'' = {}^{t+\Delta t} \mathbf{e}' - {}^t \mathbf{e}^P - {}^t \mathbf{e}^C$$

is known since the deviatoric strains ${}^{t+\Delta t} \mathbf{e}'$, plastic strains ${}^t \mathbf{e}^P$ and creep strains ${}^t \mathbf{e}^C$ are known from the current displacements and the stress/strain state at the start of the current time step.

The following scalar function $f({}^{t+\Delta t} \bar{\sigma})$ is obtained from Eq. (3.6-3)

$$f({}^{t+\Delta t} \bar{\sigma}) = a^2 {}^{t+\Delta t} \bar{\sigma}^2 + b {}^{\tau} \gamma - c^2 {}^{\tau} \gamma^2 - d^2 = 0 \quad (3.6-4)$$

The zero of equation (3.6-4) provides the solution for the effective stress ${}^{t+\Delta t} \bar{\sigma}$, where

$$\begin{aligned} a &= {}^{t+\Delta t} a_E + \alpha \Delta t {}^{\tau} \gamma + \Delta \lambda \\ b &= 3(1-\alpha) \Delta t {}^{t+\Delta t} e'_{ij} {}^t s_{ij} \\ c &= (1-\alpha) \Delta t {}^t \bar{\sigma} \\ d^2 &= \frac{3}{2} {}^{t+\Delta t} e''_{ij} {}^{t+\Delta t} e''_{ij} \end{aligned}$$

with summation on the indices i, j .

Once the solution for $\bar{\sigma}$ has been determined from Eq. (3.6-4), simultaneously with the scalars ${}^{\tau} \gamma$ and $\Delta \lambda$ from the creep and

plasticity conditions, the deviatoric stress ${}^{t+\Delta t}\mathbf{s}$ is calculated from Eq. (3.6-3), and the plastic and creep strains at the end of the time step are obtained as

$$\begin{aligned} {}^{t+\Delta t}\mathbf{e}^P &= {}^t\mathbf{e}^P + \Delta\lambda {}^{t+\Delta t}\mathbf{s} \\ {}^{t+\Delta t}\mathbf{e}^C &= {}^t\mathbf{e}^C + \left[(1-\alpha) {}^t\mathbf{s} + \alpha {}^{t+\Delta t}\mathbf{s} \right] \Delta t \tau \gamma \end{aligned}$$

The above equations correspond to isotropic hardening conditions and a general 3-D analysis. The solution details for kinematic hardening conditions and for special problems (for the plane stress, shell, isoparametric beam elements) are given in the above cited reference and also in the following reference.

ref. M. Kojić and K.J. Bathe, "Thermo-Elastic-Plastic and Creep Analysis of Shell Structures", *Computers & Structures*, Vol. 26, No 1/2, pp. 135-143, 1987.

3.6.5 Shifting rule for the irradiation creep model

- Irradiation tests obtaining irradiation strain data and material property changes of graphite have been carried out generally at a constant temperature. They cannot be determined directly from the test data due to variation of temperature.

Therefore, the irradiation strain at temperature θ and under fast neutron fluence ϕ is calculated as follows

$$\varepsilon^W(\theta, \phi) = \sum_{i=1}^n \Delta \varepsilon^W(\theta_i, \phi_{i-1}, \phi_i - \phi_{i-1})$$

where

$$\Delta \varepsilon^W(\theta_i, \phi, \Delta \phi) = \varepsilon^W(\theta_i, \phi + \Delta \phi) - \varepsilon^W(\theta_i, \phi)$$

This is illustrated in Figure 3.6-1, and documented in the following reference.

ref. T. Iyoku and M. Ishihara, "Development of Thermal/Irradiation Stress Analytical Code VIENUS for HTTR Graphite Block", *Journal of Nuclear Science and*

Technology, 28[10], pp. 921-931, October 1991.

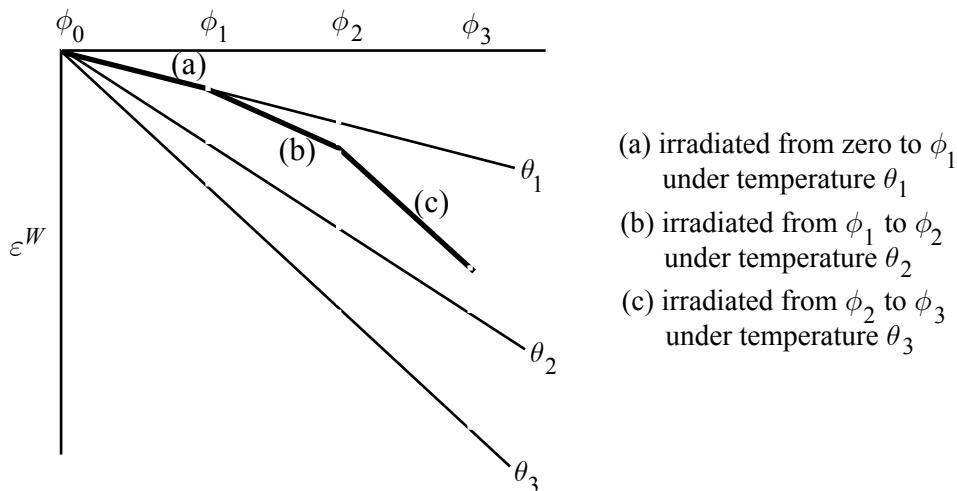


Figure 3.6-1: Shifting rule for irradiation creep model

3.7 Concrete material model

3.7.1 General considerations

- The concrete model can be employed with the **2-D solid** and **3-D solid** elements.
- The concrete model can be used with the **small displacement** and **large displacement** formulations. In all cases, small strains are assumed.
 When used with the small displacement formulation, a materially-nonlinear-only formulation is employed, and when used with the large displacement formulation, a TL formulation is employed.
- Although the model is entitled "concrete model", the basic constitutive characteristics are such that the model can also be useful when representing other materials. The basic material characteristics are:

- ▶ Tensile failure at a maximum, relatively small principal tensile stress
- ▶ Compression crushing failure at high compression
- ▶ Strain softening from compression crushing failure to an ultimate strain, at which the material totally fails.

The tensile and compression crushing failures are governed by tensile failure and compression crushing failure envelopes.

These material characteristics pertain, for example, to a variety of rocks.

- It is well accepted that concrete is a very complex material. The model provided in ADINA may not contain all the detailed material characteristics that you may look for. However, considering the variability of concrete materials that need to be described in practice, and recognizing that the model may also be useful in the modeling of rock materials, the objective is to provide an effective model with sufficient flexibility to model most of the commonly used material behaviors.

3.7.2 Formulation of the concrete material model in ADINA

- The notation used in the formulation of the model is given below.

$'E$ = equivalent multiaxial tangent Young's modulus at time t (the left superscript " t " refers to time t)

\tilde{E}_0 = uniaxial initial tangent modulus (all uniaxial qualities are identified with a curl " \sim " placed over them)

\tilde{E}_s = uniaxial secant modulus corresponding to uniaxial maximum stress, $\tilde{E}_s = \frac{\tilde{\sigma}_c}{\tilde{\epsilon}_c}$

\tilde{E}_u = uniaxial secant modulus corresponding to uniaxial ultimate stress, $\tilde{E}_u = \frac{\tilde{\sigma}_u}{\tilde{\epsilon}_u}$

${}^t\tilde{E}_{pi}$	= uniaxial tangent modulus in the direction of ${}^t\sigma_{pi}$
${}^t\epsilon_{ij}$	= total strains
ϵ_{ij}	= incremental strains
${}^t\tilde{\epsilon}$	= uniaxial strain
$\tilde{\epsilon}_c$	= uniaxial strain corresponding to $\tilde{\sigma}_c$ ($\tilde{\epsilon}_c < 0$)
$\tilde{\epsilon}_u$	= ultimate uniaxial compressive strain ($\tilde{\epsilon}_u < 0$)
${}^t\sigma_{ij}$	= total stresses
σ_{ij}	= incremental stresses
${}^t\tilde{\sigma}$	= uniaxial stress
$\tilde{\sigma}_t$	= uniaxial cut-off tensile strength ($\tilde{\sigma}_t > 0$)
$\tilde{\sigma}_{tp}$	= post-cracking uniaxial cut-off tensile strength $(\tilde{\sigma}_{tp} > 0)$. Note that if $\tilde{\sigma}_{tp} = 0$, ADINA sets $\tilde{\sigma}_{tp} = \tilde{\sigma}_t$.
$\tilde{\sigma}_c$	= maximum uniaxial compressive stress ($\tilde{\sigma}_c < 0$)
$\tilde{\sigma}_u$	= ultimate uniaxial compressive stress ($\tilde{\sigma}_u < 0$)
${}^t\sigma_{pi}$	= principal stress in direction i (${}^t\sigma_{p1} \geq {}^t\sigma_{p2} \geq {}^t\sigma_{p3}$)

- Three basic features are used in the concrete model:
 - ▶ A nonlinear stress-strain relation to allow for the weakening of the material under increasing compressive stresses
 - ▶ Failure envelopes that define failure in tension and crushing in compression
 - ▶ A strategy to model the post-cracking and crushing behavior of the material

Stress-strain relations: The general multiaxial stress-strain relations are derived from a uniaxial stress-strain relation ${}^t\tilde{\sigma}$ versus ${}^t\tilde{\epsilon}$.

A typical uniaxial stress ${}^t\tilde{\sigma}$ to uniaxial strain ${}^t\tilde{e}$ relation is shown in Fig. 3.7-1. This stress-strain relation shows that there are three strain phases; namely, ${}^t\tilde{e} \geq 0$, $0 > {}^t\tilde{e} \geq \tilde{e}_c$, and $\tilde{e}_c > {}^t\tilde{e} \geq \tilde{e}_u$ where \tilde{e}_c is the strain corresponding to the minimum (crushing) stress $\tilde{\sigma}_c$ that can be reached, and \tilde{e}_u is the ultimate compressive strain.

If $\tilde{E}_0 > 0$, the material is in tension and the stress-strain relation is linear until tensile failure at the stress $\tilde{\sigma}_t$. A constant Young's modulus \tilde{E}_0 is employed, i.e.,

$${}^t\tilde{\sigma} = \tilde{E}_0 {}^t\tilde{e} \quad (3.7-1)$$

For ${}^t\tilde{e} \leq 0$, we assume the following relation

$$\frac{{}^t\tilde{\sigma}}{\tilde{\sigma}_c} = \frac{\left(\frac{\tilde{E}_0}{\tilde{E}_s}\right)\left(\frac{{}^t\tilde{e}}{\tilde{e}_c}\right)}{1 + A\left(\frac{{}^t\tilde{e}}{\tilde{e}_c}\right) + B\left(\frac{{}^t\tilde{e}}{\tilde{e}_c}\right)^2 + C\left(\frac{{}^t\tilde{e}}{\tilde{e}_c}\right)^3}$$

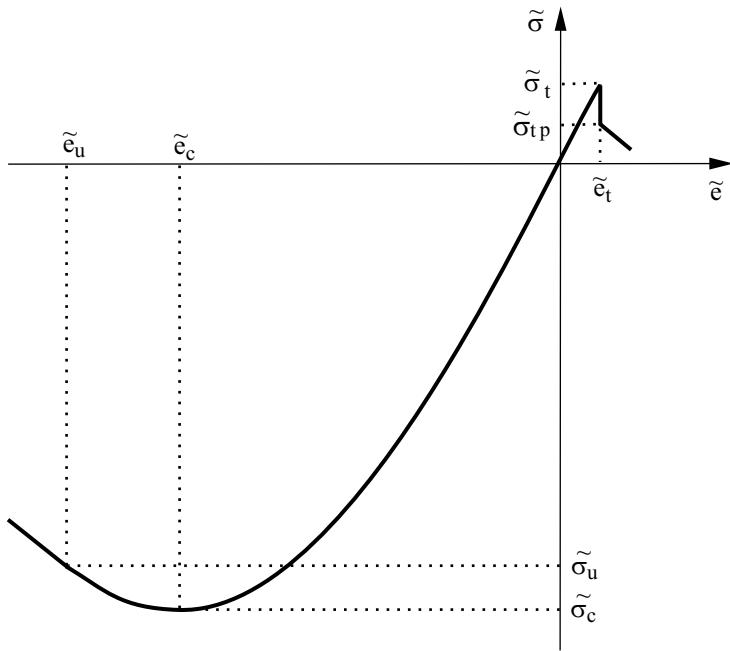


Figure 3.7-1: Uniaxial stress-strain relation used in the concrete model

and hence

$${}^t\tilde{E} = \frac{\tilde{E}_0 \left[1 - B \left(\frac{{}^t\tilde{e}}{\tilde{e}_c} \right)^2 - 2C \left(\frac{{}^t\tilde{e}}{\tilde{e}_c} \right)^3 \right]}{\left[1 + A \left(\frac{{}^t\tilde{e}}{\tilde{e}_c} \right) + B \left(\frac{{}^t\tilde{e}}{\tilde{e}_c} \right)^2 + C \left(\frac{{}^t\tilde{e}}{\tilde{e}_c} \right)^3 \right]^2} \quad (3.7-2)$$

where

$$A = \frac{\left[\frac{\tilde{E}_0}{\tilde{E}_u} + (p^3 - 2p^2) \frac{\tilde{E}_0}{\tilde{E}_s} - (2p^3 - 3p^2 + 1) \right]}{\left[(p^2 - 2p + 1)p \right]}$$

$$B = \left[\left(2 \frac{\tilde{E}_0}{\tilde{E}_s} - 3 \right) - 2A \right]$$

$$C = \left[\left(2 - \frac{\tilde{E}_0}{\tilde{E}_s} \right) + A \right]$$

and the parameters \tilde{E}_0 , $\tilde{\sigma}_c$, \tilde{e}_c , $\tilde{E}_s = \frac{\tilde{\sigma}_c}{\tilde{e}_c}$, $\tilde{\sigma}_u$, \tilde{e}_u , $p = \frac{\tilde{e}_u}{\tilde{e}_c}$, $\tilde{E}_u = \frac{\tilde{\sigma}_u}{\tilde{e}_u}$

are obtained from uniaxial tests.

The stress-strain relation in Eq. (3.7-2) assumes monotonic loading conditions. For unloading conditions and loading back to the stress state from which unloading occurred, the initial Young's modulus \tilde{E}_0 is used. For strain states beyond \tilde{e}_u in compression, it is assumed that stresses are linearly released to zero, using the following modulus:

$$\tilde{E}_u = \frac{\tilde{\sigma}_u - \tilde{\sigma}_c}{\tilde{e}_u - \tilde{e}_c}$$

Note that confined concrete can therefore be modeled using close values for $\tilde{\sigma}_u$ and $\tilde{\sigma}_c$.

Under multiaxial stress conditions, the stress-strain relations are evaluated differently depending on whether the material is loading or unloading. Poisson's ratio is assumed to be constant under tensile stress conditions and can vary in the compressive region.

To characterize loading and unloading conditions we define a loading scalar ${}^t g$ for each integration point,

$${}^t g = {}^t \sigma_e \quad (3.7-3)$$

where ${}^t \sigma_e$ is the effective stress at time t . The material is loading at the integration point except when the unloading conditions are determined,

$$^t g < g_{max} \quad (3.7-4)$$

where g_{max} is the maximum value of the loading scalar that has been reached during the complete solution.

During unloading, the material is assumed to be isotropic and the initial Young's modulus, \tilde{E}_0 , is used to form the incremental stress-strain matrix, both for stiffness and stress calculations.

In loading conditions, the principal stresses ${}^t \sigma_{pi}$ (with ${}^t \sigma_{p1} \geq {}^t \sigma_{p2} \geq {}^t \sigma_{p3}$) are calculated. For each principal stress direction, a tangent Young's modulus ${}^t \tilde{E}_{pi}$ corresponding to the given strain state ${}^t e_{pi}$ is evaluated using Eqs. (3.7-1) and (3.7-2).

In Eq. (3.7-2), the strains ${}^t e_{pi}$ are used together with the parameters $\tilde{\sigma}'_c$, $\tilde{\sigma}'_u$, \tilde{e}'_c and \tilde{e}'_u defined in Eq. (3.7-8) which accounts for the multiaxial stress conditions.

The material is considered as orthotropic with the directions of orthotropy being defined by the principal stress directions. Once cracking occurs in any direction i , that direction is fixed from that point onward in calculating ${}^t \sigma_{pi}$.

The stress-strain matrix corresponding to these directions is, for three-dimensional stress conditions,

$$\mathbf{C} = \frac{1}{(1+v)(1+2v)} \times \begin{bmatrix} (1-v) {}^t \tilde{E}_{p1} & v {}^t E_{12} & v {}^t E_{13} & 0 & 0 & 0 \\ (1-v) {}^t \tilde{E}_{p2} & v {}^t E_{23} & 0 & 0 & 0 & 0 \\ (1-v) {}^t \tilde{E}_{p3} & 0 & 0 & 0 & 0 & 0 \\ & 0.5(1-2v) {}^t E_{12} & 0 & 0 & 0 & 0 \\ \text{symmetric} & & 0.5(1-2v) {}^t E_{13} & 0 & 0 & 0 \\ & & & 0.5(1-2v) {}^t E_{23} & 0 & 0 \end{bmatrix} \quad (3.7-5)$$

where ν is the constant Poisson's ratio and the ${}^t E_{ij}$ with $i \neq j$ are evaluated using

$${}^t E_{ij} = \frac{\left| {}^t \sigma_{pi} \right|^t \tilde{E}_{pi} + \left| {}^t \sigma_{pj} \right|^t \tilde{E}_{pj}}{\left| {}^t \sigma_{pi} \right| + \left| {}^t \sigma_{pj} \right|} \quad (3.7-6)$$

Note that the above stress-strain relations for material loading conditions are only employed in the calculation of the stiffness matrix at time t . Considering the evaluation of the stress increment from t to time $t + \Delta t$, we have

$$\boldsymbol{\sigma} = \hat{\mathbf{C}} \mathbf{e} \quad (3.7-7)$$

where the stress-strain matrix $\hat{\mathbf{C}}$ is the one defined in Eq. (3.7-5) but using the Young's moduli ${}^t \tilde{E}_{pi}$. These moduli are evaluated using three point Gaussian integration of \tilde{E}_{pi} between ${}^t e_{pi}$ and ${}^{t+\Delta t} e_{pi}$, with ${}^t e_{pi}$ and ${}^{t+\Delta t} e_{pi}$ the strain components measured in the directions of the respective principal stresses ${}^t \sigma_{pi}$ and ${}^{t+\Delta t} \sigma_{pi}$.

Material failure envelopes: The failure envelopes shown in Figs. 3.7-2 to 3.7-5 are employed to establish the uniaxial stress-strain law accounting for multiaxial stress conditions, and to identify whether tensile or crushing failure of the material has occurred.

Uniaxial stress law under multiaxial stress conditions: Having established the principal stresses ${}^t \sigma_{pi}$ with ${}^t \sigma_{p1} \geq {}^t \sigma_{p2} \geq {}^t \sigma_{p3}$ the stresses ${}^t \sigma_{p1}$ and ${}^t \sigma_{p2}$ are held constant and the minimum stress that would have to be reached in the third principal direction to cause crushing of the material is calculated using the failure envelope, see Fig. 3.7-5(a). Let this stress be $\tilde{\sigma}'_c$, we have

$$\gamma_1 = \frac{\tilde{\sigma}'_c}{\tilde{\sigma}_c}$$

and

$$\begin{aligned}\tilde{\sigma}'_u &= \gamma_1 \tilde{\sigma}_u; \quad \vec{e}'_c = (C_1 \gamma_1^2 + C_2 \gamma_1) \vec{e}_c; \\ \vec{e}'_u &= (C_1 \gamma_1^2 + C_2 \gamma_1) \vec{e}_u\end{aligned}\quad (3.7-8)$$

where C_1 and C_2 are input parameters. Normally, $C_1 = 1.4$ and $C_2 = -0.4$. The constants $\tilde{\sigma}'_c$, $\tilde{\sigma}'_u$, \vec{e}'_c and \vec{e}'_u are employed instead of the unprimed variables in order to establish, using Eq. (3.7-2), the uniaxial stress-strain law under multiaxial conditions (see Fig. 3.7-5(b)).

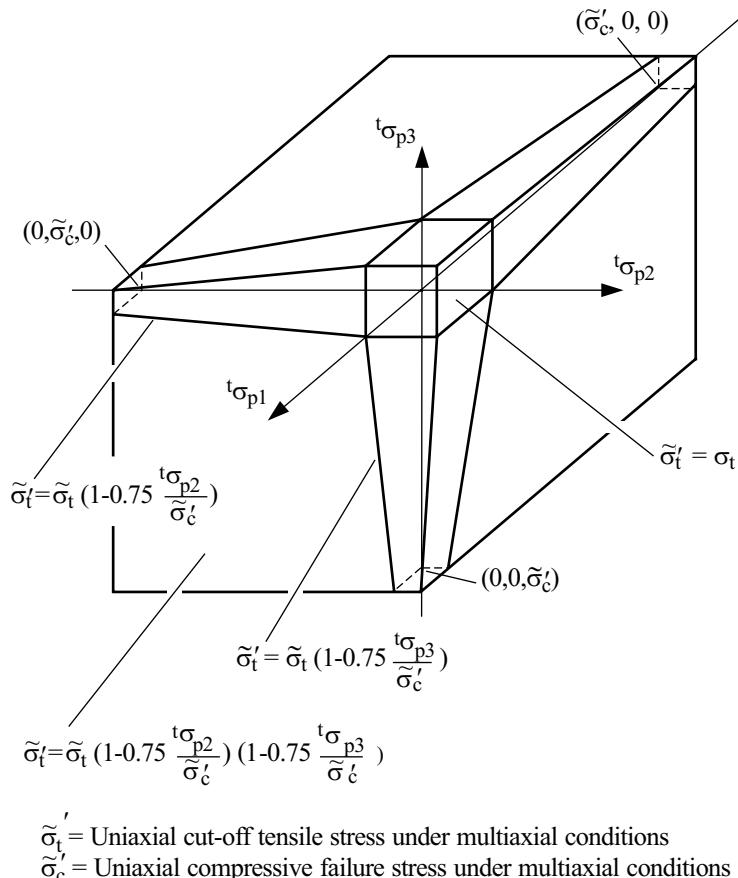


Figure 3.7-2: Three-dimensional tensile failure envelope of concrete model

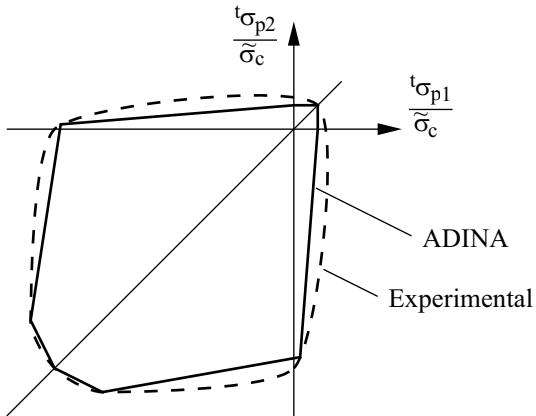


Figure 3.7-3: Biaxial concrete compressive failure envelope

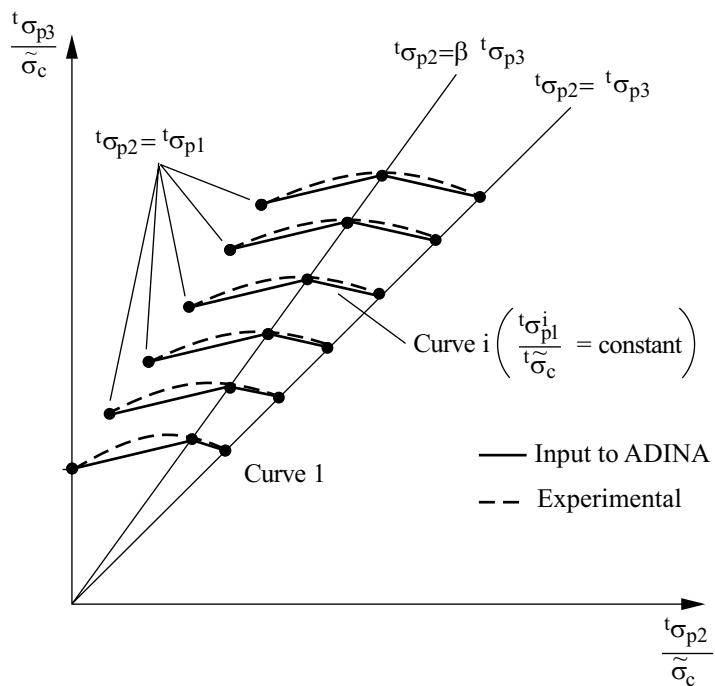
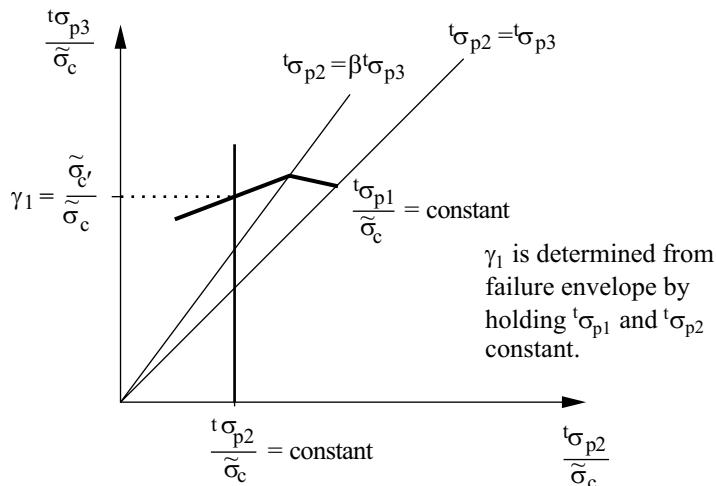
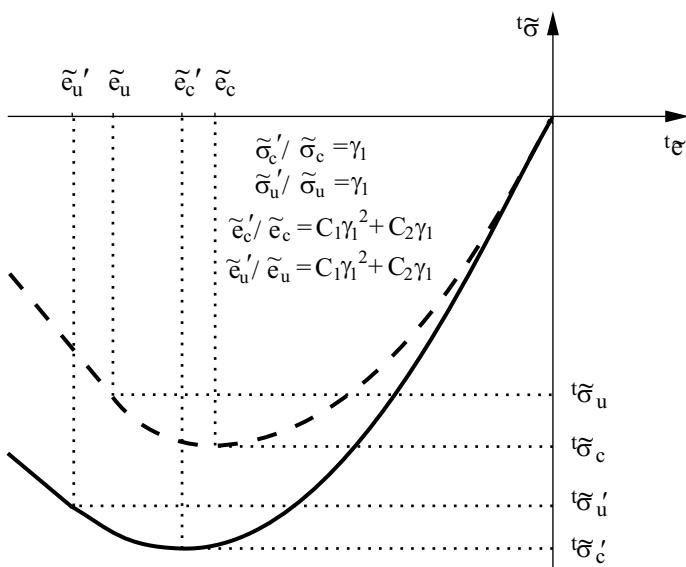


Figure 3.7-4: Triaxial compressive failure envelope

(a) Determination of $\tilde{\sigma}'_c$ from given $(t\sigma_{p1}, t\sigma_{p2})$ 

(b) Determine uniaxial behavior for multiaxial stress conditions

Figure 3.7-5: Definitions used for evaluation of one-dimensional stress-strain law under multiaxial stress conditions

Tensile failure envelope: The tensile failure envelope used in the concrete model is shown in Fig. 3.7-2. To identify whether the material has failed, the principal stresses are used to locate the current stress state. Note that the tensile strength of the material in a principal direction does not depend on tensile stresses in the other principal stress directions, but depends on compressive stresses in the other directions.

Tensile failure occurs if the tensile stress in a principal stress direction exceeds the tensile failure stress. In this case it is assumed that a plane of failure develops perpendicular to the corresponding principal stress direction. The effect of this material failure is that the normal and shear stiffnesses and stresses across the plane of failure are reduced and plane stress conditions are assumed to exist at the plane of tensile failure, as discussed in more detail below.

Prior to tensile failure the stress-strain material law is given by Eqs. (3.7-5) to (3.7-7). Assuming that $'\sigma_{pl}$ is larger than the tensile failure stress, the new material stress-strain matrix used in the stiffness matrix calculation is given by

$$\mathbf{C} = \begin{bmatrix} \tilde{E}_0 \eta_n & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{(1-v^2)} \begin{bmatrix} {}' \tilde{E}_{p2} & v {}' E_{23} \\ {}' \tilde{E}_{p3} \end{bmatrix} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{\tilde{E}_0 \eta_s}{2(1+v)} & 0 & 0 & 0 & 0 & 0 \\ \text{symmetric} & \frac{\tilde{E}_0 \eta_s}{2(1+v)} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{{}' E_{23}}{2(1+v)} & 0 & 0 \end{bmatrix} \quad (3.7-9)$$

where the ${}^t\tilde{E}_{pi}$ are the uniaxial Young's moduli evaluated in the principal stress directions using Eqs. (3.7-1) or (3.7-2), and the ${}^tE_{ij}$ are evaluated using Eq. (3.7-6).

The constants η_n and η_s are the stiffness and shear reduction factors, respectively. Typically, $\eta_n = 0.0001$ and $\eta_s = 0.5$. The factor η_n is not exactly equal to zero in order to avoid the possibility of a singular stiffness matrix. The factor η_s depends on a number of physical factors and you must use judgement to choose its value. For the concrete model in ADINA, η_n and η_s are both input parameters.

For stress calculation, the following stress-strain matrices are used:

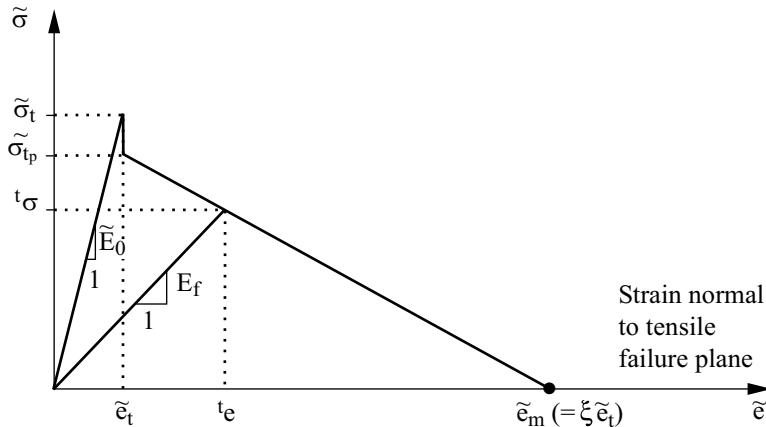
- ▶ For the tensile stress normal to the tensile failure plane and the shear stresses in this plane, we use the total strains to calculate the total stresses with

$$\hat{\mathbf{C}} = \begin{bmatrix} E_f & 0 & 0 \\ & G_{12}^f & 0 \\ \text{sym.} & & G_{13}^f \end{bmatrix} \quad (3.7-10a)$$

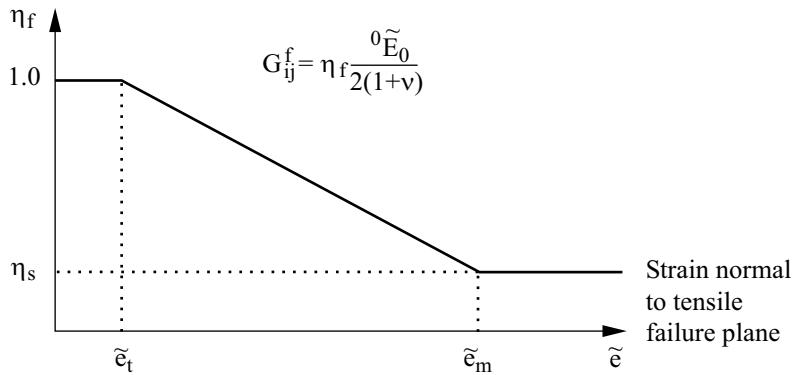
where E_f and G_{12}^f, G_{13}^f are evaluated as shown in Fig. 3.7-6. In this figure, ξ is a user input variable that defines the amount of tension stiffening. Fig. 3.7-6(b) shows that G_{ij}^f is evaluated from the initial shear modulus. Also, Fig. 3.7-6(a) assumes loading from zero stress directly into the tensile region. If the tensile stress is reached by unloading from a compressive stress, the strain normal to the tensile failure plane is measured from the specific strain value at which the stress is zero (see Fig. 3.7-7 – point 11).

Special care in the analysis must be taken if ξ is chosen to be greater than 1.0, because strain softening may yield non-unique solutions.

To obtain a mesh independent solution, the fracture energy G_f can be provided instead of ξ . In doing so, ξ is evaluated at each integration point, based on the size of the finite elements (see Fig. 3.7-6(c)). In this case, ADINA also calculates η_s internally, overwriting the user input.



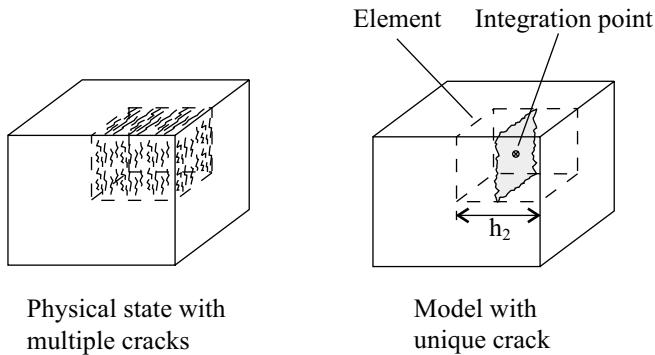
- a) Calculation of Young's modulus E_f normal to tensile failure plane.
 $\tilde{\sigma}_{tp}$ is the post-cracking uniaxial cut-off tensile stress.



- b) Calculation of shear modulus in tensile failure plane

Figure 3.7-6: Material moduli for stress calculation after tensile failure

- For the remaining stress components, the increments are evaluated from the strain increments using



$$\xi = \frac{2 \tilde{E}_0 G_f}{\tilde{\sigma}_t^2 h_2} \text{ with } G_f \text{ energy released per unit area in direction perpendicular to } h_2 \text{ direction}$$

c) Smeared crack approach for the calculation of the parameter ξ

Figure 3.7-6: (continued)

$$\mathbf{C} = \frac{1}{(1-v^2)} \begin{bmatrix} {}^T \tilde{E}_{p2} & v {}^T \tilde{E}_{23} & 0 \\ {}^T \tilde{E}_{p3} & 0 & \\ \text{sym.} & (1-v^2) {}^T G_{23} & \end{bmatrix} \quad (3.7-10b)$$

where the ${}^T \tilde{E}_{pi}$ are the uniaxial Young's moduli used in Eq. (3.7-7) and ${}^T E_{23}$ and ${}^T G_{23}$ are evaluated using Eq. (3.7-6) but with ${}^T \tilde{E}_{pi}$ instead of ${}^T \tilde{E}_{pi}$.

Compressive failure envelope: The triaxial failure envelope used in the concrete model is shown in Fig. 3.7-4. Note that the biaxial failure envelope shown in Fig. 3.7-3 is curve 1 of the triaxial failure envelope. Also, note that the biaxial and triaxial envelope curves can be used to represent a large number of specific envelopes by the input explained below, and this flexibility makes it possible to model various concrete and rock materials.

The triaxial compression failure envelope is shown in Fig. 3.7-4. First, the values $\frac{\sigma_{p1}^i}{\tilde{\sigma}_c}$ are input. These values define for which levels of the $'\sigma_{p1}$ the two-dimensional failure envelopes functions of stresses $'\sigma_{p2}$ and $'\sigma_{p3}$ are input.

Six failure envelopes must be defined, each with three points corresponding to the locations $'\sigma_{p2} = '\sigma_{p1}$, $'\sigma_{p2} = \beta '\sigma_{p3}$, and $'\sigma_{p2} = '\sigma_{p3}$ where β is an input parameter.

To identify compression failure, the largest principal stress, $'\sigma_{p1}$, is employed to establish the biaxial failure envelope function of $'\sigma_{p2}$ and $'\sigma_{p3}$, using the interpolation shown in Fig. 3.7-5(a). If the stress state corresponding to $'\sigma_{p2}$ and $'\sigma_{p3}$ lies on or outside this biaxial failure envelope, then the material has crushed.

It should be noted that you must choose appropriate values for the input of the failure surfaces and other parameters. These may vary significantly for different materials and structures and must be obtained from experimental data.

Material behavior after failure: The post failure material behaviors considered in the concrete model in ADINA include the post tensile cracking, post compression crushing, and strain-softening behaviors.

Post tensile cracking behavior: Once a tensile plane of failure has formed, it is checked in each subsequent solution step to see whether the failure is still active. The failure is considered to be inactive provided the normal strain across the plane becomes negative and less than the strain at which the "last" failure occurred. It is otherwise active. Therefore, a tensile failure plane can repeatedly be active and inactive. Consider the uniaxial cyclic loading case shown in Fig. 3.7-7 in which the strain is prescribed. As the strain reaches the tensile cut-off limit, a tensile failure plane becomes active. This tensile failure plane remains active while the strain continues to increase to point 2 and then decreases to point 3 and then increases to point 4 and then decreases to point 5. The failure plane becomes inactive at point 5 and remains so while the

strain decreases to point 8 and increases again to point 11. At point 11, the tensile failure plane again becomes active, and it remains active until the strain reaches point 13 and beyond.

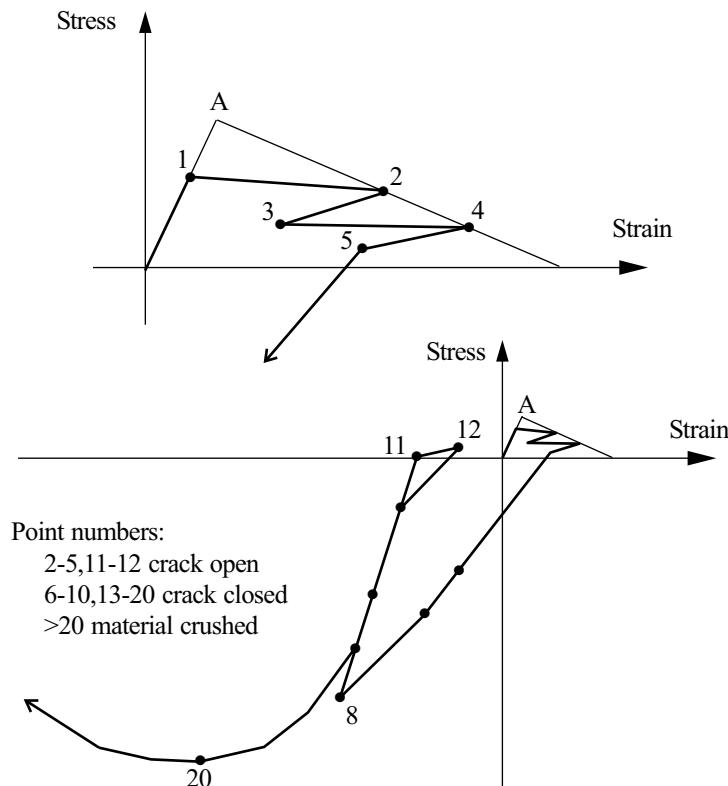


Figure 3.7-7: Uniaxial cyclic loading stress-strain experiment with the concrete model

If a tensile failure plane has developed, which may or may not be active, the material stress-strain relations are always established, as described above, corresponding to the directions along and perpendicular to the plane of failure. Hence, instead of using the principal stresses and corresponding directions as done for the unfailed material, the stresses in the directions defined by the material tensile failure plane are used to evaluate the stress-strain matrix, corresponding to those directions. Once a failure plane has been initiated, a subsequent failure plane is assumed to form

perpendicular to the direction of the first failure plane whenever a normal stress along the original failure plane has reached the tensile failure stress. It follows that at any integration point, the direction of the third tensile failure plane is fixed once failure has occurred in two directions.

Postcompression crushing behavior: If the material has crushed in compression, it is assumed that the material strain-softens in all directions.

Consider first uniaxial stress conditions. As shown in Fig. 3.7-1, when the uniaxial strain is smaller than \tilde{e}_c , the material has crushed and softens with increasing compressive strain, i.e., $'\tilde{E}$ is negative.

Under multiaxial stress conditions the compression crushing is identified using the multiaxial failure envelope, and once the material has crushed, isotropic conditions are assumed. As in uniaxial conditions, in the subsequent solution steps the Young's modulus is assumed to be very small but positive in the stiffness matrix calculations, but the stress increments are computed from the uniaxial stress-strain law with the constants $\tilde{\sigma}'_c$, \tilde{e}'_c and so on, see Eq. (3.7-8), corresponding to the multiaxial conditions at crushing. The Young's modulus $'E$ corresponding to the current strain increment e_{p3} is evaluated using the uniaxial stress-strain relationship in Fig. 3.7-1

$${}^{\tau}E = \frac{\left[\tilde{\sigma}|_{e_{p3}+e_{p3}} - \tilde{\sigma}|_{e_{p3}} \right]}{e_{p3}}$$

where $'e_{p3}$ and e_{p3} are the strain component at time t measured in the direction of the principal stress $'\sigma_{p3}$. To obtain the stress increment, Eq. (3.7-7) is used where the matrix $\hat{\mathbf{C}}$ corresponds to isotropic material conditions with Young's modulus $'E$. Note that when $'e_{p3}$ becomes equal to or less than \tilde{e}'_u , the stresses are linearly released to zero.

If unloading of the crushed material in the strain-softening region occurs, characterized by $e_{p3} \geq 0$, the initial Young's modulus \tilde{E}_0 is used. At any time during post-crushing calculations, if any one of the principal stresses checked individually reaches a positive value, this stress is set to zero.

Temperature effects: In some analyses, temperature strains ${}^t\mathbf{e}^{TH}$ need to be included. These strains are taken into account by replacing the total incremental strain \mathbf{e} in the governing incremental equations with the strain $\mathbf{e} - \mathbf{e}^{TH}$, where \mathbf{e}^{TH} is the thermal incremental strain and is calculated from the temperature conditions.

The following material properties can be defined as temperature-dependent:

- ▶ Uniaxial initial tangent modulus \tilde{E}_0
- ▶ Poisson's ratio ν
- ▶ Coefficient of thermal expansion α
- ▶ Uniaxial cut-off tensile stress $\tilde{\sigma}_t$
- ▶ Post-cracking uniaxial cut-off tensile stress $\tilde{\sigma}_{tp}$
- ▶ Uniaxial maximum compressive stress $\tilde{\sigma}_c$
- ▶ Uniaxial compressive strain at $\tilde{\sigma}_c$, \tilde{e}_c
- ▶ Ultimate uniaxial compressive stress $\tilde{\sigma}_u$
- ▶ Uniaxial strain at $\tilde{\sigma}_u$, \tilde{e}_u
- ▶ Fracture energy g_f
- ▶ Constant for tensile strain failure xsi_i

The nodal point temperatures are input to ADINA as discussed in Section 5.6.

Poisson's ratio in the compressive region: It has been observed in experiments that the ratio of lateral strain to principal compressive strain remains constant until approximately 80% of the maximum compressive stress $\tilde{\sigma}_c$.

Usually, we assume that the Poisson ratio is constant throughout the analysis; however, as an option, the value ν_s can be used when

the material dilates under compression. The value of v_s is given by

$$\begin{cases} v_s = v_s = v & \text{when } \gamma_2 = \frac{{}^t\sigma_{p3}}{\tilde{\sigma}'_c} \leq \gamma_a \\ v_s = v_f - (v_f - v) \sqrt{1 - \left(\frac{\gamma_2 - \gamma_a}{1 - \gamma_a} \right)^2} & \text{when } \gamma_2 > \gamma_a \end{cases}$$

with v = initial Poisson's ratio, v_f = maximum Poisson's ratio at failure = 0.42, γ_a = 0.7.

3.8 Rubber and foam material models

- The rubber and foam material models in ADINA are the **Mooney-Rivlin**, **Ogden**, **Arruda-Boyce** and **hyper-foam** material models.
- These material models can be employed with the **2-D solid** and **3-D solid** elements.
- These material models can be used with the **large displacement/large strain** formulation. A TL formulation is employed.
- These material models include the following effects:
 - ▶ isotropic effects (required), see Section 3.8.1
 - ▶ viscoelastic effects (optional), see Section 3.8.2
 - ▶ Mullins effects (optional), see Section 3.8.3
 - ▶ orthotropic effects (optional), see Section 3.8.4
 - ▶ thermal strain effects (optional), see Section 3.8.5
- These material models allow three types of temperature dependence:
 - ▶ no temperature dependence
 - ▶ time-temperature superposition for viscoelastic effects, see

Section 3.8.6. This type of material is referred to as a thermorheologically simple material (TRS material).

- ▶ full temperature dependence, see Section 3.8.7.

Table 3.8-1 shows a summary of the features.

Table 3.8-1: Summary of features for rubber and foam material models

	Viscoelastic effects ¹	Mullins effects ¹	Orthotropic effects	Thermal strain effects
No temperature dependence	yes	yes	yes	no
TRS temperature dependence	yes	yes	yes	yes
Full temperature dependence	yes	yes	yes	yes

1) Viscoelastic effects and Mullins effects cannot be used together.

- When the material temperature dependence is TRS or full, heat generation from the rubber-like materials is included in TMC (thermo-mechanical-coupling) analysis.

3.8.1 Isotropic hyperelastic effects

- The rubber and foam material models include the following models for isotropic hyperelastic effects:
 - ▶ Mooney-Rivlin
 - ▶ Ogden
 - ▶ Arruda-Boyce
 - ▶ hyper-foam
- Only the Mooney-Rivlin and Ogden material models are supported in explicit dynamic analysis.

- The isotropic hyperelastic effects are mathematically described by specifying the dependence of the strain energy density (per unit original volume) W on the Green-Lagrange strain tensor ε_{ij} .
- We now give a brief summary of the quantities and concepts used. For more information, refer to ref KJB, section 6.6.2. Here and below, we omit the usual left superscripts and subscripts for ease of writing. Unless otherwise stated, all quantities are evaluated at time t and referred to reference time 0.

Useful quantities are the Cauchy-Green deformation tensor C_{ij} , given by

$$C_{ij} = 2\varepsilon_{ij} + \delta_{ij} \quad (3.8-1)$$

where δ_{ij} is the Kronecker delta; the principal invariants of the Cauchy-Green deformation tensor,

$$I_1 = C_{kk}, \quad I_2 = \frac{1}{2}(I_1^2 - C_{ij}C_{ij}), \quad I_3 = \det \mathbf{C} \quad (3.8-2a,b,c)$$

the reduced invariants

$$J_1 = I_1 I_3^{-\frac{1}{3}}, \quad J_2 = I_2 I_3^{-\frac{2}{3}}, \quad J_3 = I_3^{\frac{1}{2}}, \quad (3.8-3a,b,c)$$

the stretches λ_i where the λ_i 's are the square roots of the principal stretches of the Cauchy-Green deformation tensor; and the reduced stretches

$$\lambda_i^* = \lambda_i (\lambda_1 \lambda_2 \lambda_3)^{-\frac{1}{3}} \quad (3.8-4)$$

Note that

$$J_3 = \lambda_1 \lambda_2 \lambda_3 \quad (3.8-5)$$

is the volume ratio (ratio of the deformed volume to the undeformed volume).

The strain energy density W is written in terms of the invariants or stretches. In many cases, the strain energy density is conveniently written as the sum of the deviatoric strain energy density W_D and the volumetric strain energy density W_V .

With knowledge of how the strain energy density W depends on the Green-Lagrange strain tensor (through the invariants or stretches), the 2nd Piola-Kirchhoff stress tensor is evaluated using

$$S_{ij} = \frac{1}{2} \left(\frac{\partial W}{\partial \varepsilon_{ij}} + \frac{\partial W}{\partial \varepsilon_{ji}} \right) \quad (3.8-6)$$

and the incremental material tensor is evaluated using

$$C_{ijrs} = \frac{1}{2} \left(\frac{\partial S_{ij}}{\partial \varepsilon_{rs}} + \frac{\partial S_{ij}}{\partial \varepsilon_{sr}} \right) \quad (3.8-7)$$

3.8.1.1 Mooney-Rivlin model

- The Mooney-Rivlin material model is based on the following expression:

$$\begin{aligned} W_D = & C_1(I_1 - 3) + C_2(I_2 - 3) + C_3(I_1 - 3)^2 + C_4(I_1 - 3)(I_2 - 3) + \\ & C_5(I_2 - 3)^2 + C_6(I_1 - 3)^3 + C_7(I_1 - 3)^2(I_2 - 3) + \\ & C_8(I_1 - 3)(I_2 - 3)^2 + C_9(I_2 - 3)^3 + D_1(\exp(D_2(I_1 - 3)) - 1) \end{aligned} \quad (3.8-8)$$

where C_1 to C_9 and D_1 to D_2 are material constants.

- This strain energy density expression assumes a totally incompressible material ($I_3 = 1$) and is modified as explained below for plane strain, axisymmetric or 3-D analysis.

Plane stress analysis In plane stress analysis, the material is assumed to be totally incompressible. Therefore W_V is zero and

$W = W_D$. A displacement-based finite element formulation is used, in which the incompressibility condition of the material is imposed by calculating the appropriate thickness of the material.

ref. KJB
Section 6.6.2

Plane strain, axisymmetric and 3-D analysis: In plane strain, axisymmetric and 3-D analysis, the material is modeled as compressible (that is, the bulk modulus is not infinite), but you can set the bulk modulus high so that the material is “almost incompressible”.

The Mooney-Rivlin strain energy density equation is modified by: 1) substituting for the invariants I_1, I_2 the reduced invariants J_1, J_2 , 2) removing the condition $I_3 = 1$, and 3) adding the volumetric strain energy density

$$W_V = \frac{1}{2} \kappa (J_3 - 1)^2 \quad (3.8-9)$$

where κ is the bulk modulus. This expression for the volumetric strain energy density yields the following relationship between the pressure and the volume ratio:

$$p = -\kappa (J_3 - 1) \quad (3.8-10)$$

The Mooney-Rivlin material model is intended for use when the bulk to shear modulus ratio is large. The large bulk-to-shear modulus ratio causes numerical difficulties for axisymmetric, plane strain and 3D elements. The mixed u/p formulation pressure formulation can be used (and is the default) for these elements. This formulation avoids the numerical difficulties by use of a separately interpolated pressure, as described in the following reference:

- ref. T. Sussman and K.J. Bathe, "A Finite Element Formulation for Nonlinear Incompressible Elastic and Inelastic Analysis," *J. Computers and Structures*, Vol. 26, No. 1/2, pp. 357-409, 1987.

The displacement-based formulation can also be used, but is not recommended.

Selection of material constants: The Mooney-Rivlin material description used here has constants C_1 to C_9 , constants D_1 to D_2 and the bulk modulus κ . Strictly speaking, this material law is termed a higher-order or generalized Mooney-Rivlin material law.

Choosing only $C_1 \neq 0$ yields the neo-Hookean material law, and choosing only $C_1 \neq 0, C_2 \neq 0$ yields the standard two-term Mooney-Rivlin material law. In ADINA, constants C_3 to C_9 can be chosen to more closely fit experimental data.

Constants D_1 and D_2 are primarily intended for modeling certain biological materials and need not be used otherwise.

The small strain shear modulus and small strain Young's modulus can be written in terms of these constants as (assuming $\kappa = \infty$)

$$G = 2[(C_1 + C_2) + D_1 D_2] \quad (3.8-11)$$

$$E = 6[(C_1 + C_2) + D_1 D_2] \quad (3.8-12)$$

These moduli must be greater than zero.

- The bulk modulus κ is used to model the compressibility of the material for plane strain, axisymmetric and 3-D analysis.
- ADINA assumes a default for the bulk modulus based on small strain near-incompressibility, i.e.,

$$\kappa = \frac{E}{3(1-2\nu)} \quad \text{with } \nu = 0.499 \quad (3.8-13)$$

where E is the small strain Young's modulus or, in terms of the small strain shear modulus G ,

$$\kappa = \frac{2G(1+\nu)}{3(1-2\nu)} = 500G \quad \text{for } \nu = 0.499 \quad (3.8-14)$$

This rule of thumb can be used to estimate the bulk modulus in the absence of experimental data. However, you can use lower values of the bulk modulus to model compressible materials.

- In explicit analysis, the program assumes the same bulk modulus based on small strain near-incompressibility. However, this can significantly reduce the stable time step. In such cases, it is better to use one that results in $\nu = 0.49$.
- When explicit analysis with automatic time step calculation is used for a Mooney-Rivlin material, the critical time step is governed by the dilatational wave speed. This is most frequently an acceptable assumption since the material is almost incompressible.
- As the material deforms, the bulk to shear modulus ratio may change, because the instantaneous shear modulus is dependent on the amount of deformation. A value of the bulk modulus that corresponds to near incompressibility for small strains may not be large enough to correspond to near incompressibility for large strains.

3.8.1.2 Ogden material model

- The Ogden material model is based on the following expression:

$$W_D = \sum_{n=1}^9 \left(\frac{\mu_n}{\alpha_n} \left[\lambda_1^{\alpha_n} + \lambda_2^{\alpha_n} + \lambda_3^{\alpha_n} - 3 \right] \right) \quad (3.8-15)$$

where μ_n and α_n are the Ogden material constants.

- This strain energy density expression assumes a totally incompressible material ($I_3 = 1$) and is modified as explained below for plane strain, axisymmetric or 3-D analysis.

Plane stress analysis: The material is assumed to be totally incompressible. Therefore W_V is zero and $W = W_D$. A displacement-based finite element formulation is used, exactly as for the Mooney-Rivlin material model described above.

Plane strain, axisymmetric and 3-D analysis: The material is modeled as compressible (that is, the bulk modulus is not infinite), but you can set the bulk modulus high so that the material is “almost incompressible”.

The Ogden strain energy density equation is modified by 1) substituting the reduced stretches for the corresponding stretches, 2) removing the condition $\lambda_1\lambda_2\lambda_3 = 1$, and 3) adding the volumetric strain energy density

$$W_V = \frac{1}{2}\kappa(\lambda_1\lambda_2\lambda_3 - 1)^2 = \frac{1}{2}\kappa(J_3 - 1)^2 \quad (3.8-16)$$

where κ is the bulk modulus. The relationship between the pressure and the volumetric ratio is the same as for the Mooney-Rivlin material description.

Either the u/p formulation (default) or the displacement-based formulation can be used. For comments about the u/p formulation, see the corresponding comments in the Mooney-Rivlin material description.

Selection of material constants: The Ogden material description used here has 19 constants: $\mu_n, \alpha_n, n = 1, \dots, 9$ and the bulk modulus. Choosing only $\mu_n, \alpha_n \neq 0, n = 1, 2, 3$ the standard three-term Ogden material description is recovered.

The small strain shear modulus and small strain Young's modulus can be written as (assuming $\kappa = \infty$)

$$G = \frac{1}{2} \sum_{n=1}^9 \mu_n \alpha_n \quad (3.8-17)$$

$$E = \frac{3}{2} \sum_{n=1}^9 \mu_n \alpha_n \quad (3.8-18)$$

These moduli must be greater than zero.

- When explicit analysis with automatic time step calculation is used for an Ogden material, the critical time step is governed by the dilatational wave speed. This is most frequently an acceptable assumption since the material is almost incompressible.
- The bulk modulus κ is used to model the compressibility of the

material for plane strain, axisymmetric and 3-D analysis. For comments about the bulk modulus, see the corresponding comments about the bulk modulus in the Mooney-Rivlin material description.

3.8.1.3 Arruda-Boyce material model

- The Arruda-Boyce model is based on the following expression:

$$W_D = \mu \sum_{i=1}^5 \left[\frac{C_i}{\lambda_m^{2i-2}} (I_1^i - 3^i) \right] \quad (3.8-19)$$

in which $C_1 = \frac{1}{2}$, $C_2 = \frac{1}{20}$, $C_3 = \frac{11}{1050}$, $C_4 = \frac{19}{7050}$,

$C_5 = \frac{519}{673750}$ and in which there are two material constants: μ is

the initial shear modulus and λ_m is the locking stretch.

- The Arruda-Boyce material model is described in the following reference:

ref. E.M. Arruda and M. C. Boyce, “A three-dimensional constitutive model for the large stretch behavior of rubber elastic materials”, *J. Mech. Phys. Solids*, Vol., 41 (2), pp 389-412 (1993).

- This strain energy density expression assumes a totally incompressible material ($I_3 = 1$) and is modified as explained below for plane strain, axisymmetric or 3-D analysis.

Plane stress analysis: The material is assumed to be totally incompressible. Therefore W_V is zero and $W = W_D$. A displacement-based finite element formulation is used, exactly as for the Mooney-Rivlin material model described above.

ref. KJB
Section 6.6.2

Plane strain, axisymmetric and 3-D analysis: The material is modeled as compressible (that is, the bulk modulus is not infinite),

but you can set the bulk modulus high so that the material is “almost incompressible”.

The Arruda-Boyce strain energy density equation is modified by: 1) substituting for the strain invariant I_1 the reduced strain invariants J_1 , 2) removing the condition $I_3 = 1$, and 3) adding the volumetric energy term

$$W_V = \frac{\kappa}{2} \left[\frac{(J_3^2 - 1)}{2} - \ln J_3 \right] \quad (3.8-20)$$

where κ is the small-strain bulk modulus. The relationship between the pressure and the volume ratio is

$$p = \frac{\kappa}{2} \left(J_3 - \frac{1}{J_3} \right) \quad (3.8-21)$$

Either the u/p formulation (default) or the displacement-based formulation can be used. For comments about the u/p formulation, see the corresponding comments in the Mooney-Rivlin material description.

When the u/p formulation is used, there should be at least one solution unknown. This is because the constraint equation used in the u/p formulation is nonlinear in the unknown pressures. Therefore equilibrium iterations are required for convergence, even when all of the displacements in the model are prescribed.

Selection of material constants: The default values of the material constants are $\mu = 12.6008$, $\lambda_m = 1$, $\kappa = 63000$. It is likely that you will need to change these values in your analysis.

3.8.1.4 Hyper-foam material model

- The hyper-foam model is based on the following expression:

$$W = \sum_{n=1}^N \frac{\mu_n}{\alpha_n} \left[\lambda_1^{\alpha_n} + \lambda_2^{\alpha_n} + \lambda_3^{\alpha_n} - 3 + \frac{1}{\beta_n} (J_3^{-\alpha_n \beta_n} - 1) \right] \quad (3.8-22)$$

in which there are the material constants $\mu_n, \alpha_n, \beta_n, n = 1, \dots, N$. The maximum value of N is 9.

- A material model similar to the hyper-foam material model is described in the following reference:

ref. B. Storåkers, “On material representation and constitutive branching in finite compressible elasticity”, *J. Mech. Phys. Solids*, Vol., 34(2), pp 125-145 (1986).

In this reference, β_n is the same for all values of n .

- When it is required to compute the deviatoric and volumetric parts of the strain energy density, we use

$$W_D = \sum_{n=1}^N \frac{\mu_n}{\alpha_n} \left[\lambda_1^{\alpha_n} + \lambda_2^{\alpha_n} + \lambda_3^{\alpha_n} - 3 J_3^{\alpha_n / 3} \right] \quad (3.8-23)$$

$$W_V = \sum_{n=1}^N \frac{\mu_n}{\alpha_n} \left[3 (J_3^{\alpha_n / 3} - 1) + \frac{1}{\beta_n} (J_3^{-\alpha_n \beta_n} - 1) \right] \quad (3.8-24)$$

Notice that $W = W_D + W_V$. This decomposition of the strain energy density has the advantage that the stresses obtained from the deviatoric and volumetric parts separately are zero when there are no deformations:

$$\begin{aligned} S_{ij}^D \Big|_{\varepsilon_{ij}=0} &= \frac{1}{2} \left(\frac{\partial W_D}{\partial \varepsilon_{ij}} + \frac{\partial W_D}{\partial \varepsilon_{ji}} \right) \Bigg|_{\varepsilon_{ij}=0} = 0, \\ S_{ij}^V \Big|_{\varepsilon_{ij}=0} &= \frac{1}{2} \left(\frac{\partial W_V}{\partial \varepsilon_{ij}} + \frac{\partial W_V}{\partial \varepsilon_{ji}} \right) \Bigg|_{\varepsilon_{ij}=0} = 0 \end{aligned}$$

Notice that W_D contains the volumetric part of the motion through the term $3J_3^{\alpha_n/3}$. Therefore W_D is not entirely deviatoric.

Plane stress analysis: The material is *not* assumed to be totally incompressible. A displacement-based finite element formulation is used.

Plane strain, axisymmetric and 3-D analysis: The material is *not* assumed to be totally incompressible.

Because both W_D and W_V contain the volumetric part of the motion, the mixed u/p formulation cannot be used with the hyper-foam material. A displacement-based formulation is used.

Selection of material constants: The hyper-foam material description used here has 27 constants: $\mu_n, \alpha_n, \beta_n, n = 1, \dots, 9$.

The small strain shear modulus and small strain bulk modulus can be written as

$$G = \frac{1}{2} \sum_{n=1}^9 \mu_n \alpha_n \quad (3.8-25)$$

$$\kappa = \sum_{n=1}^9 \left(\beta_n + \frac{1}{3} \right) \mu_n \alpha_n \quad (3.8-26)$$

These moduli must be greater than zero, hence we note that β_n should be greater than $-1/3$.

When all of the β_n are equal to each other = β , then the Poisson's ratio is related to β using

$$\beta = \frac{\nu}{1 - 2\nu} \quad (3.8-27)$$

The default values of the material constants are $\mu_1 = 1.85$, $\alpha_1 = 4.5$, $\beta_1 = 9.2$, $\mu_2 = 9.2$, $\alpha_2 = -4.5$, $\beta_2 = 9.2$. It is likely that you will need to change these values in your analysis.

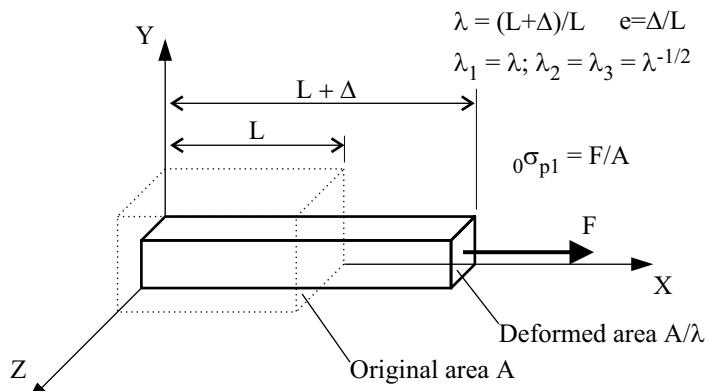
- The hyper-foam material model is generally used for highly compressible elastomers.
If the ratio of the bulk modulus to shear modulus is high (greater than about 10), the material is almost incompressible and we recommend that one of the other hyperelastic materials be used.

3.8.1.5 Curve fitting

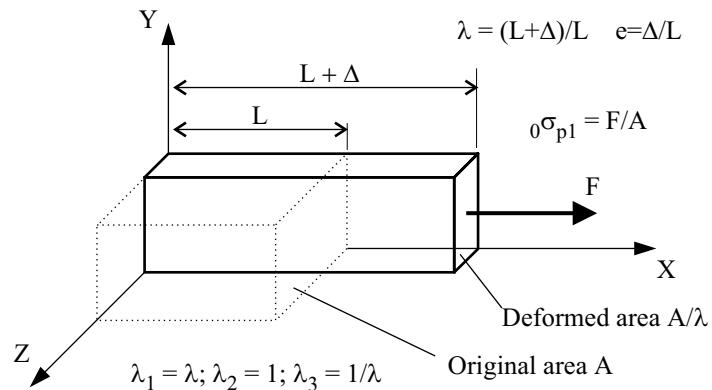
- You can have the AUI determine the material constants for the Mooney-Rivlin, Ogden, Arruda-Boyce or hyper-foam materials by a curve-fit to experimental stress-strain data.
- The AUI allows for data from three experimental cases as shown in Figure 3.8-1: simple tension, pure shear and equibiaxial tension. Input from a single experiment from one of these cases or a combination of data from any two, or all three of these experimental cases is allowed.

A single set of constants representing a best fit in the least squares sense is calculated. You must select the chosen order of the approximation (giving the number of constants used in the model). Note: the total number of data points must be greater than the number of material constants to be determined.

For each experimental case, you can enter either the stretch or engineering strain in the strain column and you enter the engineering stress in the stress column. Refer to Figure 3.8-1 to see the definitions of stretch, engineering strain and engineering stress for the experimental cases.

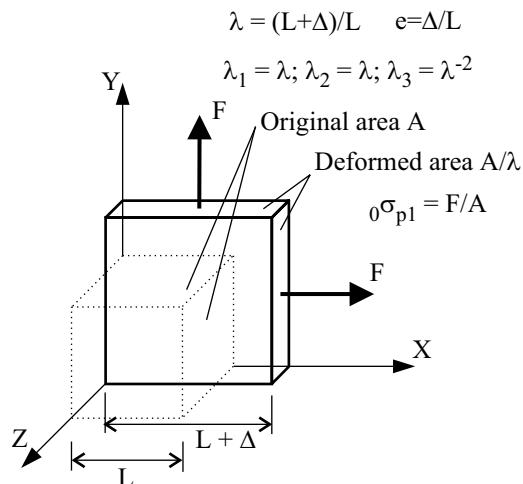


a) Simple tension test (incompressible material)



b) Pure shear test (incompressible material)

Figure 3.8-1: Experimental test cases for rubber and foam materials



c) Equibiaxial tension test (incompressible material)

Figure 3.8-1: (continued)

Mooney-Rivlin model: The strain energy density for total incompressibility is given in Eq. 3.8-8 with the material constants D_1 and D_2 set to zero.

Given M non-zero material constants, C_i , then M is equal to 2, 5 and 9 for first, second and third order approximations, respectively.

The principal stress in the direction of principal stretch λ_1 is given by

$${}_0\sigma_{pl} = \frac{\partial W}{\partial \lambda} = {}_0\sigma_{pl}(C_i, \lambda) \quad (i=1, \dots, M) \quad (3.8-28)$$

Notice that this is the engineering principal stress, that is, the force divided by the original area of the specimen. Thus, we have an expression for the principal stress that is linear in terms of the material constants C_i , and dependent on the principal stretch determined from the allowed experimental cases. Hence, given a set of data points (λ_k, σ_k) from one or more of the test cases, we can establish the least squares fit by a curve of the form given by Eq. (3.8-28).

Consider the sum of discrete error terms at the given data points of one of the experimental test cases (simple tension, pure shear or

equibiaxial tension)

$$S = \sum_{k=1}^N (\sigma_k - \hat{\sigma}_{pl}(\lambda_k))^2 \quad (3.8-29)$$

where $\hat{\sigma}_{pl}$ is the approximating function dependent on the constants C_i ($i = 1, \dots, M$) given by Eq. (3.8-28). We can expand this sum to include data from all three test cases, i.e.:

$$S = \sum_{j=1}^3 \left(\sum_{k=1}^{N^j} (\sigma_k^j - \hat{\sigma}_1^j(\lambda_k^j))^2 \right) \quad (3.8-30)$$

where the superscript j denotes the experimental test case (simple tension $j = 1$, pure shear $j = 2$, equibiaxial tension $j = 3$). Note: the form of $\hat{\sigma}_1^j$ differs for $j = 1, 2, 3$ due to the different principal stretch terms λ_2 and λ_3 .

The least squares fit is determined by minimizing S in Eq. (3.8-30) with respect to the constants C_i , which gives M algebraic equations for the constants C_i ($i = 1, \dots, M$).

Ogden model: The strain energy density function is, for total incompressibility, given by Eq. (3.8-15). Thus, for an M -term Ogden model, there are $2M$ constants, i.e., the pairs (μ_i, α_i) for $i = 1, \dots, M$.

Assuming the principal stretches for the three test cases shown in Figure 3.8-1, the following form for the principal stress ${}_0\sigma_{pl}$ can be derived:

$${}_0\sigma_{pl} = \sum_{i=1}^M \mu_i (\lambda^{\alpha_i-1} - \lambda^{c\alpha_i-1}) \quad (3.8-31)$$

with $c = -0.5$ for simple tension, -1.0 for pure shear, -2.0 for equibiaxial tension.

Equation (3.8-31) is linear in terms of the coefficients μ_i , but nonlinear in terms of the exponents α_i . A nonlinear least squares

algorithm could be employed to iterate to a solution for the constants (μ_i, α_i) . However, in our curve-fitting implementation in the AUI, you specify the desired exponent terms α_i (with the default $\alpha_i = i$), and the least squares fit of the resulting function yields the corresponding coefficients μ_i . You can select the number of terms M desired ($1 \leq M \leq 9$). The assumption of integer values for α_i has in practice given excellent fits to experimental data for rubber materials. Nevertheless, you can use other values for α_i to possibly give a better fit for specific materials.

Arruda-Boyce model: The strain energy density function is, for total incompressibility, given by Eq. (3.8-19). There are only two material constants, μ and λ_m .

Assuming the principal stretches for the three test cases shown in Figure 3.8-1, the following forms for the principal stress ${}_0\sigma_{pl}$ can be derived:

Uniaxial tension:

$${}_0\sigma_{pl} = 2\mu(\lambda - \lambda^{-2}) \sum_{i=1}^5 \frac{iC_i}{\lambda_m^{2i-2}} I_1^{i-1}, \quad I_1 = \lambda^2 + 2\lambda^{-1} \quad (3.8-32)$$

Pure shear:

$${}_0\sigma_{pl} = 2\mu(\lambda - \lambda^{-3}) \sum_{i=1}^5 \frac{iC_i}{\lambda_m^{2i-2}} I_1^{i-1}, \quad I_1 = \lambda^2 + \lambda^{-2} + 1 \quad (3.8-33)$$

Equibiaxial tension:

$${}_0\sigma_{pl} = 2\mu(\lambda - \lambda^{-5}) \sum_{i=1}^5 \frac{iC_i}{\lambda_m^{2i-2}} I_1^{i-1}, \quad I_1 = 2\lambda^2 + \lambda^{-4} \quad (3.8-34)$$

The AUI uses a nonlinear curve-fitting algorithm to determine μ and λ_m .

Hyper-foam model: The strain energy density function is given by Eq. (3.8-22). However, we seek a curve-fit for which all of the β_n terms are equal. Thus, for an M-term hyper-foam model, there are

$2M+1$ material constants, i.e., the constants (μ_n, α_n) for $n = 1, \dots, M$, and β .

First β is determined using information about the lateral stretch λ_T . There are two options for entering the lateral stretch information:

1) Specify the Poisson's ratio. β is then calculated using (3.8-27).

2) Enter the lateral stretches as part of the stress-strain data input. Then β is determined with the use of the following formulas:

Uniaxial tension:

$$\lambda_T = \lambda^{-\frac{\beta}{1+2\beta}} \quad (3.8-35)$$

Pure shear:

$$\lambda_T = \lambda^{-\frac{\beta}{1+\beta}} \quad (3.8-36)$$

Equibiaxial tension:

$$\lambda_T = \lambda^{-\frac{2\beta}{1+\beta}} \quad (3.8-37)$$

Since each of these formulas can be also written in the form $\ln \lambda_T = -f(\beta) \ln \lambda$, a linear curve-fit is performed to obtain β .

Next, the constants (μ_n, α_n) are determined. Assuming the principal stretches for the three test cases shown in Figure 3.8-1, the following form for the principal stress ${}_0\sigma_{pI}$ can be derived:

$${}_0\sigma_{pI} = \frac{1}{\lambda} \sum_{n=1}^M \mu_n \left(\lambda^{\alpha_n} - J_3^{-\alpha_n \beta} \right) \quad (3.8-38)$$

where $J_3 = \lambda \lambda_T^2$ for uniaxial tension, $J_3 = \lambda \lambda_T$ for pure shear and

$J_3 = \lambda^2 \lambda_T$ for equibiaxial tension. (3.8-38) is then used as the basis of a nonlinear curve-fitting algorithm to obtain the constants (μ_n, α_n) .

Notes for curve-fitting for the Mooney-Rivlin and Ogden material models:

The system of equations used to solve for the least squares fitted constants is likely to be ill-conditioned. Direct use of Gaussian elimination may yield constants with large magnitudes, which are balanced to cancel out when the fitted function $\hat{\sigma}$ is evaluated. Furthermore, and perhaps more importantly, there is no guarantee that the fitted function has physically feasible behavior outside the range of the data supplied.

The reason for the ill-conditioning of the least squares system matrix is that the data may be fitted equally well by two or more of the "basis" functions or combinations thereof, thus rendering the matrix nearly singular.

The AUI provides an alternative matrix solution algorithm to the Gauss-Jordan elimination technique, namely that of Singular Value Decomposition (SVD) described in

ref. Forsythe, G.E., Malcom, M.A. and Moler, C.B.,
Computer Methods for Mathematical Computations,
Chapter 9, Prentice Hall, Inc. Englewood Cliffs, NJ,
1977.

The essence of the SVD algorithm is that combinations of the basis functions which correspond to near zero eigenvalues of the matrix system are eliminated.

Successive elimination of the near singular terms always increases the error (in the least squares sense). However, the increase in error is often small compared to the overall improvement in the fitted solution – the constants obtained are usually smaller in magnitude and physically reasonable.

The SVD algorithm is controlled by successively eliminating combinations corresponding to the smallest eigenvalues until a fitted function is obtained which gives a monotonic increase in stress with strain.

Depending on data, the number of terms eliminated from a higher order approximation may be large, giving a poor fit, even though the monotonicity criterion has been achieved.

As an example, consider the curve fitting results illustrated in Fig. 3.8-2. Fig. 3.8-2(a) shows the fitted curves for a 9-term ($\alpha_i = i, i = 1, \dots, 9$) Ogden model approximating a set of data points for a simple tension test. The sum of error squares S is given in the key of the graphs.

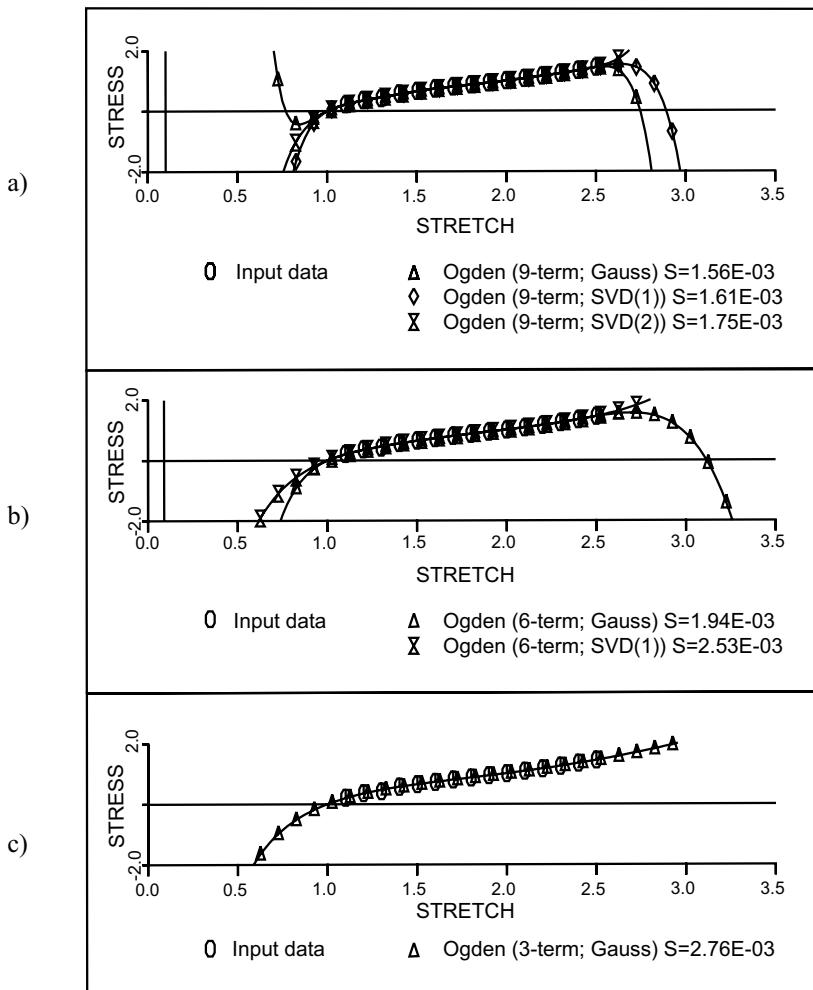


Figure 3.8-2: Curve fitting results for Ogden material model

While the Gaussian elimination curve has the minimum S , its solution is asymptotically unacceptable. Two terms were removed to give a monotonic increasing solution (SVD(2)) which still gives

a good approximation to the input data. Fig. 3.8-2(b) shows the fitted curves for a 6-term Ogden model for the same input data. A monotonic increasing solution was obtained after removal of one term (SVD(1)), while the Gauss elimination solution was not asymptotically feasible. For the 3-term Ogden fit to the data, the Gauss elimination solution was accepted since it provided a monotonic increasing curve, as shown in Fig. 3.8-2(c).

Notes for curve-fitting for the Arruda-Boyce and hyper-foam material models: The Marquardt-Levenberg algorithm from the following reference is used in the curve fitting:

ref. W.H. Press, B.P. Flannery, S.A. Teukolsky and W.T. Vetterling, *Numerical Recipes – the Art of Scientific Computing (Fortran Version)*, Cambridge University Press, pp 521-528 (1990).

General comments: It can be seen that the curve fitting technique provided gives a valuable tool, but one which must be used with caution. Always check the resulting fitted function. If a physically reasonable fit is not achieved, we recommend the following strategy

- a) Use a lower order approximation.
- b) Use a different material model
- c) Provide more data points, covering a greater range of anticipated strains, including values for compression ($\lambda < 1$).

Specification of input: All of the information required for the curve fitting is contained in a data set of type curve-fitting. The curve-fitting data contains the numbers of the data sets for each of the three test cases, the order of the curve fit and other information required for the curve fit.

3.8.2 Viscoelastic effects

- Viscoelastic effects can be included in the Mooney-Rivlin, Ogden, Arruda-Boyce and hyper-foam material models.

The viscoelastic model used is due to Holzapfel, see the following references:

- ref. G. A. Holzapfel, “On large strain viscoelasticity: continuum formulation and finite element applications to elastomeric structures”, *Int. J. Num. Meth. Engng.*, Vol. 39, pp 3903-3926, 1996.
- ref. G. A. Holzapfel, *Nonlinear solid mechanics. A continuum approach for engineering*. John Wiley & Sons, Chichester, pp 278-295, 2000.
- ref. G. A. Holzapfel, “Biomechanics of soft tissue”, in Lemaitre (ed.), *Handbook of Materials Behavior Models: Nonlinear Models and Properties*, Academic Press, 2001, pp 1057-1071.

In the following, we give a brief discussion of the Holzapfel model for finite strain viscoelasticity.

Equivalent 1D model: The equivalent 1D model is shown in Fig 3.8-3. It is the same as a generalized Maxwell model with many chains. A generic chain is denoted with superscript α , as shown in the figure.

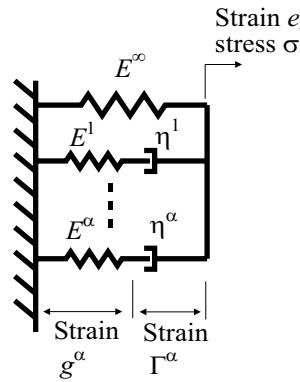


Figure 3.8-3: Equivalent 1D model for viscoelastic effects

The spring E^∞ is equivalent to the elastic stiffness of the model. Each chain contains a spring with stiffness E^α and dashpot

with viscosity η^α . (Note that the superscripts ∞ and α do not denote exponentiation.) The strain in each chain is the sum of the strain in the spring g^α and the strain in the dashpot Γ^α . The observed stress is

$$\sigma = \sigma^\infty + \sum_\alpha q^\alpha \quad (3.8-39)$$

where $\sigma^\infty = E^\infty e$ is the elastic stress and $q^\alpha = E^\alpha g^\alpha = \eta^\alpha \dot{\Gamma}^\alpha$ is the stress in chain α . Using the definition $\tau^\alpha = \frac{\eta^\alpha}{E^\alpha}$ and the assumption $E^\alpha = \beta^\alpha E^\infty$, the following expression is obtained:

$$\dot{q}^\alpha + \frac{1}{\tau^\alpha} q^\alpha = \beta^\alpha \dot{\sigma}^\infty \quad (3.8-40)$$

Assuming that ${}^0 q^\alpha = 0$, (3.8-40) can be written in convolution form as

$$q^\alpha = \int_0^t \exp\left(-\frac{t-t'}{\tau^\alpha}\right) \beta^\alpha {}^{t'} \dot{\sigma}^\infty dt' \quad (3.8-41)$$

from which the total stress is

$$\sigma = \int_0^t E^\infty \left[1 + \sum_\alpha \beta^\alpha \exp\left(-\frac{t-t'}{\tau^\alpha}\right) \right] {}^{t'} \dot{e} dt' \quad (3.8-42)$$

Evidently the relaxation modulus is

$E(t) = E^\infty \left[1 + \sum_\alpha \beta^\alpha \exp\left(-\frac{t}{\tau^\alpha}\right) \right]$ which is a Prony series expression.

The dissipation in dashpot α is

$$D^\alpha = q^\alpha \dot{\Gamma}^\alpha = q^\alpha (\dot{e} - \dot{g}^\alpha) = q^\alpha \left(\dot{e} - \frac{1}{\beta^\alpha} \frac{\dot{q}^\alpha}{E^\infty} \right) \quad (3.8-43)$$

and the total dissipation is $D = \sum_\alpha D^\alpha$. In the above, the viscoelastic material constants for each chain are τ^α and β^α .

Potential-based 1D model: The 1D model can be written in terms of a potential as follows:

$$\Psi = \Psi^\infty(e) + \sum_\alpha \Psi^\alpha(g^\alpha) \quad (3.8-44)$$

where $\Psi^\infty(e) = \frac{1}{2} E^\infty e^2$ is the strain energy of the elastic chain and $\Psi^\alpha(g^\alpha) = \frac{1}{2} E^\alpha (g^\alpha)^2$ is the strain energy in the spring of chain α . In terms of β^α , $\Psi^\alpha(g^\alpha) = \beta^\alpha \Psi^\infty(g^\alpha)$. The 1D model is recovered by defining

$$\sigma = \left. \frac{\partial \Psi}{\partial e} \right|_{\Gamma^\alpha \text{ fixed}}, \quad q^\alpha = - \left. \frac{\partial \Psi}{\partial \Gamma^\alpha} \right|_{e \text{ fixed}} \quad (3.8-45a,b)$$

Notice that (3.8-39) and (3.8-45a) imply

$$q^\alpha = \left. \frac{\partial \Psi^\alpha}{\partial e} \right|_{\Gamma^\alpha \text{ fixed}} = \left. \frac{\partial \Psi^\alpha}{\partial g^\alpha} \right|_{\Gamma^\alpha \text{ fixed}}.$$

Finite strain model: The finite strain model is derived from the potential-based 1D model as follows. The elastic potential is defined as

$$\Psi^\infty(\boldsymbol{\varepsilon}) = W(\boldsymbol{\varepsilon}) \quad (3.8-46)$$

where $W(\boldsymbol{\varepsilon}_{ij})$ is the strain energy density from the elastic part of

the material model. The potential of each chain α is defined as

$$\begin{aligned}\Psi^\alpha(\varepsilon_{ij}, \Gamma_{ij}^\alpha) &= \beta^\alpha W(G_{ij}^\alpha), \text{ usage=combined} \\ &= \beta^\alpha W_D(G_{ij}^\alpha), \text{ usage=deviatoric} \\ &= \beta^\alpha W_V(G_{ij}^\alpha), \text{ usage=volumetric}\end{aligned}\quad (3.8-47a,b,c)$$

in which the usage flag (which is a user-input flag) determines whether the entire elastic strain energy density, deviatoric strain energy density or volumetric strain energy density is taken for chain α . Here G_{ij}^α is analogous to the strain in the 1D spring g^α , and we assume $G_{ij}^\alpha = \varepsilon_{ij} - \Gamma_{ij}^\alpha$. Note that with this definition of G_{ij}^α , we have

$$S_{ij} = \frac{\partial \Psi}{\partial \varepsilon_{ij}} \Bigg|_{\Gamma_{ij}^\alpha \text{ fixed}}, \quad Q_{ij}^\alpha = \frac{\partial \Psi^\alpha}{\partial \varepsilon_{ij}} \Bigg|_{\Gamma_{ij}^\alpha \text{ fixed}} = -\frac{\partial \Psi}{\partial \Gamma_{ij}^\alpha} \Bigg|_{\varepsilon_{ij} \text{ fixed}} = \frac{\partial \Psi}{\partial G_{ij}^\alpha} \quad (3.8-48a,b)$$

where S_{ij} are the 2nd Piola-Kirchhoff stresses and Q_{ij}^α is analogous to the stress q^α .

Following exactly the same arguments as in the 1D case, we obtain

$$\dot{Q}_{ij}^\alpha + \frac{1}{\tau^\alpha} Q_{ij}^\alpha = \beta^\alpha \dot{S}_{ij}^\alpha \quad (3.8-49)$$

Assuming that ${}^0Q_{ij}^\alpha = 0$, (3.8-50) can be written in convolution form as

$$Q_{ij}^\alpha = \int_0^t \exp\left(-\frac{t-t'}{\tau^\alpha}\right) \beta^\alpha {}^{t'} \dot{S}_{ij}^\alpha dt' \quad (3.8-50)$$

and (3.8-50) can be numerically approximated using

$${}^{t+\Delta t}Q_{ij}^{\alpha} = \exp\left(-\frac{\Delta t}{\tau^{\alpha}}\right){}^tQ_{ij}^{\alpha} + \beta^{\alpha} \frac{1 - \exp\left(-\frac{\Delta t}{\tau^{\alpha}}\right)}{\frac{\Delta t}{\tau^{\alpha}}} \left({}^{t+\Delta t}S_{ij}^{\alpha} - {}^tS_{ij}^{\alpha}\right). \quad (3.8-51)$$

(3.8-51) is exact when S_{ij}^{α} does not change during the time step, and is more accurate than the formula given by Holzapfel:

$${}^{t+\Delta t}Q_{ij}^{\alpha} = \exp\left(-\frac{\Delta t}{\tau^{\alpha}}\right){}^tQ_{ij}^{\alpha} + \beta^{\alpha} \exp\left(-\frac{\Delta t}{2\tau^{\alpha}}\right) \left({}^{t+\Delta t}S_{ij}^{\alpha} - {}^tS_{ij}^{\alpha}\right) \quad (3.8-52)$$

especially when $\frac{\Delta t}{\tau^{\alpha}}$ is large.

Dissipation calculations: If the dissipation is required, it is calculated using

$$D^{\alpha} = Q_{ij}^{\alpha} \dot{\Gamma}_{ij}^{\alpha} = Q_{ij}^{\alpha} \left(\dot{\epsilon}_{ij} - \dot{G}_{ij}^{\alpha} \right) \quad (3.8-53)$$

where

$$\frac{\partial^2 \Psi^{\alpha}}{\partial G_{ij}^{\alpha} \partial G_{rs}^{\alpha}} \dot{G}_{rs}^{\alpha} = \dot{Q}_{ij}^{\alpha} \quad (3.8-54)$$

is used to compute the unknown \dot{G}_{rs}^{α} from the known \dot{Q}_{ij}^{α} .

If usage=combined,

$$\frac{\partial^2 \Psi^{\alpha}}{\partial G_{ij}^{\alpha} \partial G_{rs}^{\alpha}} = \beta^{\alpha} \frac{\partial^2 W}{\partial \epsilon_{ij} \partial \epsilon_{rs}} = \beta^{\alpha} C_{ijrs}. \quad (3.8-55)$$

where the tensor C_{ijrs} is evaluated at the strain state G_{ij}^{α} . The dissipation calculation requires the solution of a set of simultaneous linear equations of at most order 6 (in the 3D case) at each integration point.

If usage=deviatoric,

$$\frac{\partial^2 \Psi^\alpha}{\partial G_{ij}^\alpha \partial G_{rs}^\alpha} = \beta^\alpha \frac{\partial^2 W_D}{\partial \varepsilon_{ij} \partial \varepsilon_{rs}} = \beta^\alpha (C_D)_{ijrs} \quad (3.8-56)$$

where the tensor $(C_D)_{ijrs}$ is evaluated at the strain state G_{ij}^α . Here, the dissipation calculation requires a singular value decomposition of $(C_D)_{ijrs}$, since $(C_D)_{ijrs}$ has a zero eigenvalue. A similar situation applies when usage=volumetric, except that the corresponding material tensor has only one nonzero eigenvalue.

The procedure given in (3.8-53) to (3.8-56) is only approximate, since the fundamental assumption $G_{ij}^\alpha = \varepsilon_{ij} - \Gamma_{ij}^\alpha$ strictly speaking only holds for small strain analysis.

Restrictions and recommendations: The allowed values of the usage flag depend upon the material model and finite element type, as shown in Table 3.8-2.

In view of the restrictions, we recommend that usage="combined" be used in conjunction with the hyper-foam material model, and that usage="deviatoric" be used in conjunction with the Mooney-Rivlin, Ogden and Arruda-Boyce material models.

Time-temperature superposition: The preceding derivation assumes that the viscoelastic response is not temperature-dependent. One method of including the effects of temperature is the method of time-temperature superposition.

In time-temperature superposition, the actual time t is replaced by the reduced time ζ . The relationship between the actual time and reduced time is given by

$$\frac{d\zeta}{dt} = \frac{1}{a_T(''\theta)} \quad (3.8-57)$$

where $''\theta$ is the temperature and $a_T(''\theta)$ is the shift function.

Table 3.8-2: Allowed values of the usage flag

	Mooney-Rivlin, Ogden, Arruda-Boyce		hyper-foam	
	Plane stress ¹	Plane strain, axisymmetric, 3D ²	Plane stress ³	Plane strain, axisymmetric, 3D
usage=combined	yes	no	yes	yes
usage=deviatoric	yes	yes	no	yes
usage=volumetric	no	no	no	yes

- 1) Usage cannot be equal to “volumetric”. This is because the material is assumed to be fully incompressible, hence the volumetric strain energy density is zero.
- 2) When the u/p formulation is used, the usage flag cannot be “combined” or “volumetric”. This is because the modification to the volumetric stresses caused when the usage flag is “combined” or “volumetric” is not taken into account in the u/p formulation.
- 3) The only allowable value of the usage flag is “combined”. This is because the out-of-plane stress component S_{xx} must be zero, and in the Holzapfel finite strain viscoelastic model, the only way that this can happen is if S_{xx}^∞ is zero.

Evidently

$${}^t\zeta = \int_0^t \frac{1}{a_T({}^t\theta)} dt' \quad (3.8-58)$$

The shift function used here is the WLF shift function,

$$\log_{10} a_T({}^t\theta) = -\frac{C_1({}^t\theta - \theta_{ref})}{C_2 + {}^t\theta - \theta_{ref}} \quad (3.8-59)$$

where θ_{ref} is the reference temperature and C_1 , C_2 are material

constants. Notice that as $'\theta$ increases, $a_T(''\theta)$ decreases and $\frac{d\zeta}{dt}$ increases.

For the viscoelastic model used here, the differential equation of the 1D model (3.8-40) becomes

$$\frac{dq^\alpha}{d\zeta} + \frac{1}{\tau^\alpha} q^\alpha = \beta^\alpha \frac{d\sigma^\infty}{d\zeta} \quad (3.8-60)$$

and using (3.8-57), (3.8-60) can be written as

$$\dot{q}^\alpha + \frac{1}{a_T(\theta)\tau^\alpha} q^\alpha = \beta^\alpha \dot{\sigma}^\infty \quad (3.8-61)$$

It is seen that the effect of temperature is to modify the time constants. As the temperature increases, the modified time constants become smaller, that is, the material relaxes more quickly.

The convolution equation of the finite strain model becomes

$$Q_{ij}^\alpha = \int_0^{\zeta} \exp\left(-\frac{\zeta - \zeta'}{\tau^\alpha}\right) \beta^\alpha \frac{d^{t'} S_{ij}^\infty}{d\zeta'} d\zeta' \quad (3.8-62)$$

and (3.8-62) is numerically approximated by

$${}^{t+\Delta t} Q_{ij}^\alpha = \exp\left(-\frac{\Delta\zeta}{\tau^\alpha}\right) {}^t Q_{ij}^\alpha + \beta^\alpha \frac{1 - \exp\left(-\frac{\Delta\zeta}{\tau^\alpha}\right)}{\frac{\Delta\zeta}{\tau^\alpha}} \left({}^{t+\Delta t} S_{ij}^\alpha - {}^t S_{ij}^\alpha \right) \quad (3.8-63)$$

The only additional consideration is to calculate $\Delta\zeta$, and this is done using

$$\Delta\zeta = \int_t^{t+\Delta t} \frac{1}{a_T(t')\theta} dt' \quad (3.8-64)$$

This integration is performed numerically assuming that $\ln a_T(\theta)$ varies linearly over the time step.

Heat generation: A user-specified fraction of the energy dissipated by the viscoelastic model can be considered as heat generation. This heat generation can cause heating in a TMC (thermo-mechanical-coupling) analysis.

Specification of input: To add viscoelastic effects to the rubber-like material model, you need to define the viscoelastic data using a data set of type rubber-viscoelastic, then specify the number of the rubber-viscoelastic data set in the rubber-like material model.

The rubber-viscoelastic data set includes:

- ▶ A flag indicating whether the WLF shift function is used.
- ▶ The shift function material constants C_1 , C_2
- ▶ A table with one row for each chain. Each row contains β^α , τ^α , the heat generation factor (fraction of dissipation considered as heat generation, default value is 0.0), and the usage flag (default value is deviatoric). There is no restriction on the number of chains permitted. The usage flag can be different for each chain.

In order to include time-temperature superposition, the material must have a TRS temperature dependence.

- It is seen that the dissipation calculation can be quite expensive. Furthermore the dissipation is not required for the stress solution. Therefore it is the default in ADINA to not perform the dissipation calculation. The dissipation is only calculated for the chain α when the heat generation factor is non-zero.

3.8.3 Mullins effect

When rubber is loaded to a given strain state, unloaded, then reloaded to the same strain state, the stress required for the reloading is less than the stress required for the initial loading. This phenomenon is referred to as the Mullins effect.

The Mullins effect can be included in the rubber-like materials. The material model used is the one described in the following reference:

ref. R.W. Ogden and D. G. Roxburgh, “A pseudo-elastic model for the Mullins effect in filled rubber”, *Proc. R. Soc. Lond. A* (1999) 455, 2861-2877.

We briefly summarize the main concepts below.

Fig 3.8-4 shows the Mullins effect in simple tension. On initial loading to force F_c , the specimen follows the force-deflection curve a-b-c. When the load is removed, the specimen follows the unloading curve c-d-a. On reloading to force F_c , the specimen follows the reloading curve a-d-c, and on further loading to force F_f , the specimen follows the loading curve c-e-f. When the load is removed, the specimen follows the unloading curve f-g-a, and, on reloading to force F_g , the specimen follows the reloading curve a-g-f.

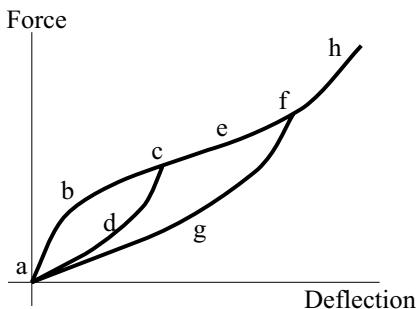


Figure 3.8-4: Mullins effect loading-unloading-reloading curves

Note that any permanent set associated with the Mullins effect

is not included in the Ogden-Roxburgh model used here.

The Ogden-Roxburgh model, as implemented in ADINA, uses the following strain energy density expression:

$$\begin{aligned}\tilde{W} &= \eta W_D(\varepsilon_{ij}) + \phi(\eta), \text{ Mooney-Rivlin, Ogden, Arruda-Boyce} \\ &= \eta W(\varepsilon_{ij}) + \phi(\eta), \text{ hyper-foam}\end{aligned}\quad (3.8-65a,b)$$

where $W(\varepsilon_{ij})$ is the total elastic strain energy density, $W_D(\varepsilon_{ij})$ is the deviatoric elastic strain energy density, η is an additional solution variable describing the amount of unloading and $\phi(\eta)$ is the damage function. \tilde{W} is referred to as the pseudo-energy function. In our implementation, the deviatoric strain energy density is used for the (almost) incompressible materials and the total strain energy density is used for compressible materials. For ease of writing, we discuss only the case of compressible materials; for incompressible materials, replace W by W_D in the equations below.

η is computed as

$$\eta = 1 - \frac{1}{r} \operatorname{erf} \left[\frac{1}{m} (W_m - W) \right] \quad (3.8-66)$$

where $\operatorname{erf}(x)$ is the error function

$$\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x \exp(-u^2) du \quad (3.8-67)$$

W_m is the maximum value of W encountered during the deformation history and m and r are material constants.

$\phi(\eta)$ is defined by

$$\frac{d\phi(\eta)}{d\eta} = -W \quad (3.8-68)$$

and is computed by numerical integration of $\dot{\phi}(\eta) = -W\dot{\eta}$. For a

given value of W_m , there is a minimum value of η computed as

$$\eta_m = 1 - \frac{1}{r} \operatorname{erf} \left[\frac{W_m}{m} \right] \quad (3.8-69)$$

The value of $\phi(\eta)$ at $\eta = \eta_m$ is written $\phi(\eta_m)$. (Note: the subscript m in the term W_m means “maximum”, but the subscript m in the term η_m means “minimum”.) The time rate of change of $\phi(\eta_m)$ can be written as

$$\dot{\phi}(\eta_m) = (1 - \eta_m) \dot{W}_m = \frac{1}{r} \operatorname{erf} \left[\frac{W_m}{m} \right] \dot{W}_m \quad (3.8-70)$$

Physically, $\phi(\eta_m)$ is interpreted as the dissipation.

During loading, $W_m = W$, $\dot{W}_m > 0$ and $\eta = 1$. Therefore $\dot{\phi}(\eta) = 0$ and $\dot{\phi}(\eta_m) > 0$ during loading.

During unloading or reloading, $W_m > W$, $\dot{W}_m = 0$ and $\eta_m \leq \eta < 1$. Therefore $\dot{\phi}(\eta) \neq 0$ and $\dot{\phi}(\eta_m) = 0$ during unloading or reloading.

Material constants m and r do not have any direct physical significance. However Fig 3.8-5 shows the dependence of an unloading-reloading curve in simple tension on these parameters. It is seen that, for an unloading-reloading loop in which $W_m \gg m$, the initial slope of the reloading curve is reduced by the factor

$1 - \frac{1}{r}$. r must therefore be greater than 1.

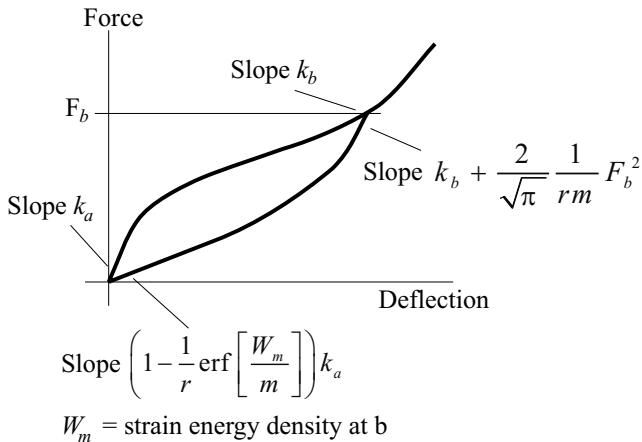


Figure 3.8-5: Dependence of reloading curve on Mullins effect material constants

It can also be shown that the dissipation of a loading-unloading cycle, as shown in Fig 3.8-6, can be written as

$$\phi = \int_A^E \sigma de = \frac{m}{r} \left[\frac{W_m}{m} \operatorname{erf}\left(\frac{W_m}{m}\right) - \frac{1}{\sqrt{\pi}} \left(1 - \exp\left(-\left(\frac{W_m}{m}\right)^2\right) \right) \right] \quad (3.8-71)$$

where

$$W_m = \int_A^C \sigma de \quad (3.8-72)$$

$_0\sigma$ is the engineering stress, e is the engineering strain. Therefore, given ϕ and W_m from two loading-unloading cycles of different amplitude, m and r can be computed.

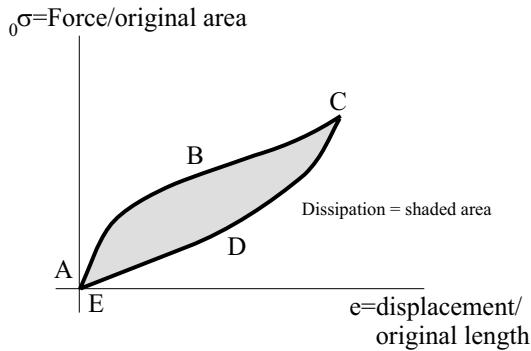


Figure 3.8-6: Dissipation in Mullins effect in a loading-reloading cycle

Heat generation: A user-specified fraction of the energy dissipated by the Mullins effect model can be considered as heat generation. This heat generation can cause heating in a TMC (thermo-mechanical-coupling) analysis.

Specification of input: To add Mullins effects to the rubber-like material model, you need to define the Mullins data using a data set of type rubber-Mullins, then specify the number of the rubber-Mullins data set in the rubber-like material model.

The rubber-Mullins data set includes:

The material constants r and m .

The heat generation factor (fraction of dissipation considered as heat generation). The default value is 0.

3.8.4 Orthotropic effect

A 2D network of orthotropic fibers can be included in the rubber-like material descriptions, see Figure 3.8-7. The directions of the fibers are described by the normal vectors \mathbf{n}_a , \mathbf{n}_b , with components $(n_a)_i$ and $(n_b)_i$ respectively. These normal vectors are defined in the undeformed configuration, and can change directions due to material deformations.

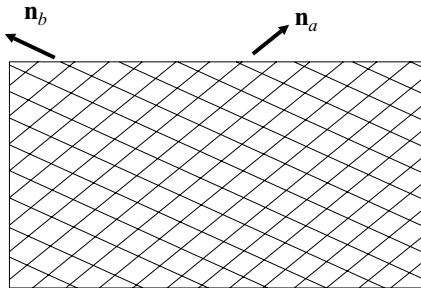


Figure 3.8-7: Network of orthotropic fibers

The orthotropic fibers are included by appending to the strain energy density the following additional term

$$W_o = \frac{k_1}{2k_2} \left\{ \exp \left[k_2 (J_4 - 1)^2 \right] + \exp \left[k_2 (J_6 - 1)^2 \right] - 2 \right\} \quad (3.8-73)$$

where $J_4 = I_4 I_3^{-1/3}$, $J_6 = I_6 I_3^{-1/3}$, $I_4 = C_{ij} (n_a)_i (n_a)_j$, $I_6 = C_{ij} (n_b)_i (n_b)_j$, C_{ij} is the Cauchy-Green deformation tensor, and k_1, k_2 are material constants.

This material description is described in the following reference:

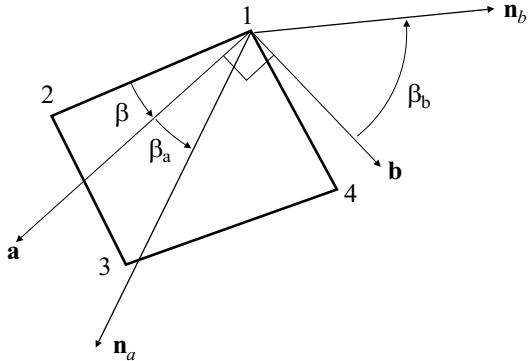
G. A. Holzapfel, T. C. Gasser, R. W. Ogden, "A New Constitutive Framework for Arterial Wall Mechanics and a Comparative Study of Material Models", *Journal of Elasticity*, 61: 1-48, 2000.

We note:

The normal directions \mathbf{n}_a and \mathbf{n}_b are defined as shown in Figure 3.8-8. In 2D analysis, the figures show how \mathbf{n}_a and \mathbf{n}_b are defined in terms of the input material axis angle β and the offset angles β_a, β_b . \mathbf{n}_a and \mathbf{n}_b lie in the plane of the element. In axisymmetric analysis, it is also allowed to set \mathbf{n}_b to the hoop (x) direction. In 3D analysis, the figure shows how \mathbf{n}_a and \mathbf{n}_b are

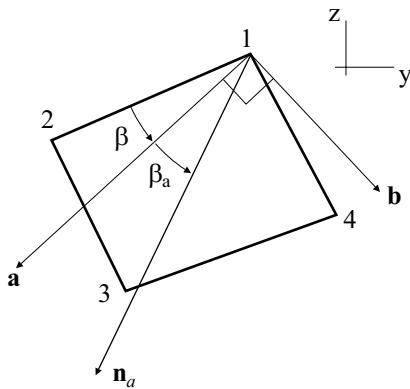
defined in terms of the material directions \mathbf{a} and \mathbf{b} . \mathbf{n}_a and \mathbf{n}_b lie in the plane defined by \mathbf{a} and \mathbf{b} .

\mathbf{n}_a and \mathbf{n}_b need not be perpendicular to each other.



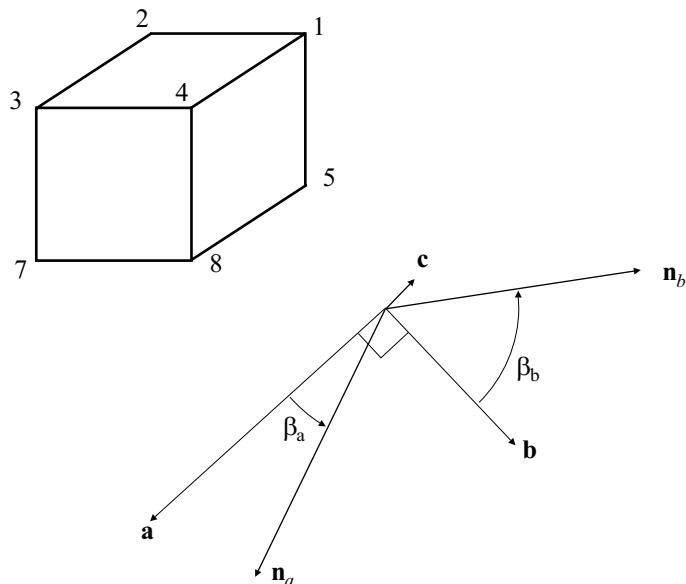
a) 2D element, hoop option not used

Figure 3.8-8: Orientation of orthotropic axes



b) 2D element, hoop option used,
 $\mathbf{n}_b = \mathbf{x}$

Figure 3.8-8: (continued)



c) 3D element, \mathbf{n}_a , \mathbf{n}_b lie in the **a**-**b** plane

Figure 3.8-8: (continued)

I_4 is interpreted as the square of the stretch of the fiber initially orientated in direction \mathbf{n}_a ; I_6 is interpreted as the square of the stretch of the fiber initially orientated in direction \mathbf{n}_b ; see Ref KJB, Example 6.7.

J_4 is interpreted as the deviatoric part of I_4 ; J_6 is interpreted as the deviatoric part of I_6 . It is assumed that the material is nearly incompressible.

When $J_4 < 1$, the material is in compression in direction \mathbf{n}_a ; when $J_6 < 1$, the material is in compression in direction \mathbf{n}_b . It is possible to remove the orthotropic effect when the material is in compression in either of the fiber directions.

Specification of input: To add orthotropic effects to the rubber-like material model, you need to define the orthotropic data using a data set of type rubber-orthotropic, then include the number of the rubber-orthotropic data set in the rubber-like material model

description.

The rubber-orthotropic data set includes:

- ▶ The material offset angles β_a , β_b (default values are 0).
- ▶ The material constants k_1 , k_2 .
- ▶ A flag indicating whether the orthotropic effect is removed when the material is in compression in either of the fiber directions (the default is to remove the orthotropic effect when the material is in compression).
- ▶ A flag indicating whether direction \mathbf{n}_b is the x (hoop) direction (the default is that \mathbf{n}_b is not the x (hoop) direction).

3.8.5 Thermal strain effect

When the material is temperature-dependent, you can include a coefficient of thermal expansion. The coefficient of thermal expansion is $\alpha(\theta)$, where θ is the temperature. Notice that the coefficient of thermal expansion is temperature dependent.

The thermal strain is calculated as

$$e_{th} = \alpha(\theta)(\theta - \theta_{REF}) - \alpha(^0\theta)(^0\theta - \theta_{REF}) \quad (3.8-74)$$

where $^0\theta$ is the initial temperature and θ_{REF} is the material reference temperature. This is the same formula as is used for the other thermo-elastic materials, see Section 3.5.

The thermal strain is assumed to be isotropic.

When the thermal strain is non-zero, the deformation gradient \mathbf{X} is assumed to be decomposed into a thermal deformation gradient \mathbf{X}_{th} and a mechanical deformation gradient \mathbf{X}_m , using

$$\mathbf{X} = \mathbf{X}_m \mathbf{X}_{th} \quad (3.8-75)$$

The thermal deformation gradient is

$$\mathbf{X}_{th} = (1 + e_{th}) \mathbf{I} \quad (3.8-76)$$

therefore the mechanical deformation gradient is

$$\mathbf{X}_m = (1 + e_{th})^{-1} \mathbf{X} \quad (3.8-77)$$

the mechanical Cauchy-Green deformation tensor is

$$\mathbf{C}_m = (1 + e_{th})^{-2} \mathbf{C} \quad (3.8-78)$$

and the mechanical Green-Lagrange strain tensor is

$$\boldsymbol{\epsilon}_m = (1 + e_{th})^{-2} \boldsymbol{\epsilon} - \frac{1}{2} (1 - (1 + e_{th})^{-2}) \mathbf{I} \quad (3.8-79)$$

For small thermal strains, (3.8-79) reduces to $\boldsymbol{\epsilon}_m \approx \boldsymbol{\epsilon} - e_{th} \mathbf{I}$, so that the strains are nearly the sum of the mechanical and thermal strains, as in small strain analysis. However, we do not assume that the thermal strains are small.

The strain energy densities are computed using the mechanical deformations. This is done by computing all invariants and stretches using the mechanical deformations, e.g. the mechanical Cauchy-Green deformation tensor.

The 2nd Piola-Kirchhoff stresses are obtained by differentiating the strain energy density with respect to the total strains. Since the strain energy density is a function of the mechanical strains, we obtain

$$\begin{aligned} S_{ij} &= \frac{1}{2} \left(\frac{\partial W}{\partial (\boldsymbol{\epsilon})_{ij}} + \frac{\partial W}{\partial (\boldsymbol{\epsilon})_{ji}} \right) \\ &= \frac{1}{2} \left(\frac{\partial W}{\partial (\boldsymbol{\epsilon}_m)_{ab}} \frac{\partial (\boldsymbol{\epsilon}_m)_{ab}}{\partial (\boldsymbol{\epsilon})_{ij}} + \frac{\partial W}{\partial (\boldsymbol{\epsilon}_m)_{ba}} \frac{\partial (\boldsymbol{\epsilon}_m)_{ba}}{\partial (\boldsymbol{\epsilon})_{ji}} \right) \\ &= ((1 + e_{th})^{-2}) \frac{1}{2} \left(\frac{\partial W}{\partial (\boldsymbol{\epsilon}_m)_{ij}} + \frac{\partial W}{\partial (\boldsymbol{\epsilon}_m)_{ji}} \right) \end{aligned} \quad (3.8-80)$$

With this definition, the 2nd Piola-Kirchhoff stresses are conjugate to the Green-Lagrange strains.

3.8.6 TRS material

The rubber-like materials can include the TRS (thermorheologically simple) material assumption. In this assumption, the material properties are independent of temperature, but the viscoelastic time constants are functions of temperature as described in the viscoelastic material effect (time-temperature superposition).

A TRS material requires a rubber-table of type TRS. The number of the rubber-table data set is specified as part of the rubber-like material description.

The rubber-table of type TRS data set is a table with a different temperature for each row of the table. Each row of the table contains the coefficient of thermal expansion (enter 0 if there are no thermal effects).

3.8.7 Temperature-dependent material

The rubber-like materials can be fully temperature-dependent.

Specification of input: The input of the material properties for a fully temperature-dependent material is done using a rubber-table. The rubber-table has different forms depending upon whether the Mooney-Rivlin, Ogden, Arruda-Boyce or hyper-foam relationships are used, hence there is a rubber-table of type Mooney-Rivlin, a rubber-table of type Ogden, etc. For brevity, we discuss only the case of a Mooney-Rivlin material.

A Mooney-Rivlin material that is fully temperature-dependent requires a rubber-table of type Mooney-Rivlin. The number of the rubber-table data set is specified as part of the Mooney-Rivlin material description.

The rubber-table of type Mooney-Rivlin data set is a table with a different temperature for each row of the table. Each row of the table contains

- ▶ The coefficient of thermal expansion.
- ▶ The material properties (C_1 to C_9 , constants D_1 to D_2 and the bulk modulus κ).

- ▶ The curve-fitting number (0 if curve-fitting is not used).
- ▶ The rubber-viscoelastic data set number, if the model includes viscoelastic effects.
- ▶ The rubber-Mullins data set number, if the model includes Mullins effects.
- ▶ The rubber-orthotropic data set number, if the model includes orthotropic effects.

You can specify a different curve-fitting number for each temperature.

You can specify a different rubber-viscoelastic data set number, rubber-Mullins data set number or rubber-orthotropic data set number for each temperature. In this way, the material constants for the viscoelastic, Mullins or orthotropic effects can be temperature-dependent. However, if one row of the rubber-table has viscoelastic data, all rows must have viscoelastic data. Similarly, if one row of the rubber-table has Mullins or orthotropic data, all rows must have this data.

Solution process: The strain energy density is computed by interpolation from the strain energy densities of the bounding temperatures as follows: Let θ be the current temperature, θ_1 be the temperature in the rubber-table just below the current temperature and θ_2 be the temperature in the rubber-table just above the current temperature. Then the program computes the strain energy densities W_1 and W_2 , and the derivatives of the strain energy densities, using the material properties for θ_1 and θ_2 respectively. Finally the program computes the strain energy density and its derivatives using linear interpolation, for example

$$W = \frac{\theta_2 - \theta}{\theta_2 - \theta_1} W_1 + \frac{\theta - \theta_1}{\theta_2 - \theta_1} W_2 \quad (3.8-81)$$

3.9 Geotechnical material models

3.9.1 Curve description material model

- The curve description model can be employed with the **2-D solid** (plane strain and axisymmetric) and **3-D solid** elements.
- The curve description model can be used with the **small displacement** and **large displacement** formulations. In all cases, small strains are assumed.

When used with the small displacement formulation, a materially-nonlinear-only formulation is employed, and when used with the large displacement formulation, a TL formulation is employed.

- The curve description model is a simple incremental stress-strain law used to represent the response of geological materials. The model describes the instantaneous bulk and shear moduli as piecewise linear functions of the current volume strain, as shown in Fig. 3.9-1. An explicit yield condition is not used and whether the material is loading or unloading is determined by the history of the volume strain only.

- To present the governing constitutive relations, let

$${}^t e_{ij} = \text{total strains (the left superscript "t" always refers to time } t)$$

$$e_{ij} = \text{incremental strains}$$

$${}^t e_m = \sum_i \frac{{}^t e_{ii}}{3} = \text{mean strain (negative in compression)}$$

$$e_m = \sum_i \frac{e_{ii}}{3} = \text{incremental mean strain}$$

$${}^t g_{ij} = {}^t e_{ij} - {}^t e_m \delta_{ij} = \text{deviatoric strains}$$

$$g_{ij} = \text{incremental deviatoric strains}$$

$${}^t \sigma_{ij} = \text{total stresses (negative in compression)}$$

$$\sigma_{ij} = \text{incremental stresses}$$

$${}^t\sigma_m = \sum_i \frac{{}^t\sigma_{ii}}{3} = \text{mean stress}$$

$$\sigma_m = \sum_i \frac{\sigma_{ii}}{3} = \text{incremental mean stress}$$

p_{min} = minimum mean stress ever reached

$${}^t s_{ij} = {}^t\sigma_{ij} - {}^t\sigma_m \delta_{ij} = \text{deviatoric stresses}$$

s_{ij} = incremental deviatoric stresses

$'G, {}^tK$ = shear and bulk moduli

The incremental stress-strain relations using the curve description model are then

$$s_{ij} = 2 {}^tG g_{ij}$$

$$\sigma_m = 3 {}^tK e_m$$

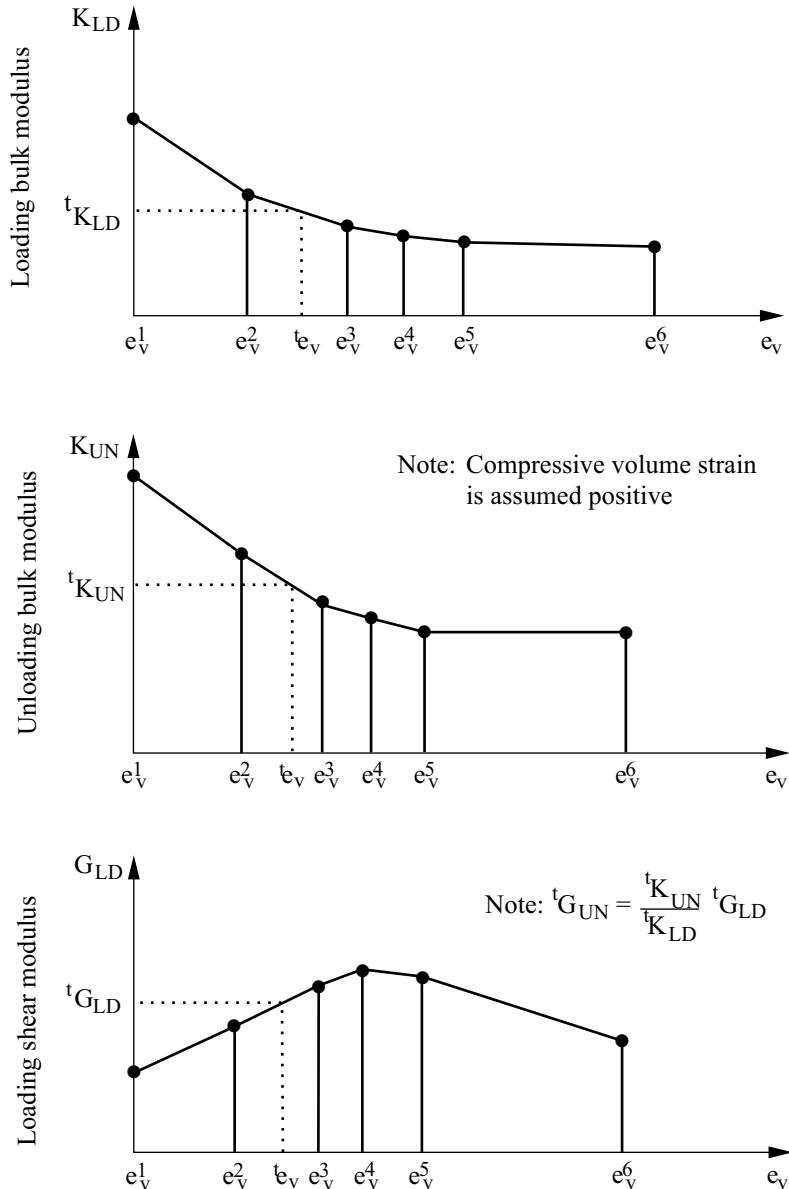


Figure 3.9-1: Input of the curve description model

- The instantaneous bulk and shear moduli, $'K$ and $'G$, are functions of the loading condition, and the volumetric strain $'e_v$ is defined as

$$'e_v = e_{grav} + (-3'e_m)$$

where e_{grav} is the volumetric strain (three times the mean strain and taken positive in compression) due to the gravity pressure and $'e_m$ is the mean strain at time t . Defining e_{min} as the minimum mean strain ever reached during the solution, we have that the material is loading if $'e_m \leq e_{min}$ and the material is unloading if $'e_m > e_{min}$ i.e.,

$$'K = \begin{cases} 'K_{LD} & \text{when } 'e_m \leq e_{min} \\ 'K_{UN} & \text{when } 'e_m > e_{min} \end{cases}$$

and

$$'G = \begin{cases} 'G_{LD} & \text{when } 'e_m \leq e_{min} \\ 'G_{UN} & \text{when } 'e_m > e_{min} \end{cases}$$

- Note that the loading conditions for both the bulk and the shear moduli are determined by the history of $'e_m$ only. The values of $'K_{LD}$, $'K_{UN}$, and $'G_{LD}$ are obtained using the curves in Fig. 3.9-1,

and the modulus $'G_{UN} = 'G_{LD} \left(\frac{'K_{UN}}{'K_{LD}} \right)$.

- The incremental solution at time t is obtained using the equations

$$^{t+\Delta t}S_{ij} = 2 'G \left(^{t+\Delta t}g_{ij} - {}^t g_{ij} \right) + {}^t S_{ij}$$

and

$$^{t+\Delta t}\sigma_{ij} = 3 'K \left(^{t+\Delta t}e_{ij} - {}^t e_{ij} \right) + {}^t \sigma_{ij}$$

To obtain $'G$ and $'K$, we check whether the loading or unloading conditions are active by comparing the current volumetric strains and previous volumetric strains. To start the procedure at time 0,

loading conditions are assumed. It should be noted that the stresses at time $t + \Delta t$ are calculated using the material moduli pertaining to time t .

- An important additional analysis option is that the material can weaken under loading conditions if tensile stresses exceed preassigned values. Since the curve description model has been developed primarily for the analysis of geological materials, the material weakening is assumed to occur once the principal tensile stresses due to the applied loading exceed the in-situ gravity pressure p (taken positive). The gravity pressure p is calculated as

$$p = \sum_{i=1}^N h_i p_i \text{ where } h_i \text{ are the element interpolation functions and}$$

p_i is the pressure at the element nodes. The nodal pressure p_i is calculated as $p_i = \gamma \cdot Z_i$ where γ is the material density and Z_i the nodal Z coordinate, which coincides with the vertical direction. The material weakening can be included using a *tension cut-off* model or a *cracking* model.

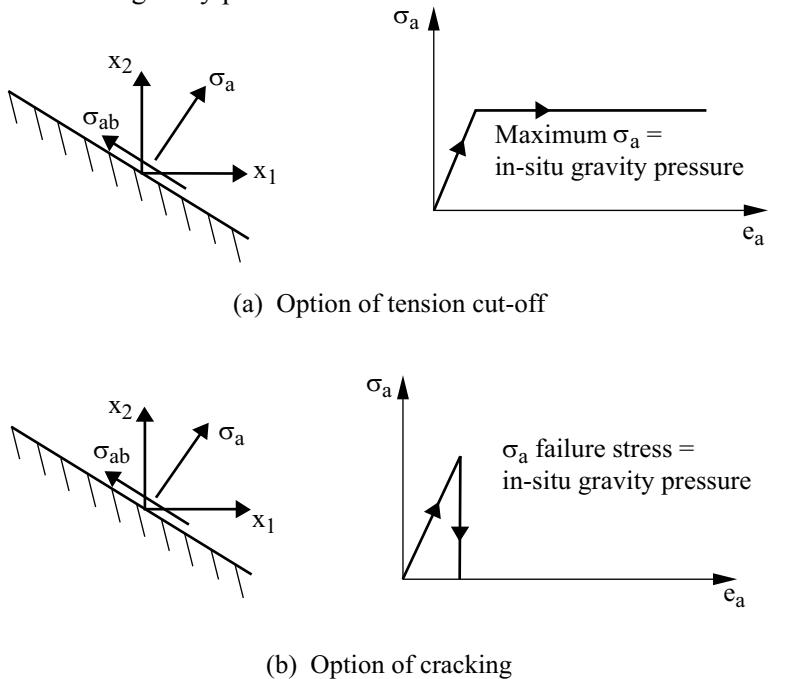
- ▶ In both modes of behavior (tension cut-off and cracking) the principal stresses due to the external loading are calculated and compared with the in-situ gravity pressure. Once the principal tensile stress is equal to the in-situ gravity pressure, the material is treated as being orthotropic, with the modulus corresponding to the direction of the principal tensile stress being multiplied by a stiffness reduction factor (an input parameter). Another factor, also an input parameter, is applied to reduce the shear stiffness.
- ▶ In the tension cut-off mode, the normal and shear stiffnesses corresponding to the direction of the maximum principal tensile stress are reduced, but the stresses are fully retained. The model therefore simulates elastic-plastic flow of the material.

On the other hand, in the cracking mode the stiffnesses are reduced, and in addition the normal and shear stresses due to external loads and corresponding to the direction of stiffness reduction are released. The reduction of stiffness together with the stress release models a crack, i.e., a tensile failure plane, that forms at right angles to the direction of the maximum principal tensile stress.

- In subsequent load steps, tension cut-off or cracking only in the direction already determined and/or other(s) perpendicular to it may be active, i.e., no change in the tension cut-off or cracking direction is considered. The tension cut-off or cracking mode is considered to be inactive provided the strain normal to the crack (in the direction of tension cut-off) becomes both negative and less than the strain at which the failure occurred initially; otherwise it remains active.

Fig. 3.9-2 illustrates the tension cut-off and cracking options.

- Note that the Z direction should be the vertical direction and that the ground level is assumed to be at $Z = 0$ in the calculation of the in-situ gravity pressure.



ADINA checks for tension cut-off or cracking at each integration point.

Figure 3.9-2: Tension cut off and cracking options

- The in-situ gravity pressure is evaluated using the input for material density which corresponds to the weight per unit volume.

This is different than the mass density which is used in the element mass matrix calculation.

3.9.2 Drucker-Prager material model

- The Drucker-Prager model is based on
 - ▶ The Drucker-Prager yield condition (see p. 604, ref. KJB)
 - ▶ A non-associated flow rule using the Drucker-Prager and cap yield functions
 - ▶ A perfectly-plastic Drucker-Prager yield behavior
 - ▶ Tension cut-off
 - ▶ Cap hardening
- The Drucker-Prager model can be used with the **2-D solid** and **3-D solid** elements.
- The Drucker-Prager model can be used with the **small displacement/small strain**, **large displacement/small strain** and **large displacement/large strain** formulations.
When used with the small displacement/small strain formulation, a materially-nonlinear-only formulation is employed, when used with the large displacement/small strain formulation, a TL formulation is employed and when used with the large displacement/large strain formulation, the ULH formulation is employed.
- Fig. 3.9-3 summarizes some important features of the Drucker-Prager model (with tension cut-off and cap hardening).
The Drucker-Prager yield function is given by:

$${}^t f_{DP} = \alpha {}^t J_1 + \sqrt{{}^t J_{2D}} - k$$

where α and k are material parameters and are functions of frictional angle ϕ and cohesion coefficient c .

The corresponding potential function is given by

$${}^t g_{DP} = \beta {}^t J_1 + \sqrt{{}^t J_{2D}} - u$$

where β is a function of dilatation angle ψ , u is not required as input parameter.

The parameters α , k and β can be determined by matching the Drucker-Prager criterion with the Mohr-Coulomb criterion.

For triaxial compression test:

$$\alpha = \frac{2 \sin \phi}{\sqrt{3}(3 - \sin \phi)} \quad k = \frac{6c \cos \phi}{\sqrt{3}(3 - \sin \phi)} \quad \beta = \frac{2 \sin \psi}{\sqrt{3}(3 - \sin \psi)}$$

For triaxial tensile test:

$$\alpha = \frac{2 \sin \phi}{\sqrt{3}(3 + \sin \phi)} \quad k = \frac{6c \cos \phi}{\sqrt{3}(3 + \sin \phi)} \quad \beta = \frac{2 \sin \psi}{\sqrt{3}(3 + \sin \psi)}$$

For the plane strain test:

$$\alpha = \frac{\tan \phi}{\sqrt{9 + 12 \tan^2 \phi}} \quad k = \frac{3c}{\sqrt{9 + 12 \tan^2 \phi}}$$

The cap yield function depends on the shape of the cap. For a plane cap:

$${}^t f_C = -{}^t J_1 + {}^t J_1^a$$

where ${}^t J_1^a$ is a function of the volumetric plastic strain ${}^t e_V^P$:

$${}^t J_1^a = -\frac{1}{D} \ln \left(1 - \frac{{}^t e_V^P}{W} \right) + {}^0 J_1^a$$

where ${}^0 J_1^a$ is the cap initial position and W and D are material constants.

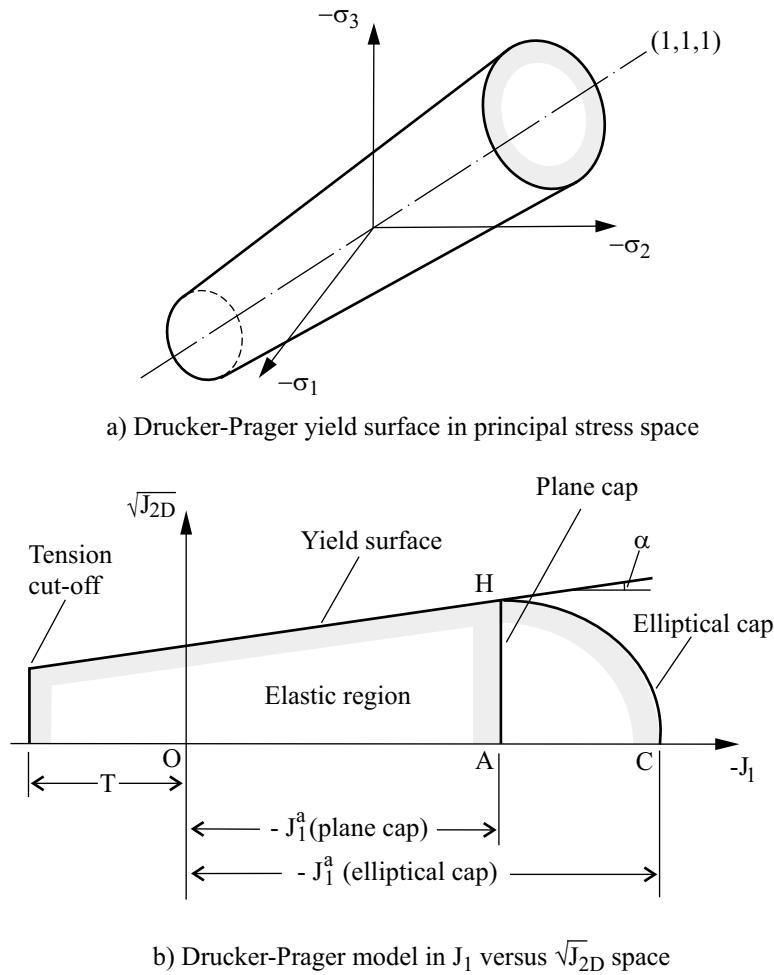


Figure 3.9-3: Drucker-Prager model

For an elliptical cap:

$${}^t f_C = \left({}^t J_1 + {}^t L \right)^2 + R^2 \left({}^t J_{2D} - {}^t B^2 \right)$$

where ${}^t B$ is the vertical semi-axis of the ellipse (AH), R is the cap ratio (AC/AH) and ${}^t L$ is equal to OA. The dependence of ${}^0 J_1^a$ on the volumetric plastic strain is as for the plane cap. In the case of tension cut-off, T is the maximum value that ${}^t J_1^a$ can take.

- We note that:
 - ▶ Unlike for the von Mises model, the Drucker-Prager yield function cannot be used with hardening (the material is always assumed to be elastic perfectly-plastic, except for cap hardening).
 - ▶ If $\alpha(\phi)$ approaches zero (minimum value of $\alpha(\phi)$ is taken to be 10^{-5}), the initial position of the cap ${}^0J_1^a$ is moved far to the right in Figure 3.9-3 and is not reached in the analysis, and the position of the tension cut-off T is moved far to the left and is also not reached in the analysis, then the Drucker-Prager yield condition approaches the von Mises elastic-perfectly-plastic yield condition.
 - ▶ In the case of Drucker-Prager yielding, the dilatancy (volume expansion of the material in shear) is governed by the magnitude of $\alpha(\psi)$ (for the von Mises yield condition $\alpha(\psi) = 0$ and there is no dilatancy).
 - ▶ Cap yielding leads to an increase of compressive plastic volumetric strain. If the plane cap is used, there is no change of deviatoric plastic strains, while if the elliptical cap is used, the deviatoric components of the plastic strain change during the cap yielding. Fig. 3.9-4 shows the relation between the cap position ${}^tJ_1^a$ and the volumetric strain

$${}^t\epsilon_V^P = {}^t\epsilon_{11}^P + {}^t\epsilon_{22}^P + {}^t\epsilon_{33}^P$$

that is, the constraint that exists between these two quantities. We note that the cap movement corresponds to a hardening and that the volumetric plastic strain is constrained to be smaller than the input parameter W in absolute value.

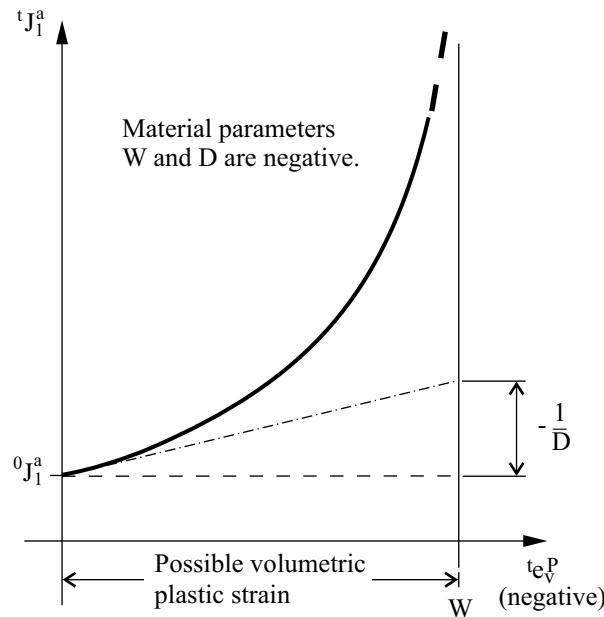


Figure 3.9-4: Relation between cap position tJ_1^a and volumetric plastic strain te_v^P

- ▶ In the case of vertex yielding (the stress state is represented by point H in Fig. 3.9-3), the plastic deformation corresponds to the Drucker-Prager and cap yielding. The vertex yielding leads to changes in volumetric and deviatoric plastic strains with the cap hardening behavior.
- ▶ Stress states beyond the tension cut-off value T (input to ADINA as a positive value) are not possible. When this limit is reached or exceeded, the program sets all the shear stress components equal to zero and the normal stress components all equal to $\frac{T}{3}$. The elastic constitutive law is used for the stiffness matrix.

3.9.3 Cam-clay material model

- The Cam-clay material model is a pressure-dependent plasticity model. It is based on

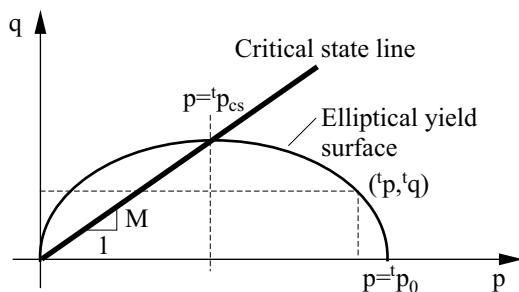
- ▶ An associated flow rule using an elliptical yield function
- ▶ The critical state line, which controls the failure of the material
- ▶ The consolidation behavior of clayey materials
- The Cam-clay model can be used with the **2-D solid** and **3-D solid** elements.
- The Cam-clay model can be used with the **small displacement/small strain**, **large displacement/small strain** and **large displacement/large strain** formulations.

When used with the small displacement formulation, a materially-nonlinear-only formulation is employed and when used with the large displacement/small strain formulation, a TL formulation is employed and when used with the large displacement/large strain formulation, a ULH formulation is employed.
- The Cam-clay model is able to simulate the following mechanical behaviors for clayey materials, which are confirmed by lab tests and in-situ tests:
 - ▶ Strain hardening and softening under normal consolidation states or overconsolidation states
 - ▶ Nonlinear dependence of the elastic volumetric strain on the hydrostatic pressure
 - ▶ An ultimate condition of perfect plasticity at which plastic shearing can continue indefinitely without changes in volume or effective stress
- The Cam-clay yield function is given by

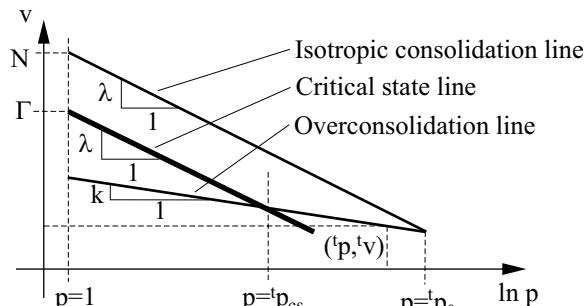
$${}^t F = \frac{{}^t q^2}{M^2} + {}^t p \left({}^t p - {}^t p_0 \right) = 0$$

where the mean stress ${}^t p$ and the distortional stress ${}^t q$ at time t are related to the first stress invariant ${}^t I_1$ and the second deviatoric stress invariant ${}^t J_2$ by ${}^t p = \frac{{}^t I_1}{3}$ and ${}^t q = \sqrt{3 {}^t J_2}$. M , a material constant, is the slope of the critical state line and ${}^t p_0$, called the pre-consolidation pressure, is the diameter of the ellipsoid at time t along the axis p . All of the variables are shown in Fig. 3.9-5(a).

- The hardening rule is written as



a) Cam-clay yield surface



b) Compression behavior

Figure 3.9-5: Cam-clay model

$${}^t v = N - \lambda \ln {}^t p_0 + k \ln \frac{{}^t p_0}{{}^t p}$$

or, in rate form,

$${}^t\dot{v}^P = -(\lambda - k) \ln \frac{{}^t p_0}{{}^t p}$$

where ${}^t v$ is the specific volume at time t and ${}^t \dot{v}^P$ is the plastic specific volume rate at time t . λ , k and N are material constants, in which λ is the slope of the isotropic consolidation line, k is the slope of the overconsolidation line and N is the specific volume at the isotropic consolidation state when ${}^t p$ is 1.0. N is related to Γ , the specific volume at the critical state when ${}^t p$ is 1.0, by

$$N = \Gamma + (\lambda - k) \ln 2$$

- The effective bulk modulus at time t can be expressed as

$${}^t K = \frac{{}^t v {}^t p}{k}$$

The corresponding shear modulus at time t is obtained using

$${}^t G = \frac{3(1-2\nu)}{2(1+\nu)} {}^t K, \text{ in which } \nu \text{ is the Poisson's ratio (constant}$$

throughout the analysis).

- In an analysis using the Cam-clay model, ADINA requires either the initial stresses or an initial stiffness to be defined.

Initial stresses: The initial stresses are either directly input or elastically calculated before or at the first load step for any specified element group. They can be tiny isotropic compressive stresses (such as 5% of the gravity-induced stresses), or can be the full gravity-induced stresses.

When the analysis starts with initial stresses, the initial size of the yield surface can either be directly input or computed from the initial stresses. If the initial stresses are induced by the full gravity load, the initial size of the yield surface (pre-

consolidation pressure) would be estimated by employing the input parameters OCR and KNULL, unless the initial size of the yield surface is specified in the Cam-clay model definition.

- If the initial stresses are not directly input, ADINA calculates the initial stresses as the stresses computed at load step 1. Then,
- (a) The results for load step 1 are calculated using the input elastic material properties E and ν .
 - (b) The results for the remaining load steps depend upon the solution for load step 1.

When the analysis starts with initial stresses, ADINA takes the true initial size of the yield surface as the greater of the specified initial size and the computed initial size.

Initial stiffness: The analysis can be performed with an initial stiffness, which is calculated using the initial Young's modulus and Poisson's ratio. In this case, the initial size of the yield surface must be directly specified.

- Parameter KNULL is defined as the coefficient of earth pressure at the normal consolidation state. Usually it is different than the coefficient of current earth pressure. KNULL is approximated by $K_0 = 1 - \sin \phi$ where ϕ is the internal friction angle of the soil.
- The material constant M is computed based on ϕ . If triaxial compression tests are performed, M is given by $M = \frac{6 \sin \phi}{3 - \sin \phi}$. If triaxial extension tests are performed, M is given by $M = \frac{6 \sin \phi}{3 + \sin \phi}$.
- For more information, see references:
 - ref. D.M. Wood, *Soil behavior and critical state soil mechanics*, Cambridge University Press, pp. 84-225, 1990.
 - ref. A.M. Britto and M.J. Gunn, *Critical state soil mechanics via finite elements*, Ellis Horwood Limited, pp. 161-184, 1987.

3.9.4 Mohr-Coulomb material model

- The Mohr-Coulomb model is based on
 - ▶ A non-associated flow rule
 - ▶ A perfectly-plastic Mohr-Coulomb yield behavior
 - ▶ Tension cut-off
- The Mohr-Coulomb model can be used with the **2-D solid** and **3-D solid** elements.
- The Mohr-Coulomb model can be used with the **small displacement/small strain, large displacement/small strain** and **large displacement/large strain** formulations.

When used with the small displacement formulation, a materially-nonlinear-only formulation is employed and when used with the large displacement/small strain formulation, a TL formulation is employed and when used with the large displacement/large strain formulation, a ULH formulation is employed.

- The Mohr-Coulomb yield function is given by

$$'f_{MC} = 'I_1 \sin \phi + \frac{1}{2} \left(3(1 - \sin \phi) \sin 't\theta + \sqrt{3}(3 + \sin \phi) \cos 't\theta \right) \sqrt{'J_2} - 3C \cos \phi$$

and the corresponding potential function is written as

$$'g_{MC} = 'I_1 \sin \psi + \frac{1}{2} \left(3(1 - \sin \psi) \sin 't\theta + \sqrt{3}(3 + \sin \psi) \cos 't\theta \right) \sqrt{'J_2} - 3C \cos \psi$$

where

$$'t\theta = \frac{1}{3} \cos^{-1} \left(\frac{3\sqrt{3}}{2} \frac{'J_3}{'J_2^{3/2}} \right)$$

ϕ is the friction angle (a material constant), C is the cohesion (a material constant), ψ is the dilatation angle (a material constant), $'I_1$ is the first stress invariant at time t , $'J_2$ is the second deviatoric stress invariant at time t and $'J_3$ is the third deviatoric stress invariant at time t .

- Some important features of the Mohr-Coulomb model are demonstrated in Figures 3.9-6 to 3.9-9.
- It is noted that:
 - ▶ A hardening rule does not apply to the Mohr-Coulomb model; the Mohr-Coulomb model is only used in conjunction with the elastic-perfectly-plastic yield condition.

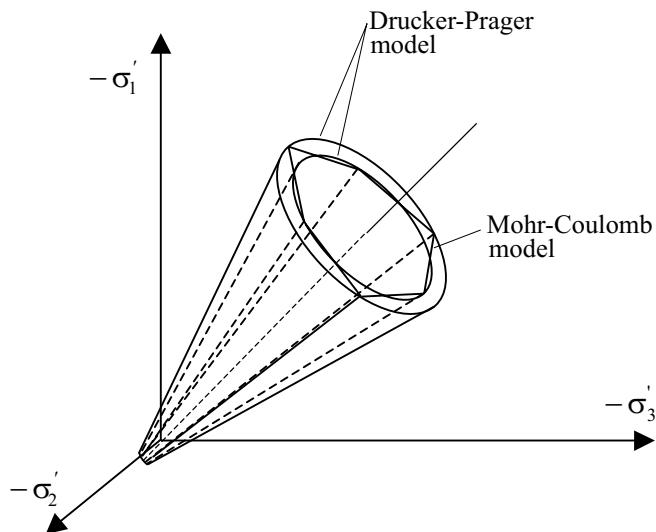


Figure 3.9-6: The yield surfaces of the Mohr-Coulomb model and the Drucker-Prager model

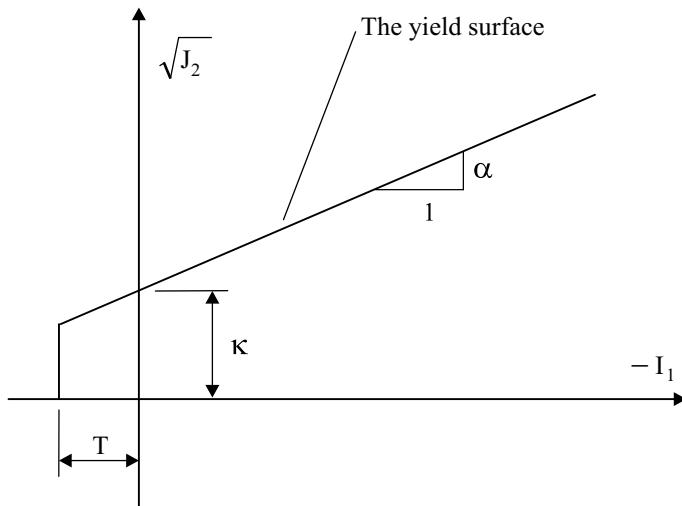


Figure 3.9-7: The Mohr-Coulomb model in I_1 vs $\sqrt{J_2}$ space

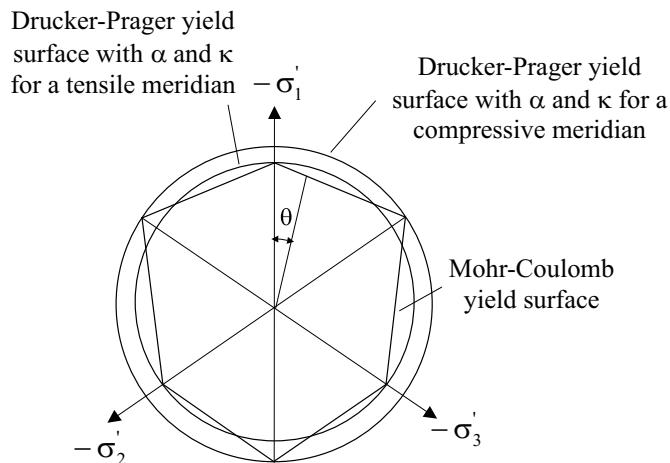


Figure 3.9-8: Shapes of the Mohr-Coulomb and the Drucker-Prager yield criteria on the π plane

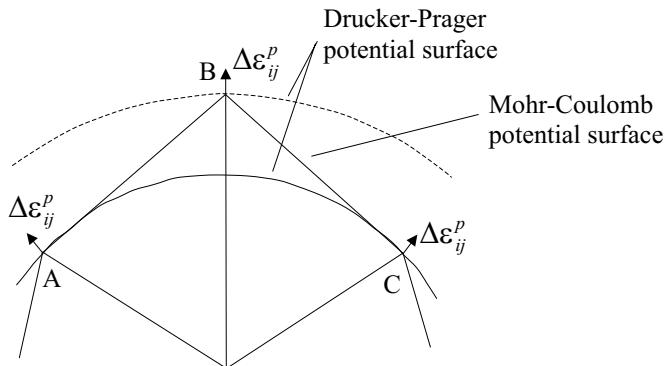


Figure 3.9-9: Corner treatment of the Mohr-Coulomb model

- ▶ When using the Mohr-Coulomb model, the volume dilatancy due to shearing is only governed by the dilatancy angle ψ . In order to reduce the pathologically large material dilatation, one usually specifies the dilatation angle smaller than the friction angle.
- ▶ In the case of vertex yielding (the stress states represented by points A, B and C in Figure 3.9-9), the plastic strain increments are obtained by applying the flow rule to the Drucker-Prager equation. In other words, the flow rule is taken to be identical with the projection of the axis of either the tensile or the compressive meridian on the deviatoric plane.
- ▶ Stress states beyond the tension cut-off value T are not possible. When this limit is reached or exceeded, the Maximum-Tensile-Stress Criterion (Rankine) is applied by shifting the hydrostatic stress component to the hydrostatic pressure corresponding to the tension cut-off. The tension cut-off value T is three times as large as the tensile strength of the material.
- ▶ The Mohr-Coulomb yield equation can be written as

$${}^t f_{MC} = {}^t \alpha {}^t I_1 + \sqrt{{}^t J_2} - {}^t k$$

Here $'\alpha$ and $'k$ are stress dependent:

$$' \alpha = \frac{2 \sin \phi}{3(1 - \sin \phi) \sin ' \theta + \sqrt{3}(3 + \sin \phi) \cos ' \theta}$$

$$' k = \frac{6C \sin \phi}{3(1 - \sin \phi) \sin ' \theta + \sqrt{3}(3 + \sin \phi) \cos ' \theta}$$

If stress states fall on a compressive meridian ($'\theta = \pi/3$), then

$$' \alpha = \frac{2 \sin \phi}{\sqrt{3}(3 - \sin \phi)}, \quad ' k = \frac{6C \cos \phi}{\sqrt{3}(3 - \sin \phi)}$$

If stress states fall on a tensile meridian ($'\theta = 0$), then

$$' \alpha = \frac{2 \sin \phi}{\sqrt{3}(3 + \sin \phi)}, \quad ' k = \frac{6C \cos \phi}{\sqrt{3}(3 + \sin \phi)}$$

- For more information, see the following reference:

ref. W.F. Chen, *Nonlinear analysis in soil mechanics*, Elsevier, pp. 139-370, 1990.

3.10 Fabric material model with wrinkling

- The elastic-orthotropic material model (discussed in Section 3.2) can be used with the **2-D solid element** (3-D plane stress option) and the **large displacement/small strain formulation** to model the behavior of fabric structures.
- Fabric materials are typically very thin and flexible. They can support only in-plane tensile stresses, and will wrinkle under compressive stresses.

- The analysis of fabric structures with ADINA is performed in two steps: first, initial strains must be specified, Young's moduli must be set to a fraction (usually 10^{-4}) of their actual values, and the wrinkling of the material must not be taken into account. Then, in a second step (restart analysis), the same orthotropic material model is used except that the Young's moduli are now set to their actual values, and wrinkling is taken into account.
- For each integration point, wrinkling occurs if the smallest of the two in-plane principal stresses ($'\sigma_{p1}, '\sigma_{p2}$) becomes negative. If wrinkling occurs, then this smallest principal stress $'\sigma_{p2}$ is set to zero. If the other in-plane principal stress $'\sigma_{p1}$ is greater than zero, the program calculates a new "wrinkling stress" $'\sigma'_{p1}$ using

$$'\sigma'_{p1} = A'^t \varepsilon_{p1}$$

where $'\varepsilon_{p1}$ is the strain in the direction of $'\sigma_{p1}$ and A is calculated from the orthotropic material law, using the condition $'\sigma_{p2} = 0$. If $'\sigma_{p1}$ is smaller than zero, ADINA sets $'\sigma_{p1}$ to zero; in this case, wrinkling occurs in two directions.

The element constitutive matrix corresponding to the principal directions is also modified for the wrinkling behavior and becomes

$$\begin{bmatrix} \alpha A & 0 & 0 \\ 0 & \eta A & 0 \\ 0 & 0 & \eta A \end{bmatrix}$$

where η is a reduction factor equal to 10^{-6} , and $\alpha = 1$ if $'\sigma_{p1} \geq 0$ or $\alpha = \eta$ if $'\sigma_{p1} < 0$.

3.11 Viscoelastic material model

- The viscoelastic model can be used with the **truss**, **2-D solid**, **3-D solid** and **shell** elements.

- The viscoelastic model can be used with the **small displacement/small strain**, **large displacement/small strain** and **large displacement/large strain** formulations (2-D solid and 3-D solid elements only).

When used with the small displacement/small strain formulation, a materially-nonlinear-only formulation is employed, when used with the large displacement/small strain formulation, a TL formulation is employed and when used with the large displacement/large strain formulation, the ULH formulation is employed.

- The mechanical behavior for an isotropic and linear viscoelastic material may be expressed in tensor notation as

$$s_{ij}(t) = 2G(0)e_{ij}(t) + 2 \int_{0^+}^t e_{ij}(t-\tau) \frac{dG(\tau)}{d\tau} d\tau \quad (3.11-1)$$

$$\sigma_{kk}(t) = 3K(0)e_{kk}(t) + 3 \int_{0^+}^t \varepsilon_{kk}(t-\tau) \frac{dK(\tau)}{d\tau} d\tau \quad (3.11-2)$$

where t is the time, $s_{ij} = \sigma_{ij} - \frac{1}{3}\delta_{ij}\sigma_{kk}$ is the deviatoric stress, δ_{ij} is the Kronecker delta, σ_{ij} is the stress, $e_{ij} = \varepsilon_{ij} - \frac{1}{3}\delta_{ij}\varepsilon_{kk}$ is the deviatoric strain, ε_{ij} is the strain, $G(t)$ is the shear modulus and $K(t)$ is the bulk modulus.

In the presence of a temperature variation $T(t)$, the stresses for an isotropic and thermorheologically linear viscoelastic material may be written as

$$s_{ij}(t) = 2G(0)e_{ij}(t) + 2 \int_{0^+}^{\xi} e_{ij}(\xi-\zeta) \frac{dG(\zeta)}{d\zeta} d\zeta \quad (3.11-3)$$

$$\sigma_{kk}(t) = 3K(0)[e_{kk}(t) - 3\alpha_0\theta(t)] + 3 \int_{0^+}^{\xi} [\varepsilon_{kk}(\xi-\zeta) - 3\alpha_0\theta(\zeta)] \frac{dK(\zeta)}{d\zeta} d\zeta \quad (3.11-4)$$

where

$$\zeta = \int_{0^+}^{\tau} \psi[T(\eta)] d\eta, \quad \xi = \int_{0^+}^t \psi[T(\tau)] d\tau \quad (3.11-5)$$

the pseudo-temperature $\theta(t)$ is given by

$$\theta(t) = \frac{1}{\alpha_0} \int_{T_0}^{T_1} \alpha(T') dT', \quad \alpha_0 = \alpha(T_0) \quad (3.11-6)$$

$\alpha(t)$ is the temperature-dependent coefficient of thermal expansion and $\psi(t)$ is the shift function, which obeys

$$\psi(T_0) = 1, \quad \psi(T) > 0, \quad \frac{d\psi}{dT} > 0 \quad (3.11-7)$$

Please note that the definition of the temperature-dependent coefficient of thermal expansion $\alpha(t)$ is slightly different than the definition of ' α ' used in other material models of ADINA; see Section 3.5 for the definition of ' α '. If the coefficient of thermal expansion is independent of temperature, then both definitions are the same.

In equations (3.11-3) and (3.11-4) it is assumed that the mechanical and thermal responses are uncoupled. Furthermore if the temperature is constant, equations (3.11-3) and (3.11-4) reduce to equations (3.11-1) and (3.11-2).

- We assume the following thermo-material properties:

$$G(t) = G_\infty + \sum_{i=1}^{\eta_G} G_i e^{-\beta_i t} \quad (3.11-8)$$

$$K(t) = K_\infty + \sum_{i=1}^{\eta_K} K_i e^{-\gamma_i t} \quad (3.11-9)$$

$$T(t) \neq 0 \quad (3.11-10)$$

$$\alpha = \alpha(T) \quad (3.11-11)$$

where G_∞ and K_∞ are the long-time shear modulus and bulk modulus respectively, β_i and γ_i are the decay constants for the shear modulus and bulk modulus respectively and η_G and η_K are the number of time-dependent terms for the shear modulus and bulk modulus respectively. Equations (3.11-8) and (3.11-9) are referred to in the literature as Prony or Dirichlet series. η_G and η_K are limited to a maximum value of 15.

The shift function used is the Williams-Landell-Ferry (WLF) equation, written as follows

$$\log_{10} \psi(T) = \frac{C_1(T - T_0)}{C_2 + (T - T_0)} \quad (3.11-12)$$

in which C_1 and C_2 are material constants.

- You can either enter the Prony series constants in the AUI, or you can enter the measured data for the shear and bulk modulus and let the AUI construct the Prony series. In the latter case, the AUI constructs the Prony series using a least-squares technique, see the reference by Frutiger and Woo given below.
- The nodal point temperatures are input to ADINA as discussed in Section 5.6.
- For more information, see the following references:
 - ref. W.N. Findley, J.S. Lai and K. Onaran, *Creep and relaxation of nonlinear viscoelastic materials*, Dover Publications, 1976.
 - ref. R.L. Frutiger and T.C. Woo, "A thermoviscoelastic analysis for circular plates of thermorheologically simple material", *Journal of Thermal Stresses*, 2:45-60, 1979.

3.12 Porous media formulation

- The porous media formulation is applicable to porous structures subject to static or dynamic loading. It deals with the interaction between the porous solids and pore fluids, which flow through the porous solid skeleton.
- The porous media formulation only applies to 2D and 3D solid elements. These elements have pore pressure variables at their corner node points.
- The pore pressures are considered to be independent variables, along with the displacements. The pore pressures are output by ADINA along with the displacements, velocities, stresses and other solution results.
- The porous media model is implemented with a fully coupled formulation, as described in detail below.
- The ADINA porous media formulation is based on the following assumptions:
 - ▶ The pore fluids are incompressible or slightly compressible.
 - ▶ The porous solid skeleton is fully saturated with pore fluid.
 - ▶ The pore fluid flows through the porous solid according to Darcy's law.
- With the porous media formulation, undrained analysis (static pore fluid) and consolidation or swelling analysis (pore fluid flowing through the porous solid skeleton) can be performed.
- In the analyses of porous structures, all the nonlinear formulations in ADINA (small displacement/small strain, large displacement/small strain and large displacement/large strain) can be used. A materially-nonlinear-only formulation is employed for the small displacement/small strain formulation; the TL formulation for the large displacement/small strain formulation and the ULH formulation for the large displacement/large strain formulation.

- The mixed (u/p) formulation can be employed in the analyses of porous structures. However the incompatible modes feature cannot be employed in the analyses of porous structures.
- The types of problems for which the porous media model can be employed are:

Transient static analysis (consolidation): The pore pressure distribution, the displacement and stress distribution, and their changes with time are of great interest.

Transient dynamic analysis: In addition to the pore pressure, displacement and stress results, the potential failure of the porous structure will draw much more attention. Liquefaction, which is an extremely important failure mode for infrastructures in an earthquake disaster, can be studied with the porous media model.

Undrained analysis: No specific formulation is given to relate the pore pressure to total stresses under undrained conditions. The undrained analysis can be performed by specifying undrained boundary conditions and small time steps.

- All the material models available for the solid elements (including elastic-plastic material models) can be employed in the porous media analysis. You can use whichever material model is most physically meaningful for your problem.
- The general equations governing the porous media model are as follows:
 - The macroscopic stresses must satisfy the equilibrium condition:

$$\sigma_{ij,j} + f_i = 0 \quad (3.12-1)$$

where the total (Cauchy) stress is

$$\sigma_{ij} = C^e_{ijkl} \varepsilon^e_{kl} + P_f \delta_{ij}, \quad i, j, k, l = 1, 2, 3 \quad (3.12-2)$$

P_f is the pore pressure, C_{ijkl}^e is the fourth order elasticity tensor, ε_{kl}^e is the elastic strain tensor and δ_{ij} is the Kronecker delta.

- The continuity condition for slightly compressible fluid flows must be satisfied:

$$\nabla \cdot (\mathbf{k} \cdot \nabla P_f) = \frac{\partial \varepsilon_{ii}}{\partial t} + \frac{n}{\kappa_f} \frac{\partial P_f}{\partial t} \quad (3.12-3)$$

where \mathbf{k} is the permeability matrix of the porous material, ε_{ii} is the volumetric strain of the porous skeleton, n is the porosity of the material and κ_f is the bulk modulus of the fluid. If the fluid is assumed incompressible, the last term in (3.12-3) will be zero.

- After a finite element discretization, equations (3.12-1) and (3.12-3) yield the following equations.

Transient statics:

$$\begin{bmatrix} \mathbf{0} & \mathbf{0} \\ t+\Delta t \mathbf{K}_{u p_f}^T & t+\Delta t \mathbf{K}_{p_f p_f}^{(1)} \end{bmatrix} \begin{bmatrix} t+\Delta t \dot{\mathbf{U}} \\ t+\Delta t \dot{\mathbf{P}}_f \end{bmatrix} + \begin{bmatrix} t+\Delta t \mathbf{K}_{uu} & t+\Delta t \mathbf{K}_{u p_f} \\ \mathbf{0} & -t+\Delta t \mathbf{K}_{p_f p_f}^{(2)} \end{bmatrix} \begin{bmatrix} t+\Delta t \mathbf{U} \\ t+\Delta t \mathbf{P}_f \end{bmatrix} = \begin{bmatrix} t+\Delta t \mathbf{R}_u \\ t+\Delta t \mathbf{R}_{p_f} \end{bmatrix} \quad (3.12-4)$$

Transient dynamics:

$$\begin{bmatrix} t+\Delta t \mathbf{M} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} t+\Delta t \ddot{\mathbf{U}} \\ t+\Delta t \ddot{\mathbf{P}}_f \end{bmatrix} + \begin{bmatrix} \mathbf{C} & \mathbf{0} \\ t+\Delta t \mathbf{K}_{u p_f}^T & t+\Delta t \mathbf{K}_{p_f p_f}^{(1)} \end{bmatrix} \begin{bmatrix} t+\Delta t \dot{\mathbf{U}} \\ t+\Delta t \dot{\mathbf{P}}_f \end{bmatrix} + \begin{bmatrix} t+\Delta t \mathbf{K}_{uu} & t+\Delta t \mathbf{K}_{u p_f} \\ \mathbf{0} & -t+\Delta t \mathbf{K}_{p_f p_f}^{(2)} \end{bmatrix} \begin{bmatrix} t+\Delta t \mathbf{U} \\ t+\Delta t \mathbf{P}_f \end{bmatrix} = \begin{bmatrix} t+\Delta t \mathbf{R}_u \\ t+\Delta t \mathbf{R}_{p_f} \end{bmatrix} \quad (3.12-5)$$

For a finite element, these matrices are defined as follows (omitting the left superscript for brevity):

$$\begin{aligned}\mathbf{K}_{uu} &= \int_V \mathbf{B}_u^T \mathbf{D} \mathbf{B}_u dV \\ \mathbf{K}_{up_f} &= \int_V \mathbf{B}_u^T \mathbf{I} \mathbf{H}_{p_f} dV \\ \mathbf{K}_{p_fp_f}^{(1)} &= \frac{1}{\kappa_f} \int_V n \mathbf{H}_{p_f}^T \mathbf{H}_{p_f} dV \\ \mathbf{K}_{p_fp_f}^{(2)} &= \int_V \mathbf{B}_{p_f}^T \mathbf{k} \mathbf{B}_{p_f} dV \\ \mathbf{R}_u &= \int_V \mathbf{H}_u^T \mathbf{f} dV + \int_{S_f} \left(\mathbf{H}_u^s \right)^T \mathbf{f} dS_f \\ \mathbf{R}_{p_f} &= \int_{S_q} \left(\mathbf{H}_{p_f}^s \right)^T \mathbf{q} dS_q\end{aligned}$$

In the above expressions, \mathbf{D} is the constitutive matrix (possibly elasto-plastic), \mathbf{f} and \mathbf{q} are load vectors (force, traction and flux), \mathbf{B}_u and \mathbf{B}_{p_f} are gradient matrices for the displacements \mathbf{U} and pore pressures \mathbf{P}_f , \mathbf{H}_u and \mathbf{H}_{p_f} are gradient matrices for the displacements \mathbf{U} and pore pressures \mathbf{P}_f , and \mathbf{I} is the identity tensor.

- Numerically integrating equations (3.12-4) and (3.12-5) with respect to time gives the final equations used for the porous media model:

Transient statics: the Euler backward method is used:

$$\begin{aligned}
& \begin{bmatrix} {}^{t+\Delta t} \mathbf{K}_{uu} & {}^{t+\Delta t} \mathbf{K}_{u p_f} \\ {}^{t+\Delta t} \mathbf{K}_{u p_f}^T & {}^{t+\Delta t} \mathbf{K}_{p_f p_f}^{(1)} - \Delta t {}^{t+\Delta t} \mathbf{K}_{p_f p_f}^{(2)} \end{bmatrix} \begin{bmatrix} {}^{t+\Delta t} \mathbf{U} \\ {}^{t+\Delta t} \mathbf{P}_f \end{bmatrix} \\
& = \begin{bmatrix} {}^{t+\Delta t} \mathbf{R}_u \\ -\Delta t {}^{t+\Delta t} \mathbf{R}_{p_f} + {}^{t+\Delta t} \mathbf{K}_{u p_f}^T {}^t \mathbf{U} + {}^{t+\Delta t} \mathbf{K}_{p_f p_f}^{(1)} {}^t \mathbf{P}_f \end{bmatrix} \tag{3.12-6}
\end{aligned}$$

Transient dynamics: the Euler backward method is used for the continuity equation and the Newmark method is used for the equilibrium equation:

$$\begin{aligned}
& \begin{bmatrix} {}^{t+\Delta t} \mathbf{K}_{uu} + a_0 {}^{t+\Delta t} \mathbf{M} + a_1 {}^{t+\Delta t} \mathbf{C} & {}^{t+\Delta t} \mathbf{K}_{u p_f} \\ {}^{t+\Delta t} \mathbf{K}_{u p_f}^T & {}^{t+\Delta t} \mathbf{K}_{p_f p_f}^{(1)} - \Delta t {}^{t+\Delta t} \mathbf{K}_{p_f p_f}^{(2)} \end{bmatrix} \begin{bmatrix} {}^{t+\Delta t} \mathbf{U} \\ {}^{t+\Delta t} \mathbf{P}_f \end{bmatrix} \\
& = \begin{bmatrix} {}^{t+\Delta t} \mathbf{R}_u^d \\ -\Delta t {}^{t+\Delta t} \mathbf{R}_{p_f} + {}^{t+\Delta t} \mathbf{K}_{u p_f}^T {}^t \mathbf{U} + {}^{t+\Delta t} \mathbf{K}_{p_f p_f}^{(1)} {}^t \mathbf{P}_f \end{bmatrix} \tag{3.12-7}
\end{aligned}$$

in which

$$\begin{aligned}
{}^{t+\Delta t} \mathbf{R}_u^d &= {}^{t+\Delta t} \mathbf{R}_u + {}^{t+\Delta t} \mathbf{M} (a_0 {}^t \mathbf{U} + a_2 {}^t \dot{\mathbf{U}} + a_3 {}^t \ddot{\mathbf{U}}) + \\
&\quad {}^{t+\Delta t} \mathbf{C} (a_1 {}^t \mathbf{U} + a_4 {}^t \dot{\mathbf{U}} + a_5 {}^t \ddot{\mathbf{U}}) \tag{3.12-8}
\end{aligned}$$

and where a_0, a_1, \dots, a_5 are the integration constants for the Newmark method (see Ref. KJB, Section 9.2.4).

- Please note the following:
 - Usually, the global stiffness matrix is no longer positive definite in porous media analysis. This is because the diagonal components of ${}^{t+\Delta t} \mathbf{K}_{p_f p_f}^{(1)} - \Delta t {}^{t+\Delta t} \mathbf{K}_{p_f p_f}^{(2)}$ in equations (3.12-6) and (3.12-7) are usually negative. Therefore, you should request that ADINA continue execution after ADINA detects a non-positive definite stiffness matrix (see Section 7.1.1).

- ▶ It is important to have a reasonable scheme for choosing time steps. Theoretically, the solution process is unconditionally stable for any time step size because of the use of the Euler backward method. However, this does not necessarily imply that any time step size is permissible. A small time step size and large mesh size will result in the zigzag pore pressure distribution at the initial stage of pore fluid dissipation (Britto & Gunn, 1987). It is logical to use progressively larger time steps in the porous media analysis.
 - ▶ The pore pressures are only computed by ADINA at the corner nodes of solid elements. During postprocessing, the AUI interpolates the pore pressures from the corner nodes to the other nodes (e.g. mid-side nodes) in higher-order elements.
 - ▶ Once the porous solid material properties are specified, the corresponding pore fluid properties are automatically computed. Certainly, they can be redefined if necessary.
 - ▶ Porous media cannot be used in explicit dynamic analysis.
- For more information, see the following reference:
ref. Britto, A.M. and Gunn, M.J., *Critical state soil mechanics via finite elements*, Ellis Horwood Limited, 1987.

3.13 Gasket material model

- Gaskets are relatively thin components placed between two bodies/surfaces to create a sealing effect and prevent fluid leakage (see Figure 3.13-1). While most gaskets are flat, any arbitrary gasket geometry can be modeled in ADINA.

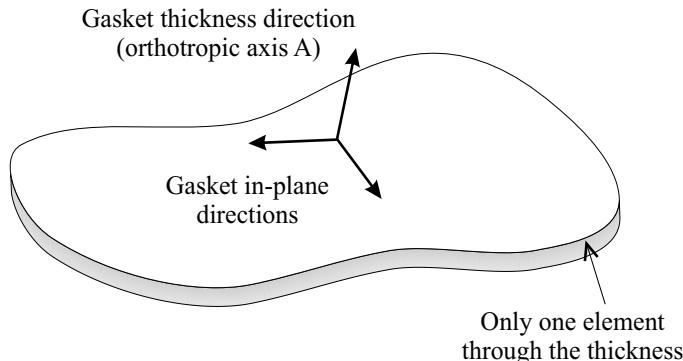


Figure 3.13-1: Schematic of gasket

- The sealing effect is created when the compressive load, applied in the direction of the gasket thickness, exceeds the initial yield stress of the gasket. The sealing effect is maintained as long as the compressive stress does not drop beyond a specified threshold value. The gasket ruptures if the compressive stress exceeds the gasket's ultimate stress. Unlike rupture, if a gasket leaks it still maintains its load-deflection characteristics.
- The gasket model can be used with 2-D solid and 3-D solid elements. It can also be used with **small displacement/small strain, large displacement/small strain** formulations.
- The gasket behaves as a nonlinear elastic-plastic material when compressed in the thickness or gasket direction. Its load-deformation characteristics are typically represented by pressure-closure curves. Tensile stiffness can be assumed to be constant or zero. Hardening is assumed to be isotropic. The closure strain is always measured as the change in gasket thickness divided by the original gasket thickness. The gasket's uni-directional plasticity model speeds up computations, and allows more flexibility in defining the shape of the loading and unloading curves.
- Two gasket types are implemented in ADINA, the full or regular gasket, and the simple gasket. In the simple gasket all in-plane deformations are neglected. In this case, the gasket only possesses thickness stiffness and the in-plane degrees of freedom can be eliminated. In the full gasket, elastic material properties are

assumed for the gasket plane.

- Figure 3.13-2 shows a typical pressure-closure relationship. It consists of a main loading curve consisting of any number of elastic segments and any number of plastic loading segments. Loading/unloading curves can be provided for different points on the loading curve. However, it is not necessary to define a loading/unloading curve for each point on the main loading curve.

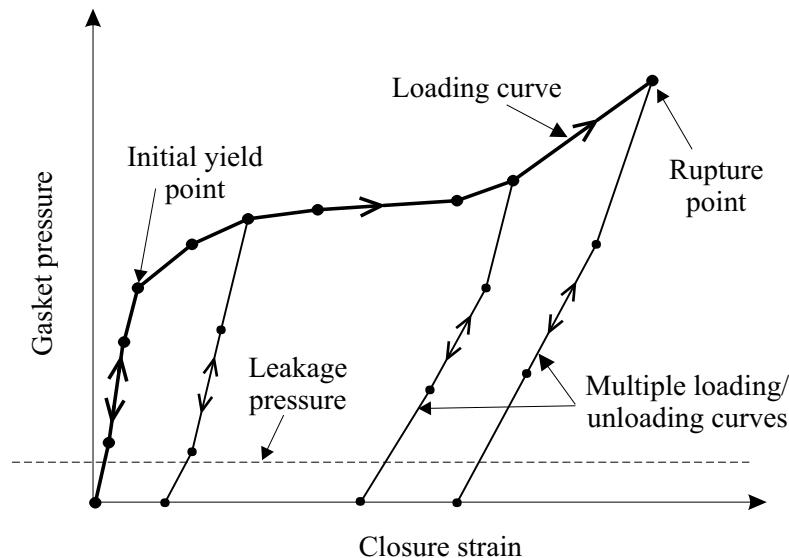


Figure 3.13-2: Pressure-closure relationship for a gasket material

- Each gasket can have one of the following five states:

Open:	The gasket pressure is less than the leakage pressure.
Closed:	The gasket pressure is higher than the leakage pressure but has not yet caused plasticity.
Sealed:	There has been plastic gasket deformation and the current pressure is above the gasket leakage pressure.
Leaked:	After plastic deformation, the gasket pressure has dropped below gasket leakage pressure.

Crushed: Gasket closure strain has exceeded the rupture value.

Modeling issues

- The gasket should be modeled as a single layer of 2-D or 3-D elements. Only linear elements are possible (3 and 4-noded elements in 2-D and 4,6 and 8-noded elements in 3-D).
- Since the gasket has different properties in different directions, it is considered by ADINA to be an orthotropic material. Material axis A is set to the gasket normal direction. The AUI attempts to automatically define the gasket's material axes if they are not explicitly defined by the user.
- If a simple gasket is selected, the local X-axis at the gasket nodes (or Y-axis in 2-D) must coincide with the gasket's normal direction, using skew coordinate systems if necessary. The simple gasket is disconnected from the remaining in-plane degrees of freedom. The AUI attempts to automatically define the gasket's skew directions if they are not set by the user.
- The top and bottom surfaces of a gasket can be separate from those of the mating surfaces. In this case, they should be connected via contact. The gasket can also share a common surface with the intended mating surface. In this case, contact is not needed, however, the gasket cannot separate from its target.
- The number of points in all loading/unloading curves must be identical for efficiency. Also the last point in each loading/unloading curve must be one of the input point on the main loading curve.

The input: In addition to the pressure-closure relationships the following material properties are needed for a gasket: density, transverse shear modulus, tensile Young's modulus, thermal expansion coefficient in thickness direction, reference temperature, and leakage pressure.

The following properties are only needed for a full gasket: in-plane Young's modulus, in-plane Poisson's ratio and in-plane thermal expansion coefficient.

Output variables: The following gaskets output variables are available for both gasket types: GASKET_PRESSURE, GASKET_CLOSURE_STRAIN, GASKET_THERMAL_STRAIN, GASKET_STRESS-BB, GASKET_STRESS-CC, GASKET_STRESS-AB, GASKET_STRESS-AC, GASKET_STRESS-BC, GASKET_STRAIN-BB, GASKET_STRAIN-CC, GASKET_STRAIN-AB, GASKET_STRAIN-AC, GASKET_STRAIN-BC, GASKET_THERMAL_STRAIN-BB, GASKET_THERMAL_STRAIN-CC, GASKET_THERMAL_STRAIN-AB, GASKET_THERMAL_STRAIN-AC, GASKET_THERMAL_STRAIN-BC, GASKET_YIELD_STRESS, GASKET_PLASTIC_CLOSURE_STRAIN, GASKET_STATUS, GASKET_DEFORMATION_MODE

See Section 11.1.1 for the definitions of these variables. Note that in 2-D analysis, only the AA, BC, CC components are output.

3.14 Shape Memory Alloy (SMA) material model

- The Shape Memory Alloy (SMA) material model is intended to model the superelastic effect (SE) and the shape memory effect (SME) of the shape-memory alloys using the finite element method.
- One representative of SMA is Nickel Titanium (NiTi), can undergo solid-to-solid phase transformations induced by stress or temperature. The high temperature phase is called austenite (A) with a body-centered cubic structure and the low-temperature phase is called martensite with a monoclinic crystal structure in several variants. Fig.3.14-1 shows a schematic SMA stress-temperature diagram. The martensite (M) phase in two generalized variants and the austenite phase are shown, as well as stress- and temperature-dependent transformation conditions. The martensite phase is favored at low temperatures or high stresses.

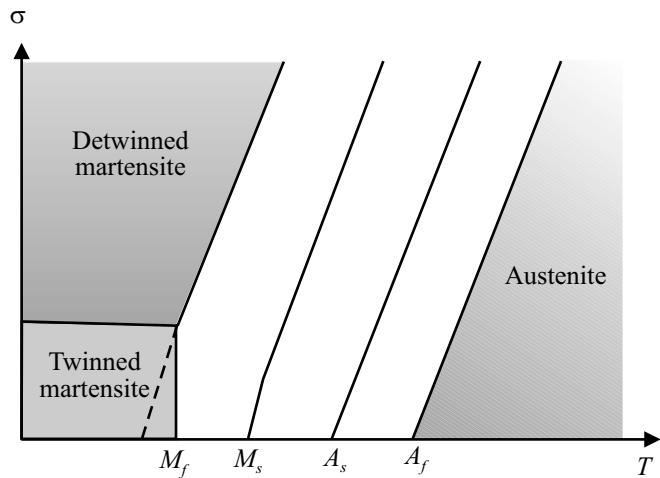


Figure 3.14-1: SMA stress-temperature phase diagram

- The typical uniaxial isothermal stress-strain curve is shown in Fig. 3.14-2. The SE effect is defined at temperature $T > A_f$ and is displayed in Fig. 3.14-2(a). The stress cycle application induces transformations from A \rightarrow M and then from M \rightarrow A to exhibit the hysteresis loop. The SME effect is defined at temperature $T < A_s$ and is displayed in Fig. 3.14-2(b). A residual transformation strain remains after unloading; however heating the material to temperature above A_f leads to thermally induced M \rightarrow A transformation and the recovery of transformation strain.

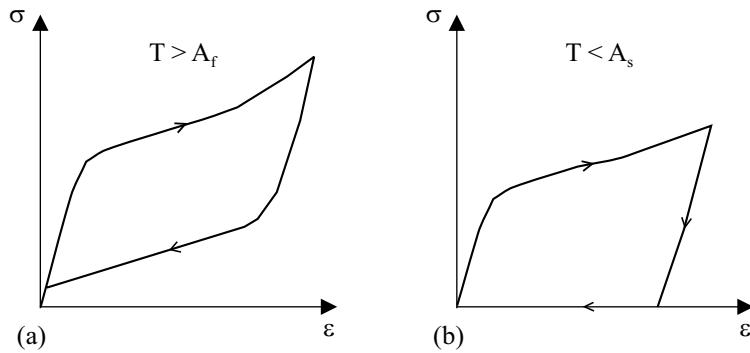


Figure 3.14-2: Schematic of stress-strain curves for shape-memory alloys
 (a) pseudoelasticity, and (b) shape memory effect

- The SMA material model in ADINA is based on the following equations:

- ▶ The equation of total strain,

$$\varepsilon = \varepsilon^e + \varepsilon^t + \varepsilon^\theta$$

where

ε^e = elastic strain

ε^θ = thermal strain

ε^t = transformation strain; to be evaluated

- ▶ The equations of one-dimensional macro-scale modeling,

$$\xi = \xi_s + \xi_t ; \quad 0 \leq \xi \leq 1$$

$$\xi + \xi_A = 1$$

$$\varepsilon^t = \varepsilon_{max}^t \xi_s$$

$$\sigma = ((1 - \xi) E_A + \xi E_M) (\varepsilon - \varepsilon_{max}^t \xi_s)$$

ξ_t	= twinned martensite volume fraction
ξ_s	= detwinned martensite volume fraction
ξ_A	= austenite volume fraction
ε_{max}^t	= maximum residual strain; a material property

- The flow rule of three-dimensional constitutive model,

$$\Delta \varepsilon_{ij}^t = \Delta \xi_s \varepsilon_{max}^t n_{ij}^t$$

$$n_{ij}^t = \sqrt{\frac{3}{2}} \left(\frac{s_{ij}}{\bar{\sigma}} \right); \text{ for the martensitic transformation}$$

$$n_{ij}^t = \sqrt{\frac{3}{2}} \left(\frac{\varepsilon_{ij}^t}{\bar{\varepsilon}^t} \right); \text{ for the reverse transformation}$$

s_{ij} = deviatoric stresses

$$\bar{\sigma} = \sqrt{\frac{3}{2} s_{ij} s_{ij}}$$

$$\bar{\varepsilon}^t = \sqrt{\frac{3}{2} \varepsilon_{ij}^t \varepsilon_{ij}^t}$$

- The phase transformation conditions,
 - for the martensitic transformation as starting condition

$$f_{M_S} = \sqrt{3J_2} - C_M (\theta - M_S)$$

- for the martensitic transformation as ending condition

$$f_{M_f} = \sqrt{3J_2} - C_M (\theta - M_f)$$

3. for the reverse transformation as starting condition

$$f_{A_S} = \sqrt{3J_2} - C_A(\theta - A_S)$$

4. for the reverse transformation as ending condition

$$f_{A_f} = \sqrt{3J_2} - C_A(\theta - A_f)$$

- The phase transformation rate using linear kinetic rule (Auricchio and E. Sacco, 1999),

$$\Delta\xi = \frac{R_\xi}{f_f} \Delta f$$

$$\Delta f = \frac{3}{2} \frac{{}^{t+\Delta t}J_2 - {}^tJ_2}{{}^{t+\Delta t}\bar{\sigma}} - c({}^{t+\Delta t}\theta - {}^t\theta)$$

where

c = the specific heat per unit volume

$f_f = -f_{M_f}c = C_M$ and $R_\xi = 1 - \xi$ (for the martensitic transformation)

$f_f = -f_{A_f}c = C_A$ and $R_\xi = \xi$ (for the reverse transformation)

- Computational steps for the stress-integration of the SMA model are as follows (Kojic and Bathe, 2005):

1. Calculate trial stresses, ${}^{t+\Delta t}S_{ij} = \frac{E({}^t\xi)}{1 + \nu({}^t\xi)} ({}^{t+\Delta t}\dot{\epsilon}_{ij} - {}^t\dot{\epsilon}_{ij}^t)$

2. Check for martensitic transformation,

$$f_{M_f} f_{M_s} < 0 \text{ and } {}^t\xi < 1 \text{ and } \Delta f > 0$$

Check for reverse transformation,

$$f_{A_f} f_{A_s} < 0 \text{ and } {}^t\xi > 0 \text{ and } \Delta f < 0$$

3. Solve for martensitic transformation with the governing equations as follows,

$$\begin{aligned} g(\Delta\xi) &= \Delta\xi - \frac{t+\Delta t}{t+\Delta t} R_\xi \left[\frac{3}{2} \frac{{}^{t+\Delta t} J_2(\Delta\xi) - {}^{t+\Delta t} J_2}{{}^{t+\Delta t} \bar{\sigma}} - c({}^{t+\Delta t} \theta - {}^t \theta) \right] \\ {}^{t+\Delta t} J_2(\Delta\xi) &= \frac{1}{2} {}^{t+\Delta t} S_{ij} {}^{t+\Delta t} S_{ij} \\ {}^{t+\Delta t} S_{ij} &= \frac{{}^{t+\Delta t} E(\Delta\xi)}{1 + {}^{t+\Delta t} \nu(\Delta\xi)} \left({}^{t+\Delta t} \varepsilon_{ij}'' - \Delta\xi_s(\Delta\xi) \cdot \varepsilon_{\max}^t {}^{t+\Delta t} n_{ij}^t \right) \end{aligned} \quad (3.14-1)$$

$$R_\xi = 1 - \xi \text{ and } f_f = -f_{M_f}$$

4. Solve for reverse transformation using the governing equations (3.14-1) with

$$R_\xi = \xi \text{ and } f_f = f_{A_f}$$

5. Solve for martensitic re-orientation with the governing equations as follows,

$$g_2(\Delta\xi_s) = \sqrt{\frac{3}{2} {}^{t+\Delta t} S_{ij} {}^{t+\Delta t} S_{ij}} - C_R {}^{t+\Delta t} \theta - \sigma_R \quad (3.14-2)$$

$${}^{t+\Delta t} S_{ij} = \frac{{}^{t+\Delta t} E(\Delta\xi)}{1 + {}^{t+\Delta t} \nu(\Delta\xi)} ({}^{t+\Delta t} \varepsilon_{ij}^{\prime\prime} - \Delta\xi_s(\Delta\xi) \cdot \varepsilon_{\max}^t \cdot {}^{t+\Delta t} n_{ij}^t)$$

The martensite reorientation calculation step is optional; it is activated when $\sigma_R > 0$ is input.

6. Update history-dependent variables for this time step/iteration step.
7. Calculate the consistent tangent constitutive matrices.

- ref. M. Kojic and K.J. Bathe, *Inelastic Analysis of Solids and Structures*, Springer, 2005
- ref. F. Auricchio and E. Sacco, “A Temperature-Dependent Beam for Shape-Memory Alloys: Constitutive Modelling, Finite-Element Implementation and Numerical Simulation”, *Computer Methods in Applied Mechanics and Engineering*, 174:pp. 171-190 (1999)

3.15 User-supplied material model

3.15.1 General considerations

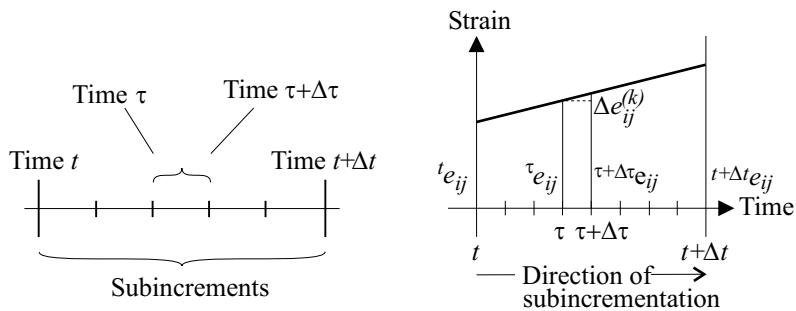
- The user-supplied model is provided in ADINA to allow you to construct any type of material model for use with 2-D or 3-D solid elements.
- The constitutive relation must be arranged in the form of an algorithm and inserted into subroutine CUSER2 for 2-D solid elements or into subroutine CUSER3 for 3-D solid elements.
- The material model can be used with the following formulations:

- ▶ Materially-nonlinear-only, for infinitesimally small displacements and strains (MNO)
 - ▶ total Lagrangian, in which the displacements and rotations can be very large, but the strains may have to be small or can be large depending on the material characteristics (TL)
 - ▶ updated Lagrangian Hencky, in which case the strains can be large (ULH)
- Either the displacement based elements or the mixed-interpolated displacement/pressure elements can be used with the user-supplied material model. The mixed-interpolated elements are preferred for (nearly) incompressible materials.
 - The use of the user-supplied model is described in the following, first for 2-D solid elements and then for 3-D solid elements. Then the user-supplied material models for which the source code is available in ADINA are described.
 - Note that even if the material properties are not temperature dependent, or the temperature loading is zero, both the temperature table and the temperature loading input must be entered accordingly in order that the .dat file be generated for computation.

3.15.2 2-D solid elements

Basic concept of implementation: The total solution process for a load (time) step involves updating the stresses from time t to time $t + \Delta t$ and updating any history dependent quantities. ADINA performs this updating by subdividing the total strain increment from time t to time $t + \Delta t$ into subincrements and, for each subincrement, calling the subroutine CUSER2 (see Figs. 3.15-1 and 3.15-2).

- A sample version of CUSER2 may be found in file ovl30u_vp1.f. The coding within this sample version is fully discussed in the discussion of the viscoplastic model (see below).



	Values provided and passed back through the calling arguments of subroutine CUSER2	
Variable or Array name	At entry, the arguments correspond to time:	At exit, the arguments correspond to time:
EPS(4)	t	t
STRAIN(4)	$t+\Delta t$	$t+\Delta t$
THSTR1(4)	τ	τ
THSTR2(4)	$\tau+\Delta\tau$	$\tau+\Delta\tau$
TMP1	τ	τ
TMP2	$\tau+\Delta\tau$	$\tau+\Delta\tau$
ALFA	τ	τ
ALFAA	$\tau+\Delta\tau$	$\tau+\Delta\tau$
CTD(*)	τ	τ
CTDD(*)	$\tau+\Delta\tau$	$\tau+\Delta\tau$
ARRAY(*)	τ	$\tau+\Delta\tau$
IARRAY(*)	τ	$\tau+\Delta\tau$
STRESS(4)	τ	$\tau+\Delta\tau$

Figure 3.15-1: Subinforcement process for forward stress integration
Convention used in subroutine CUSER3

- The stress integration can be performed by employing forward integration or backward integration. Details regarding these two stress integration schemes are given below.

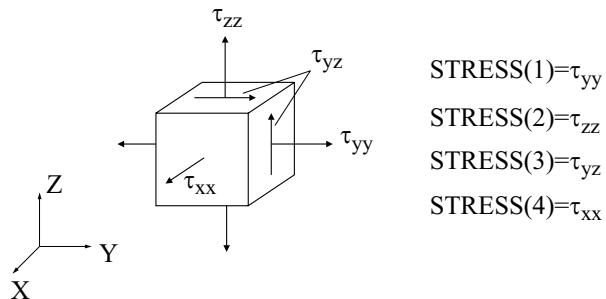


Figure 3.15-2: Stress convention for 2-D elements

Forward stress integration

In this case, the stress $\sigma^{(k)}$ at the end of a subincrement k is calculated as

$$\sigma^{(k)} = \sigma^{(k-1)} + \Delta\sigma^{(k)} \quad (3.15-1)$$

where $\sigma^{(0)} = {}^t\sigma$. The stress increment $\Delta\sigma^{(k)}$ corresponds to the strain increment $\Delta e^{(k)}$ and temperature increment $\Delta\theta$.

For example

$$\Delta\sigma^{(k)} = \mathbf{C}^E \left(\Delta e^{(k)} - \Delta e^{th(k)} - \Delta e^{IN(k)} \right) \quad (3.15-2)$$

where \mathbf{C}^E is the elastic constitutive matrix and $\Delta e^{th(k)}$ and $\Delta e^{IN(k)}$ are the increments of thermal and inelastic strains in the subincrement. It is assumed that the total strain varies linearly in the time step.

- The evaluation of the strain subincrements is performed by ADINA prior to calling subroutine CUSER2. Your coding must update the stresses (in array STRESS(4)) and other history dependent variables for each subincrement, and print the stresses once the final stresses for the load (time) step have been evaluated.

Your coding uses the temperature-dependent properties ALFA, ALFAA, CTD(*), CTDD(*) and the temperature-independent properties CTI(*) in the calculation of the stresses and the constitutive matrix. The temperature-dependent properties at time τ are calculated by ADINA at each integration point prior to calling subroutine CUSER2.

- Note that the stress integration in the subincrement can be explicit or implicit. If the integration is implicit, your coding must include the appropriate iterative scheme for the stress calculation.

Backward stress integration

- The stress calculation is as follows:

First, the subroutine CUSER2 is called for the evaluation of the trial elastic stress σ_*^E corresponding to the total and thermal strains at time $t + \Delta t$, ${}^{t+\Delta t}\mathbf{e}$ and ${}^{t+\Delta t}\mathbf{e}^{th}$, respectively, and to the inelastic strains at time t , ${}^t\mathbf{e}^{IN}$. For example

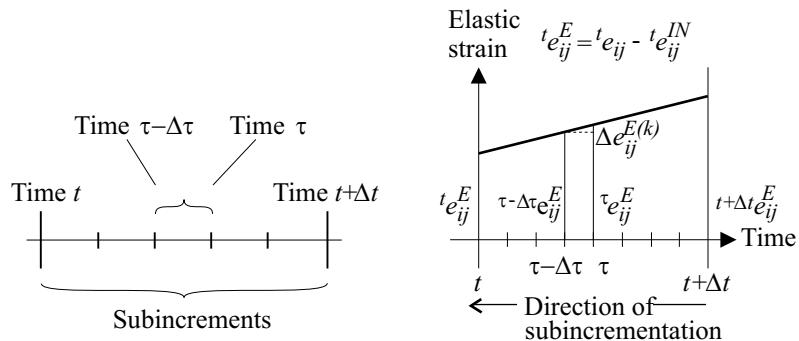
$${}^{t+\Delta t}\sigma_*^E = {}^{t+\Delta t}\mathbf{C}^E \left({}^{t+\Delta t}\mathbf{e} - {}^{t+\Delta t}\mathbf{e}^{th} - {}^t\mathbf{e}^{IN} \right) \quad (3.15-3)$$

where ${}^{t+\Delta t}\mathbf{C}^E$ is an elastic constitutive matrix corresponding to temperature ${}^{t+\Delta t}\theta$. Calculation of the trial elastic stress must be provided by your coding.

Next, the subroutine CUSER2 is called within the subincrementation loop, as schematically shown in Fig. 3.15-3. The stress correction $\Delta\sigma^{(k)}$ in subincrement k (Eq. (3.15-1)) now corresponds to the increment of inelastic strain $\Delta\mathbf{e}^{IN(k)}$. For example

$$\Delta\sigma^{(k)} = -\mathbf{C}^{E(k)} \Delta\mathbf{e}^{IN(k)}$$

where $\mathbf{C}^{E(k)}$ is the elastic constitutive matrix corresponding to temperature ${}^{t-\Delta\tau}\theta$. As indicated in Fig. 3.15-3, the direction of subincrementation is now from $t + \Delta t$ toward t .



	Values provided and passed back through the calling arguments of subroutine CUSER2	
Variable or Array name	At entry, the arguments correspond to time:	At exit, the arguments correspond to time:
EPS(4)	t	t No change in value
STRAIN(4)	$t+\Delta t$ / τ (small/large strains)	$t+\Delta t$ / $\tau-\Delta\tau$
THSTR1(4)	τ	τ
THSTR2(4)	$\tau-\Delta\tau$	$\tau-\Delta\tau$
TMP1	τ	τ
TMP2	$\tau-\Delta\tau$	$\tau-\Delta\tau$
ALFA	τ	τ
ALFAA	$\tau-\Delta\tau$	$\tau-\Delta\tau$
CTD(*)	τ	τ
CTDD(*)	$\tau-\Delta\tau$	$\tau-\Delta\tau$
ARRAY(*)	τ	$\tau-\Delta\tau$
IARRAY(*)	τ	$\tau-\Delta\tau$
STRESS(4)	τ	$\tau-\Delta\tau$
DPSP(4)	τ (ULH large strains)	$\tau-\Delta\tau$

Figure 3.15-3: Subincrementation process for backward stress integration
Convention used in subroutine CUSER3

- The concept of evaluation of variables for the current subincrement is the same as above for the forward integration. The

additional parameter in the backward (the return mapping) procedure is ISUBM. Namely, the flag ISUBM is set to zero at evaluation of the trial elastic solution, and should not be changed during the subincrementation until the criteria for discontinuing the subincrementation are satisfied (in case of viscoplasticity, for example, the stopping criterion corresponds to the time when the stress state falls inside the yield surface). When the criteria for stopping the subincrementation are satisfied, your coding must reset ISUBM to -1 and this condition signals ADINA to stop the subincrementation loop for the current integration point. The stresses and other history parameters evaluated at the condition ISUBM = -1 are considered by ADINA to be the solutions for the current time step.

Note that in the case of the updated Lagrangian Hencky formulation only the backward stress integration can be employed. Also, the increments of inelastic strains evaluated by your coding must be passed to ADINA for the calculation of the inelastic deformation gradient in ADINA (array DPSP(4)).

Note that the integration of stresses in the subincrement can be explicit or implicit. In case of an implicit scheme, your coding must include the appropriate iterative procedure for the stress calculation.

History dependent variables are stored at each integration point to characterize the history of the material behavior:

STRAIN(4): Strain components at time $t + \Delta t$. For the MNO and TL formulations, STRAIN(4) are total strains; that is, STRAIN(4) includes all strain contributions (e.g, elastic, inelastic, thermal). For the ULH formulation, STRAIN(4) are elastic strains (total minus inelastic).

ARRAY(LGTH1): Variable length working array of material history parameters (floating point variables).

IARRAY(LGTH2): Variable length working array of material history parameters (integer variables).

Material properties, algorithm control parameters: You can specify the material properties and algorithm control parameters to the user-supplied material model using the AUI, material type user-

supplied. This information is passed to subroutine CUSER2 as follows:

LGTH1: Length of array ARRAY.

LGTH2: Length of array IARRAY.

NCTI: Number of material property constants (maximum 99).

NSCP: Number of solution control parameters (maximum 99).

NCTD: Number of temperature-dependent material constants (maximum 98).

CTI(NCTI): Temperature-independent material constants.

CTD(NCTD): Temperature-dependent material constants for the given temperature at time τ . ADINA calculates CTD(NCTD) from the table of CTD(NCTD) vs. temperature entered into the AUI.

CTDD(NCTD): Temperature-dependent material constants for the given temperature at time $\tau + \Delta\tau$ (forward stress integration) or $\tau - \Delta\tau$ (backward stress integration). ADINA calculates CTDD(NCTD) from the table of CTD(NCTD) vs. temperature entered into the AUI.

SCP(NSCP): Solution control parameters.

ALFA: Coefficient of thermal expansion for the given temperature at time τ . ADINA calculates ALFA from the table of α vs. temperature entered into the AUI.

ALFAA: Coefficient of thermal expansion for the given temperature at time $\tau + \Delta\tau$ (forward stress integration) or $\tau - \Delta\tau$ (backward stress integration). ADINA calculates ALFAA from the table of α vs. temperature entered into the AUI.

Temperatures: The temperatures are passed to subroutine CUSER2 as follows:

TMP1: Temperature at time τ .

TMP2: Temperature at time $\tau + \Delta\tau$ (forward stress integration) or $\tau - \Delta\tau$ (backward stress integration).

The variable KEY: Subroutine CUSER2 is called for all integration points of each 2-D solid element and is used to perform the following four types of operations during the various phases. The different operations to be performed are controlled by the integer variable KEY.

KEY = 1: called once at the beginning of the analysis.

This indicates the initialization of the variables stored at the integration points, and is performed only once during the input phase. The arrays STRAIN, STRESS, ARRAY and IARRAY are automatically initialized by ADINA to zero. Your coding must set ARRAY and IARRAY to their proper initial values, if different from zero.

KEY = 2: called for each subincrement within a load (time) step.

This indicates the calculation of the element stresses in the solution of the governing equilibrium equations for the nodal displacements, and in the stress calculation phase after the nodal displacements have been evaluated. The calculation of the element stresses at each integration point is performed using the user-supplied coding inserted in subroutine CUSER2. The variables available in the subroutine and the notations/dimensions used are given in the listing of the sample subroutine (in file ovl30u_vp1.f). In addition to the stresses, the arrays ARRAY and IARRAY (if used to store history dependent variables) must be updated when KEY = 2.

Two stress integration schemes are available for the calculation of stresses from strains: forward integration and backward integration.

Forward stress integration: In this case, the stresses are calculated according to Eq. (3.13-1) and the user-supplied coding in the subroutine CUSER2 is in the form

```
DO 100 I = 1, N
STRESS(I) = 0.D0
DO 100 J = 1, N
100 STRESS(I) = STRESS(I) + D(I,J)*DEPS(J)
```

where N is the number of stress or strain components, D(I,J) is the constitutive matrix which is to be evaluated also using the user-supplied coding, and DEPS stores the components of the elastic and inelastic strain subincrements. Note that DEPS is evaluated by dividing the total strain increment for the step into the appropriate number of equal subincrements and the thermal strain subincrement is subtracted from DEPS prior to calling CUSER2. Note also that the array STRESS contains the stresses calculated at the end of the previous strain subincrement when the array STRESS is passed to CUSER2.

If an elastic constitutive matrix is used, then all inelastic strain subincrements must be subtracted from DEPS before the above DO loop is executed.

If the material model employs a total strain theory, then the algorithm used to perform the stress calculation can be of the following form (assuming that in this case there are no thermal effects):

```
DO 100 I = 1, N
STRESS(I) = 0.D0
DO 100 J = 1, N
100 STRESS(I) = STRESS(I) + D(I,J)*STRAIN(J)
```

where N and D(I,J) are defined above and STRAIN stores the total strain components for the current step. Using the total strain procedure, the number of subdivisions used for the strain subincrements should be set to 1 for solution effectiveness.

Backward stress integration: In this case, your coding must first calculate the trial elastic stress according to Eq. (3.15-3); arrays STRAIN and THSTR2 contain the total strains and thermal strains as described in the listing of subroutine CUSER2. Your coding must store into the working arrays ARRAY and/or IARRAY the history of deformation variables, e.g., the inelastic strains ϵ^N , etc. and use them in the calculation of stresses.

In the calls to CUSER2 following calculation of the trial elastic stress, you must provide the coding for the stress increments due to the inelastic deformation within the subincrements. It is also important to note that the stopping criteria and the resetting of the parameter ISUBM must be properly coded.

Note that when using the ULH formulation, the array STRAIN(4) contains the trial elastic strains for KTR = 0; the array is further updated in ADINA by subtracting increments of inelastic strains DEPS(4) for each subdivision. Stresses and strains passed to subroutine CUSER2 correspond to deformation without rigid body rotations. The stresses STRESS(4) which are true (Cauchy) stresses are passed to subroutine CUSER2 only for printout (KEY = 4). The transformation that takes into account the material rotation is performed in ADINA prior to calling subroutine CUSER2.

An example of coding for the backward stress integration is given in the listing of the viscoplastic material example.

Note that when using the TL formulation, the stresses in array STRESS(4) correspond to the second Piola-Kirchhoff stress tensor. The Cauchy stresses, obtained by the transformation of the second Piola-Kirchhoff stress tensor in ADINA, are passed to the subroutine CUSER2 only for printout (KEY=4).

If you require the stress invariants, your coding can call subroutine XJ1232. The call statement

```
CALL XJ1232(STRESS,XI1,XI2,XI3,XJ1,XJ2,XJ3)
```

returns the three invariants XI1, XI2 and XI3 of the stress tensor, and the three invariants XJ1, XJ2 and XJ3 of the deviatoric stress tensor. These are calculated as follows by XJ1232:

$$\begin{aligned} \text{XI1} &= \tau_{ii}, & \text{XI2} &= 1/2 \tau_{ij}\tau_{ij}, & \text{XI3} &= 1/3 \tau_{ij}\tau_{jk}\tau_{ki} \\ \text{XJ1} &= 0, & \text{XJ2} &= 1/2 s_{ij}s_{ij}, & \text{XJ3} &= 1/3 s_{ij}s_{jk}s_{ki} \end{aligned}$$

where τ_{ij} is the stress tensor passed in STRESS(4) and s_{ij} is the deviatoric stress tensor corresponding to τ_{ij} . The summation convention is used in the above expressions.

Note that if the large displacement/large strain options are used,

stresses energetically conjugated with the corresponding strains (see Table 3.15-1) are employed in these calculations, and you cannot control this.

Table 3.15-1: Stress and strain measures used in the user-supplied material model. Note that the stresses are transformed internally by ADINA into true (Cauchy) stresses for printout (KEY=4)

Kinematic formulation	Strain measure	Stress measure
MNO (materially- nonlinear-only): small displacements, small strains	Infinitesimal strains	Cauchy stresses
TL (total Lagrangian): large displacements, small or large strains (depending upon the material model)	Green-Lagrange strains	2nd Piola-Kirchhoff stresses
ULH (updated Lagrangian Hencky): large displacements, large strains	Logarithmic Hencky (elastic) strain and plastic velocity strains	Cauchy stresses

KEY=3: called when the new tangent stiffness matrix is to be calculated.

This indicates the calculation of the constitutive matrix **D** which is to be employed in the evaluation of the tangent stiffness matrix.

The variables available for the constitutive matrix calculation are the values defined or calculated in the last subdivision of the stress calculation when KEY=2, as described above.

KEY=4: called for stress printout.

This indicates the printing of the element response in the stress calculation phase. No stress calculations need to be performed when KEY=4; your coding must only include the required format statements for the printing of the desired element results.

Note that if the TL formulation is employed, the ADINA program automatically transforms, just for printout, the stresses

used in the constitutive relations (see Table 3.15-1) to the Cauchy stresses in the global coordinate system. Then subroutine CUSER2 is called for stresses and strains printout.

If the ULH formulation is employed, total stretches PRST(3) are calculated and passed to subroutine CUSER2. PRST(1) stores the maximum in-plane stretch (maximum deviation from 1.0), PRST(2) is the smaller in-plane stretch and PRST(3) is the stretch in the X-direction. Also, the angles of the first and the second principal stretches with respect to the Y-axis, stored in the array ANGLE(2), are passed to CUSER2.

3.15.3 3-D solid elements

- The user-supplied material model for 3-D solid elements is entirely analogous to that for 2-D solid elements. Hence, refer to the above section for detailed descriptions.
- The user-supplied algorithm for the user-supplied model for 3-D solid elements is to be inserted in subroutine CUSER3.
- A sample version of CUSER3 may be found in file ovl40u_vp1.f. The coding within this sample version is fully discussed in the discussion of the viscoplastic model (see below).
- The convention used for stresses and strains is shown in Fig. 3.13-4.

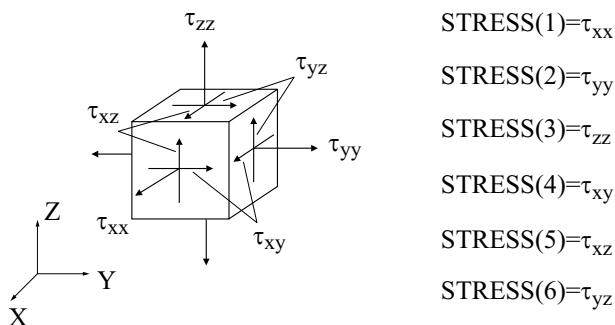


Figure 3.15-4: Stress/strain convention for 3-D elements

- When the ULH formulation is employed, the principal stretches and the direction cosines of the maximum principal stretch (with

maximum deviation from 1.0) are passed to the subroutine CUSER3 for the printout (KEY=4).

- If you require the stress invariants, your coding can call subroutine XJ123. The call statement

```
CALL XJ123(STRESS,XI1,XI2,XI3,XJ1,XJ2,XJ3)
```

returns the three invariants XI1, XI2 and XI3 of the stress tensor, and the three invariants XJ1, XJ2 and XJ3 of the deviatoric stress tensor. These invariants are defined as for the 2-D elements described above.

3.15.4 User-supplied material models supplied by ADINA R&D, Inc.

3.15.4.1 Viscoplastic material model

- A viscoplastic material model for axisymmetric or plane strain conditions is given as an example of the use of the CUSER2 subroutine. A viscoplastic model for fully 3-D conditions is given as an example of the CUSER3 subroutine. The coding in subroutine CUSER2 is based on the equations given hereafter.
- The CUSER2 and CUSER3 subroutine coding can be found in files ovl30u_vp1.f and ovl40u_vp1.f respectively.
- The rate of viscoplastic strain is given by

$${}^t\dot{e}_{ij}^{VP} = \gamma <\phi> \frac{3}{2} \frac{{}^tS_{ij}}{{}^t\bar{\sigma}} \quad (3.15-4)$$

where

${}^tS_{ij}$ = component of deviatoric stress

${}^t\bar{\sigma} = \sqrt{\frac{3}{2} {}^tS_{ij} {}^tS_{ij}}$ = effective stress

γ = fluidity parameter

$$<\phi> = \phi \quad \text{if } \phi > 0$$

$$= 0 \quad \text{otherwise}$$

In the above,

$$\phi = \frac{{}^t\bar{\sigma}}{\sigma_y} - 1 \quad (3.15-5)$$

where σ_y is the yield stress (above which the material is considered perfectly plastic).

- The material is subjected to mechanical and thermal loading and the material constants are considered to be temperature independent.
- Stress integration is performed using forward or backward integration.

Forward integration: In forward integration, the increment of stresses in the subincrement (Eq. (3.15-2)) are

$$\Delta\sigma_i^{(k)} = C_{ij}^E \left(\Delta e_j^{(k)} - \Delta e_j^{th(k)} - \Delta e_j^{VP(k)} \right) \quad (3.15-6)$$

where

$$\Delta\sigma_1 = \Delta\sigma_{yy}, \quad \Delta\sigma_2 = \Delta\sigma_{zz}, \quad \Delta\sigma_3 = \Delta\sigma_{yz}, \quad \Delta\sigma_4 = \Delta\sigma_{xx},$$

$$\Delta e_j^{(k)} = \frac{{}^{t+\Delta t}e_j - {}^t e_j}{\text{NSUBD}} \quad (3.15-7)$$

NSUBD is the number of subincrements,

$$\Delta e_j^{th(k)} = \alpha \left({}^{\tau+\Delta\tau}\theta - {}^\tau\theta \right) \text{ for } j = 1, 2, 4$$

$$\Delta e_j^{VP(k)} = \frac{3}{2} (\Delta\tau) (\gamma) \left(\frac{{}^\tau\bar{\sigma}}{\sigma_y} - 1 \right) \frac{{}^\tau s_j}{{}^\tau\bar{\sigma}} \quad (3.15-8)$$

Here α is the mean coefficient of thermal expansion, τ and $\Delta\tau$ are defined in Fig. 3.15-1, and the strains e_j are defined as

$$e_1 = e_{yy}, \quad e_2 = e_{zz}, \quad e_3 = \gamma_{yz}, \quad e_4 = e_{xx}$$

for time t and $t+\Delta t$. The same convention is used for increments of viscoplastic strains and deviatoric stresses. The coefficients C_{ij}^E represent the elastic constitutive matrix

$$\mathbf{C}^E = \frac{E(1-\nu)}{(1+\nu)(1-2\nu)} \begin{bmatrix} 1 & \frac{\nu}{(1-\nu)} & 0 & \frac{\nu}{(1-\nu)} \\ & 1 & 0 & \frac{\nu}{(1-\nu)} \\ SYMM & & \frac{(1-2\nu)}{2(1-\nu)} & 0 \\ & & & 1 \end{bmatrix} \quad (3.15-9)$$

where E and ν are the Young's modulus and Poisson's ratio.

Backward integration: In backward integration, the trial elastic state is determined according to Eq. (3.15-3),

$${}^{t+\Delta t} \boldsymbol{\sigma}_*^E = \mathbf{C}_{ij}^E \left({}^{t+\Delta t} \boldsymbol{e}_j - {}^{t+\Delta t} \boldsymbol{e}_j^{th} - {}^t \boldsymbol{e}_j^{VP} \right) \quad (3.15-10)$$

The increments of stresses are obtained from

$$\Delta \boldsymbol{\sigma}_i^{(k)} = -\mathbf{C}_{ij}^E \Delta \boldsymbol{e}_j^{VP(k)} \quad (3.15-11)$$

and the $\Delta \boldsymbol{e}_j^{VP(k)}$ are given by Eq. (3.15-8).

If during subincrementation the effective stress $\bar{\sigma}^{(k)}$ is

$$\bar{\sigma}^{(k)} < \sigma_y \quad (3.15-12)$$

the subdivision loop is stopped by setting the parameter ISUBM to -1. Also the stresses are corrected as follows:

$$\sigma^{t+\Delta t} = -x \Delta \sigma^{(k)} + \bar{\sigma}^{(k)} \quad (3.15-13)$$

where $x = \frac{\sigma_y - \bar{\sigma}^{(k)}}{\Delta \bar{\sigma}^{(k)}}$ and the $\sigma^{t+\Delta t}$ are the final stresses at time $t+\Delta t$. Similarly, the increments of the viscoplastic strains are corrected using

$$\epsilon^{VP,t+\Delta t} = -x \Delta \epsilon^{VP(k)} + \epsilon^{VP(k)} \quad (3.15-14)$$

where $\epsilon^{VP,t+\Delta t}$ are the final viscoplastic strains at time $t+\Delta t$.

Constitutive matrix: For the calculation of the constitutive matrix, the constitutive relations (3.15-6) can be written as

$$\Delta \sigma_m = c_m (\Delta e_m - \Delta e^{\text{th}}) \quad (3.15-15)$$

$$\Delta s_i = \frac{1}{a_E} (\Delta e'_i - \Delta e_i^{VP}) \quad (3.15-16)$$

where

$$\begin{aligned} \Delta e_m &= \frac{1}{3} (\Delta e_1 + \Delta e_2 + \Delta e_4) \\ \Delta e'_3 &= \frac{1}{2} \Delta e_3 \end{aligned} \quad (3.15-17)$$

are increments of the mean and deviatoric strains, and c_m and a_E are elastic constants

$$\begin{aligned} c_m &= \frac{E}{1-2v} \\ a_E &= \frac{1+v}{E} \end{aligned} \quad (3.15-18)$$

The derivatives of the mean stress and of the deviatoric stress follow from Eqs. (3.15-15) and (3.15-16):

$$\begin{aligned}\frac{\partial \sigma_m}{\partial e_j} &= \frac{1}{3} c_m \quad j = 1, 2, 4 \\ \frac{\partial s_i}{\partial e_j} &= \frac{1}{a_E} \left(\frac{\partial e'_i}{\partial e_j} - \frac{\partial (\Delta e_i^{VP})}{\partial e_j} \right)\end{aligned}\quad (3.15-19)$$

The derivatives $\frac{\partial e'_i}{\partial e_j}$ can be written in matrix form as

$$\frac{\partial e'_i}{\partial e_j} = \frac{1}{3} \begin{bmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{bmatrix} \quad i, j = 1, 2, 4 \quad (3.15-20a)$$

$$\frac{\partial e'_3}{\partial e_3} = \frac{1}{2} \quad (3.15-20b)$$

$$\frac{\partial e'_i}{\partial e_j} = 0 \quad \text{for } i = 3 \text{ or } j = 3 \text{ and } i \neq j \quad (3.15-20c)$$

In the case of viscoplastic flow in the time step, an approximate expression for $\frac{\partial (\Delta e_i^{VP})}{\partial e_j}$ can be obtained from Eq. (3.15-7) as

$$\frac{\partial (\Delta e_i^{VP})}{\partial e_j} = c^{VP} \frac{\partial s_i}{\partial e_j} \quad (3.15-21)$$

where

$$c^{VP} = \frac{3}{2} \frac{\Delta \tau}{\bar{\sigma}} \gamma \left(\frac{\bar{\sigma}_{t+\Delta t}}{\sigma_y} - 1 \right) \quad (3.15-22)$$

It follows from Eqs. (3.15-21) and (3.15-19) that the derivatives

$$\frac{\partial s_i}{\partial e_j} \text{ are}$$

$$\frac{\partial s_i}{\partial e_j} = c' \frac{\partial e'_i}{\partial e_j} \quad (3.15-23)$$

where

$$c' = a_E + c^{VP} \quad (3.15-24)$$

The constitutive matrix \mathbf{C}^{VP} can be obtained using the relation

$$\begin{aligned} \sigma_i &= s_i + \sigma_m & i = 1, 2, 4 \\ \sigma_3 &= s_3 \end{aligned} \quad (3.15-25)$$

from which it follows that

$$C_{ij}^{VP} = \frac{\partial s_i}{\partial e_j} + \frac{\partial \sigma_m}{\partial e_j} \quad i = 1, 2, 4 \quad (3.15-26a)$$

$$C_{3j}^{VP} = \frac{\partial s_3}{\partial e_j} \quad (3.15-26b)$$

By substituting Eqs. (3.15-19) and (3.15-23) into (3.15-26) and using Eq. (3.15-20), the constitutive matrix \mathbf{C}^{VP} is obtained as

$$\mathbf{C}^{VP} = \begin{bmatrix} \frac{1}{3}(c_m + 2c') & \frac{1}{3}(c_m - c') & 0 & \frac{1}{3}(c_m - c') \\ & \frac{1}{3}(c_m + 2c') & 0 & 0 \\ \text{SYMM} & & \frac{1}{2}c' & 0 \\ & & & \frac{1}{3}(c_m + 2c') \end{bmatrix}$$

3.15.4.2 Plasticity model

- An isothermal von Mises plasticity model with isotropic hardening is given as an example of the use of the CUSER2 and CUSER3 subroutines.
- The implementation provided includes the MNO, TL and ULH formulations.
- The CUSER2 and CUSER3 subroutine coding can be found in files ovl30u_pl1.f and ovl40u_pl1.f respectively.
- The implementation is based on the equations given in Section 6.6.3 of Ref. KJB. The effective-stress-function algorithm is used for stress integration.

3.15.4.3 Thermo-plasticity and creep material model

- A model for thermoplasticity and creep is given as an example of the use of the CUSER2 and CUSER3 subroutines. The model includes coding for the MNO, TL and ULH formulations.
- The CUSER2 and CUSER3 subroutine coding can be found in files ovl30u_cn1.f and ovl40u_cn1.f respectively.
- The implementation is based on the equations given in Section 6.6.3 of ref. KJB.

The creep strain increments are defined using the Prandtl-Reuss equations:

$$d\varepsilon_{ij}^C = \gamma s_{ij}$$

where ε_{ij}^C are the creep strain components, s_{ij} are the deviatoric stress components and γ is the creep multiplier. Consequently, note that the effect of the hydrostatic pressure on the creep deformations is considered to be negligible.

The creep law is the eight-parameter creep law, i.e.:

$$\bar{\varepsilon}^C = f_1(\bar{\sigma}) f_2(\theta) f_3(t)$$

where $\bar{\varepsilon}^C$ is the effective creep strain, $\bar{\sigma}$ is the effective stress, θ is the temperature and t is the time, and

$$\begin{aligned} f_1(\bar{\sigma}) &= a_0 \bar{\sigma}^{a_1}; & f_2(\theta) &= \exp(-a_7 / (\theta + 273.16)) \\ f_3(t) &= t^{a_2} + a_3 t^{a_4} + a_5 t^{a_6} \end{aligned}$$

The same creep law also is used in the standard ADINA creep material model (see Section 3.6).

For multiaxial creep deformation, the calculation of the stresses is based on the use of an effective stress versus effective (creep) strain curve. The curve is assumed to be the same as the uniaxial stress-strain curve of the material.

In case of cyclic loading, the origin of the creep strain tensor depends on the stress tensor reversal and is determined using the ORNL rule (subroutines XCRCY2 and XCRCY3).

The effective-stress-function algorithm is used for stress integration (subroutines XEFSF2 and XEFSF3). Either the time hardening procedure or the strain hardening procedure can be used in the calculation of the creep multiplier (subroutines XGAMA2 and XGAMA3). A root-finding algorithm without acceleration (subroutine UBSECT) is used for computation of both the pseudo time (strain hardening) and the effective stress.

The tangent constitutive matrix $C_{ij}^{EC} = \frac{\sigma_i}{\varepsilon_j}$, where

σ_i are the stress components and ε_j are the total strain components, is calculated using an approximation based on the effective-stress-function algorithm (subroutines XELMA2 and

XELMA3).

3.15.4.4 Concrete material model, including creep effects

- A model for concrete is given as an example of the use of the CUSER2 and CUSER3 subroutines. The model includes coding for the MNO, TL and ULH formulations.
- The CUSER2 and CUSER3 subroutine coding can be found in files ovl30u_cn2.f and ovl40u_cn2.f respectively.
- This user-supplied material model has been developed for the analysis of concrete structures, in which creep deformations depend on the loading, on material ageing, on the temperature, and possibly on other material or load related parameters.
- The creep strain increments are defined by the Prandtl-Reuss equations

$$d\varepsilon_{ij}^C = \gamma s_{ij}$$

where ε_{ij}^C are the creep strain components, s_{ij} are the deviatoric stress components and γ is the creep multiplier. Consequently, note that the effect of the hydrostatic pressure on the creep deformations is considered to be negligible.

- A generalized creep law of the following form is assumed:

$$\bar{\varepsilon} = f_1(\bar{\sigma})f_2(\theta)f_3(t)f_4(x) + f_5(\bar{\sigma})f_6(\theta)f_7(t)f_8(x)$$

where $\bar{\varepsilon}$ is the effective creep strain, $\bar{\sigma}$ is the effective stress, θ is the temperature, t is the time, and x is a user-defined variable.

Appropriate coding for the creep member functions f_1 to f_8 is provided (subroutines XFNC2 and XFNC3). In addition, coding for the time derivatives of f_3 and f_7 is also provided (subroutines XDFNC2 and XDFNC3).

- Two different time variables are used (see Figure 3.15-5): t = time since concrete was cast; t_0 = time since ADINA analysis (and

presumably loading) started. The quantity $(t - t_0)$ represents the age of the concrete at the start of the finite element analysis. The input data array SCP is used to store $(t - t_0)$ (in SCP1 = SCP(1)). Note that $t = t_0 + \text{SCP1}$.

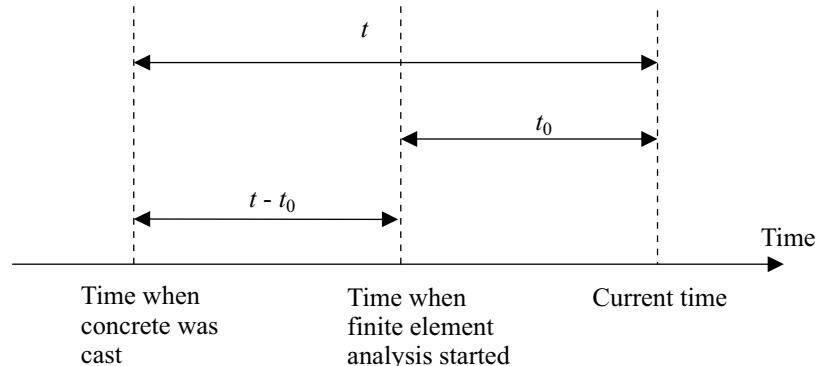


Figure 3.15-5: Time variables used for creep in concrete

- The variable SCP2 (= SCP(2)) is used to store the flag for choosing between the strain hardening ($\text{SCP2} = 1$) or the time hardening ($\text{SCP2} = 0$) procedure in the effective-stress-function algorithm.
- The elastic modulus is assumed to be time dependent and given by (see Figure 3.15-6):

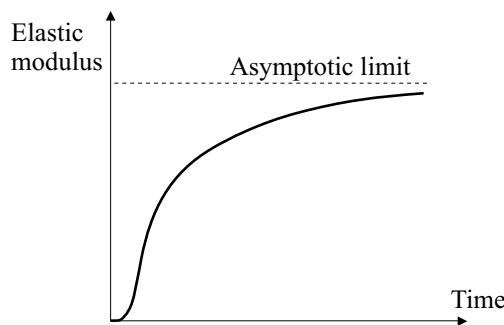


Figure 3.15-6: Time dependent elastic modulus for creep in concrete

$$E(t) = a_1 \exp \left[a_2 \left(1 - (a_3 / t)^{a_4} \right) \right]$$

- The creep law member functions are:

$$f_1(\bar{\sigma}) = \bar{\sigma}$$

$$f_2(\theta) = a_5 \phi^{a_6} + a_7 \phi^{a_8} \quad \text{with } \phi = \exp(a_9 \theta + a_{10})$$

$$f_3(t, t_0) = \frac{1}{E(t)} \left[\frac{a_{11} + t^{a_{12}}}{a_{13} + t^{a_{14}}} \right]^{a_{23}} \left[\frac{a_{15} + t_0^{a_{16}}}{a_{17} + t_0^{a_{18}}} \right]^{a_{24}} \left[\frac{a_{19} + (t - t_0)^{a_{20}}}{a_{21} + (t - t_0)^{a_{22}}} \right]^{a_{25}}$$

$$f_4(x) = 1$$

and

$$f_5(\bar{\sigma}) = \bar{\sigma}$$

$$f_6(\theta) = a_{26} (\theta + a_{27})^{a_{28}}$$

$$f_7(t) = \frac{1}{E(t)}$$

$$f_8(x) = 1$$

Note that the term of f_3 which depends only on $(t - t_0)$ could also be programmed as the member function $f_4(t - t_0)$.

- The creep constants are taken from array CTI as follows: $a_1 = \text{CTI}(1)$, $a_2 = \text{CTI}(2)$, One possible set of values is $a_1 = 2.0 \times 10^{10}$, $a_2 = 0.5$, $a_3 = 28$, $a_4 = 0.5$, $a_5 = 1.0$, $a_{12} = 0.6$, $a_{13} = 10$, $a_{14} = 0.6$, $a_{23} = 1.0$.
- In the case of multiaxial creep deformation, the calculation of the stresses is based on the use of an effective stress versus effective (creep) strain curve. The curve is assumed to be the same as the uniaxial stress-strain curve of the material.

In case of cyclic loading, the origin of the creep strain tensor depends on the stress tensor reversal and is determined using the ORNL rule (subroutines XCRCY2 and XCRCY3).

The effective-stress-function algorithm is used for stress integration (subroutines XEFSF2 and XEFSF3). The time hardening procedure or the strain hardening procedure can be used in the calculation of the creep multiplier, in case of variable stress and/or variable temperature conditions (subroutines XGAMA2 and XGAMA3). A root-finding algorithm without acceleration (subroutine UBSECT) is used for computation of both the pseudo time (strain hardening) and the effective stress.

The tangent constitutive matrix $C_{ij}^{EC} = \frac{t+dt}{\sigma_i} \frac{\sigma_i}{\varepsilon_j}$, where

σ_i are the stress components and ε_j are the total strain components, is calculated using an approximation based on the effective-stress-function algorithm (subroutines XELMA2 and XELMA3).

3.15.4.5 Viscoelastic material model

- A model for linear viscoelasticity is given as an example of the CUSER2 and CUSER3 subroutines. The model includes coding for the MNO, TL and ULH formulations.
- The CUSER2 and CUSER3 subroutine coding can be found in files ovl30u_vel.f and ovl40u_vel.f respectively.
- The implementation is fully described in Section 3.11.
- The material constants are specified in the Constant Material Properties area of the User-Supplied Material dialog box of the AUI as follows:

Constant, 1: η_G

Constant, 2: G_∞

Constant, 3: G_1

Constant, 4: β_1

...

Constant, $[2(\eta_G + 1) + 1]$: η_K

Constant, $[2(\eta_G + 1) + 2]$: K_∞
Constant, $[2(\eta_G + 1) + 3]$: K_1
Constant, $[2(\eta_G + 1) + 4]$: γ_1
...
Constant, $[2(\eta_G + \eta_K + 2) + 1]$: T_0
Constant, $[2(\eta_G + \eta_K + 2) + 2]$: α_0
Constant, $[2(\eta_G + \eta_K + 2) + 3]$: C_1
Constant, $[2(\eta_G + \eta_K + 2) + 4]$: C_2

and the total number of constants is $2(\eta_G + \eta_K + 2) + 4$. In addition, you must set the Length of Real Working Array to 70 or larger for materials used in 2-D solid elements and the Length of Real Working Array to 90 or larger for materials used in 3-D solid elements.

Using the AUI command-line interface command MATERIAL USER-SUPPLIED, enter the number of constants using parameter NCTD and the constants using parameters CTI1, CTI2,

3.15.4.6 Ramberg-Osgood material model with mixed hardening

- The Ramberg-Osgood material model with mixed hardening is provided as an example in the CUSER2 and CUSER3 subroutines. The model includes coding for the axisymmetric, plane strain, plane stress and 3-D elements and for the MNO, TL and ULH formulations.
- The CUSER2 and CUSER3 subroutine coding can be found in files ovl30u_pl2.f and ovl40u_pl2.f respectively.
- Additional files ovl30u_pl3.f and ovl40u_pl3.f are shown to use a more effective bisection algorithm, calling the ADINA subroutine BISECT.
- The Ramberg-Osgood stress-strain description of the yield curve is in the form

$${}^t\boldsymbol{\sigma} = {}^0\boldsymbol{\sigma}_y + C_y (\mathbf{e}^p)^n$$

where C_y and n are material constants and ${}^0\boldsymbol{\sigma}_y$ is the initial yield stress.

- The mixed hardening behaviour is described by

$$d\mathbf{e}^p = d\mathbf{e}^{p_i} + d\mathbf{e}^{p_k} = M d\mathbf{e}^p + (1 - M) d\mathbf{e}^p$$

where $d\mathbf{e}^{p_i}$ and $d\mathbf{e}^{p_k}$ are, respectively, the isotropic and kinematic contributions, and the yield function of the mixed hardening model is

$${}^t f_y = \frac{1}{2} ({}' \mathbf{s} - {}' \mathbf{a}) \cdot ({}' \mathbf{s} - {}' \mathbf{a}) - \frac{1}{3} {}^t \boldsymbol{\sigma}_y^2 = 0$$

where the back stress ${}' \mathbf{a}$ is evolved by

$$d\mathbf{a} = C_p (1 - M) d\mathbf{e}^p$$

C_p is Prager's hardening parameter and is calculated as

$$C_p = \frac{2}{3} \frac{E_p - M E_p^M}{1 - M}$$

where E_p^M is the derivative of the yield curve at the plastic strain $M \mathbf{e}^p$, E_p is the plastic modulus corresponding to \mathbf{e}^p , and M is the factor used in general mixed hardening ($0 < M < 1$) which can be a variable, expressed as

$$M = M_\infty + (M_0 - M_\infty) \exp(-\eta e^p)$$

- The effective-stress-function algorithm is used to calculate the plastic multiplier, see subroutines XEFSA and XEFSB in 2-D (subroutine CUSER2) and XEFSF3 in 3-D (subroutine CUSER3).

This page intentionally left blank.

4. Contact conditions

- Contact conditions can be specified in ADINA to model contact involving solid elements (2-D and 3-D solids) and/or structural elements (truss, beam, iso-beam or axisymmetric shell, plate, shell and pipe elements).
- Very general contact conditions are assumed:
 - ▶ The points of contact are assumed not known a priori.
 - ▶ Friction can be modeled according to various friction laws (only standard Coulomb friction for explicit dynamic analysis).
 - ▶ Both sticking and sliding can be modeled.
 - ▶ Repeated contact and separation between multiple bodies is permitted in any sequence.
 - ▶ Self-contact and double-sided contact are permitted.
 - ▶ Tied contact can be modeled.
 - ▶ Node-to-segment or node-to-node contact can be assumed.

Some of the contact algorithms used in ADINA are described in the following references:

ref. KJB
Section 6.7

- ref. Bathe, K.J. and Chaudhary, A., "A Solution Method for Planar and Axisymmetric Contact Problems," *Int. J. Num. Meth. in Eng.*, Vol. 21, pp. 65-88, 1985.
- ref. Eterovic, A. and Bathe, K.J., "On the Treatment of Inequality Constraints Arising From Contact Conditions in Finite Element Analysis," *J. Computers & Structures*, Vol. 40, No. 2, pp. 203-209, July 1991.
- ref. Pantuso, D., Bathe, K.J. and Bouzinov, P.A. "A Finite Element Procedure for the Analysis of Thermo-mechanical Solids in Contact," *J. Computers & Structures*, Vol. 75, No. 6, pp. 551-573, May 2000.

- Contact in ADINA is modeled using contact groups. Each contact group is composed of one or more contact surfaces. Contact pairs are then defined between contact surfaces. Contact segments are the building blocks for contact surfaces.
- Note that the presence of contact groups renders the analysis nonlinear even when no nonlinear element groups are defined.
- Most of the settings and tolerances needed for contact are provided in the contact group definition command (CROUP2 or CROUP3). Some contact settings however apply to all contact groups (such as contact convergence tolerances, suppression of contact oscillations). These settings are provided in the MASTER or CONTACT-CONTROL commands.

4.1 Overview

- Contact groups (and their contact surfaces) in ADINA can be either 2-D or 3-D. The contact surfaces should be defined as regions that are initially in contact or that are anticipated to come into contact during the solution.
 - ▶ 2-D contact surfaces are either axisymmetric or planar and must lie in the global YZ plane, with all X coordinates equal to zero. Typical two-dimensional contact surfaces are shown in Fig. 4.1-1(b).

The default contact surface representation only supports linear contact segments. Hence, contact segments on the faces of higher order elements are automatically divided into 2 or 3 linear segments. The new contact surface representation supports both linear and quadratic contact segments.

A 2-D contact surface is a closed surface if all the contact segments on the surface taken together form a closed path (see Fig. 4.1-1, contact surface 3). If the segments do not form a closed path, then the contact surface is open (see Fig. 4.1-1, contact surfaces 1 and 2).

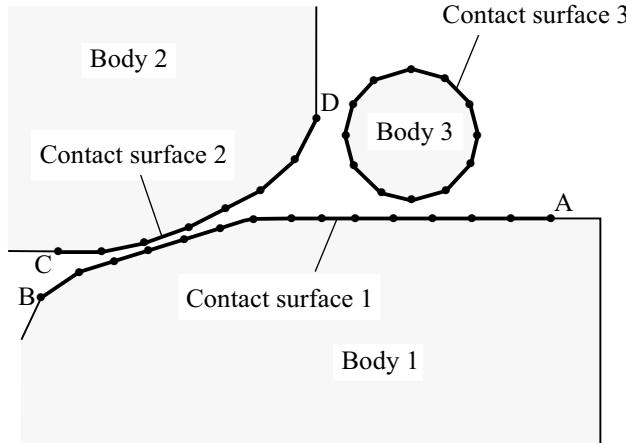


Figure 4.1-1: Typical 2-D contact surfaces

- A 3-D contact surface is made up of a group of 3-D contact segments (faces) either on solid elements, shell elements, plate elements, or attached to rigid nodes. See Fig. 4.1-2 for an illustration. The default contact surface representation supports only linear (and bilinear) segments. Hence, segments on the faces of higher order elements are automatically divided up into linear 3-node or 4-node faces. The new contact surface representation supports higher order (quadratic) contact segments.

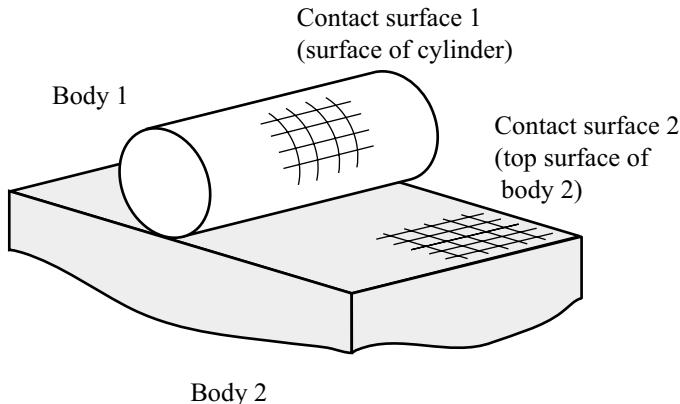


Figure 4.1-2: Typical contact surfaces and contact pair

- A contact pair consists of the two contact surfaces that may come into contact during the solution. One of the contact surfaces in the pair is selected to be the contactor surface and the other contact surface to be the target surface. In the case of self-contact, the same surface is selected to be both contactor and target.
- Within a contact pair, the nodes of the contactor surface are prevented from penetrating the segments of the target surface, and not vice versa.
- Target surfaces may be rigid, while contactor surfaces cannot be rigid. A contactor node should not also have all of its degrees of freedom dependent.
- In explicit dynamic analysis, both contactor and target surfaces can be rigid if the penalty algorithm is used. Otherwise, the same restriction mentioned above applies.
- Rigid surfaces have no underlying elements and therefore no flexibility apart from rigid body motions. All their nodal degrees of freedom must be either fixed, have enforced displacement, or be rigidly linked to a master node.
- Fig. 4.1-3 shows the effect of contactor and target selection on the different contact configurations.

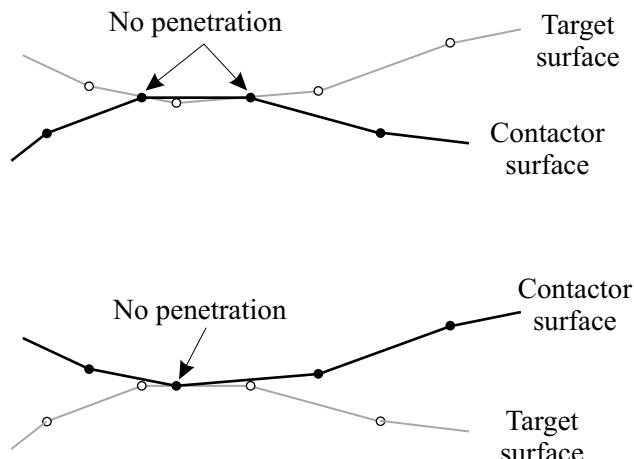


Figure 4.1-3: Contactor and target selection

- Self-contact is when a contact surface is expected to come into contact with itself during the solution.
- A default and a new contact surface representation are available in ADINA (set via the CS-TYPE parameter in the MASTER command). The new representation involves more accurate traction calculation, more detailed contact search and more accurate constraint enforcement for quadratic contact segments. The new contact surfaces are currently restricted to work with only the following options:
 - Constraint-function algorithm
 - All 3D and shell element types except for 16-node shells
 - All 2D element types except for 4-node beams
 - No consistent contact stiffness
 - Contact surface offsets = 0
 - Single-sided contact only
 - Frictionless and regular Coulomb friction (no user-supplied friction)
 - Eliminate or ignore initial penetrations (no discarding of initial penetrations)
 - Regular contact surface definitions (no contact body or contact point definitions, no analytical rigid targets)
 - Static, implicit dynamic and frequency analysis
- The direction of a contact surface is marked by a normal vector \mathbf{n} pointing towards the interior of the contact surface as shown in Fig. 4.1-4 and Fig. 4.1-5.

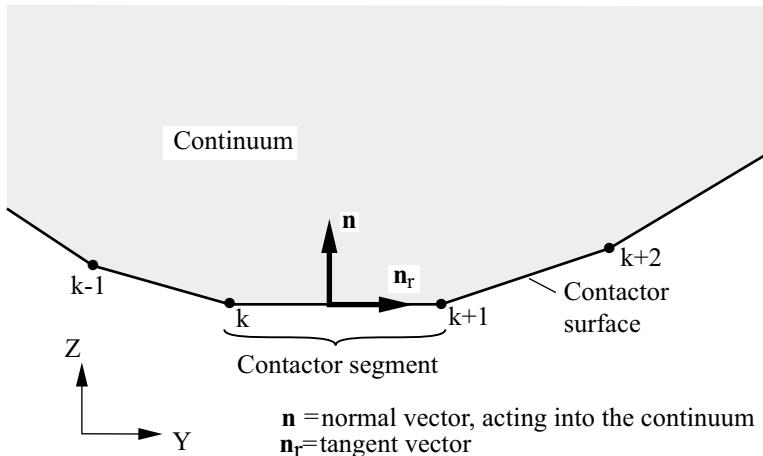


Figure 4.1-4: Normal vector of a contact segment in 2-D analysis

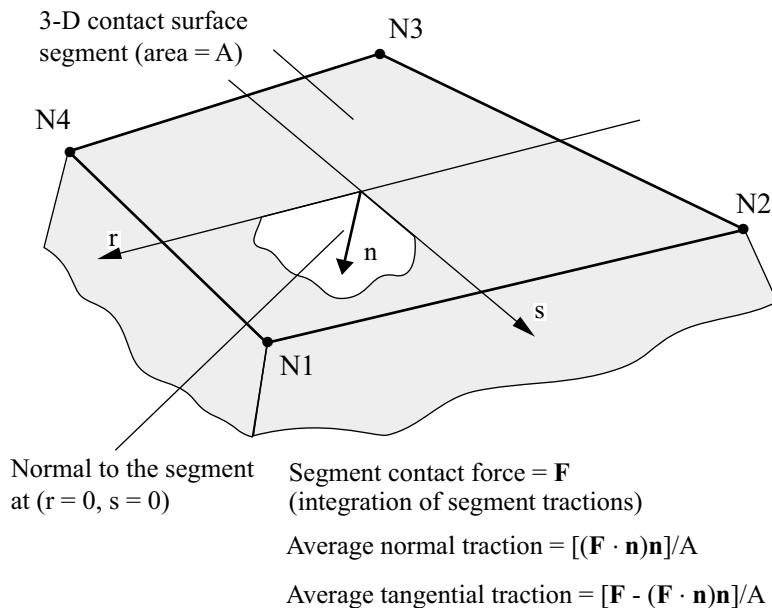


Figure 4.1-5: Normal vector of a contact segment in 3-D analysis

- For single-sided contact (see Fig. 4.1-6), one side of the contact surface is assumed to be internal and the other side to be external. Any contactor node within the internal side of a target surface is

assumed to be penetrating and will be moved back to the surface. This single-sided option is ideal for contact surfaces on the faces of solid elements since in that case it is clear that one side is internal to the solid while the other is external. In this case, the external side can usually be predicted from the geometry. This option is also useful for shells when it is known that contact will definitely occur from one direction. In this case, however, the program cannot intuitively predict the internal side of the contact surface.

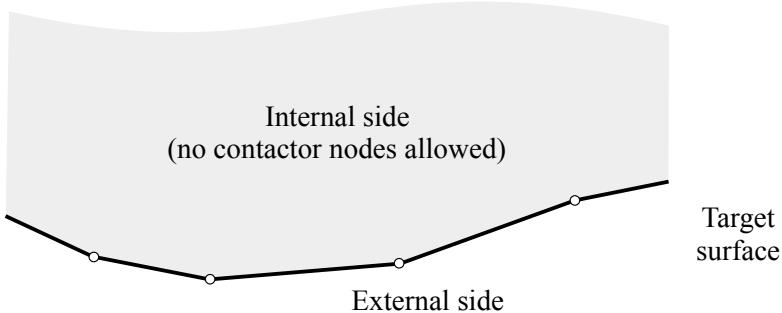


Figure 4.1-6: Single-sided contact surface

- In double-sided contact (see Fig. 4.1-7), there are no internal or external sides. The contactor surface nodes in this case are prevented from crossing from one side of the target contact surface to the other during solution. This option is more common for shell-based contact surfaces. If a contactor node is one side of the target surface at time t , it will remain on the same side at time $t + \Delta t$.

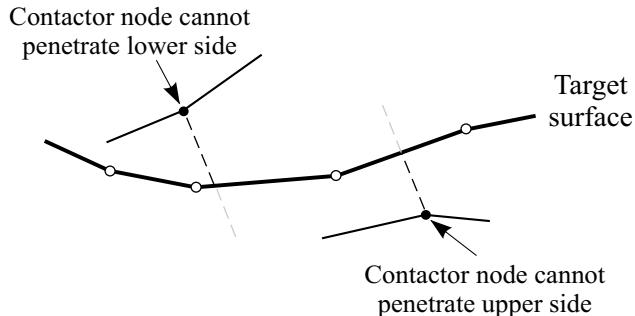


Figure 4.1-7: Double-sided contact

- When the tied contact feature is selected for a contact group, ADINA performs an initial contact check at the start of the analysis. All contactor nodes that are found to be in contact are permanently attached to their respective target segments. This is conceptually similar to using rigid links or constraints to attach the node to the target surface. The main difference is that the coefficients for the rigid elements are automatically determined by the program and they are only applied for the nodes that are initially in contact. The basic idea is illustrated in Fig. 4.1-8.

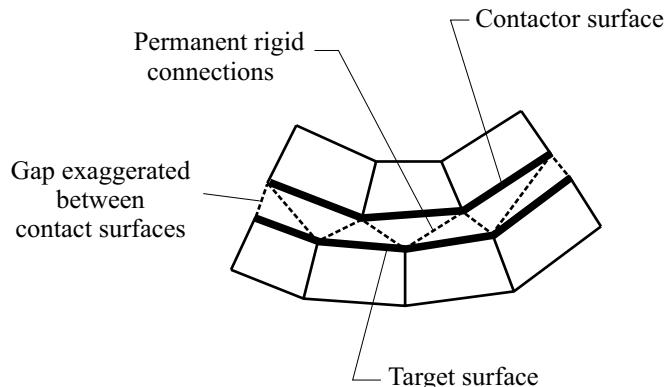


Figure 4.1-8: Tied contact option

Tied contact is not "real" contact because there can be tension between tied contact surfaces. Also no sliding can occur between tied contact surfaces.

The tied contact option can be used to glue two surfaces together if they have incompatible meshes. However, the surface glueing feature described in Section 5.16 produces more accurate "glue" equations.

If the contact surfaces initially overlap, they are not pushed back to eliminate the overlap.

The tied contact constraint equations are computed based on the initial nodal positions only. The constraints generated in tied contact are not updated during the analysis. Hence, they may be less accurate if the bodies experience large rotations.

- When node-node contact is used, each node of the contactor surface is paired with a node on the target contact surface (a node-

node contact pair). A node on the contactor surface can come into contact with the infinite plane passing through the paired node on the target surface; the infinite plane is perpendicular to the normal direction indicating the direction within the target body (Fig. 4.1-9). This normal direction, and hence the infinite plane, is constant during the response. The normal directions can be based on the geometry of the target segments or on the vector connecting each contactor and target nodes.

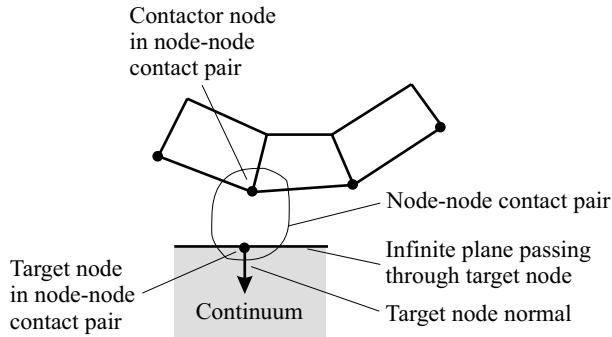


Figure 4.1-9: Node-to-node contact

- The user can request output of nodal contact forces and/or contact tractions. Using the default contact surface representation gives output of segment tractions. The new contact surface representation outputs nodal tractions as well as gap and contact status information.
- The normal contact conditions can ideally be expressed as

*ref. KJB
Section 6.7.2*

$$g \geq 0; \quad \lambda \geq 0; \quad g\lambda = 0 \quad (4.1-1)$$

where g is a gap, and λ is the normal contact force. Different algorithms may vary in the way they impose this condition.

- For friction, a nondimensional friction variable τ can be defined as

$$\tau = \frac{F_T}{\mu\lambda} \quad (4.1-2)$$

where F_T is the tangential force and λ is the normal contact force.

- The standard Coulomb friction condition can be expressed as

$$\begin{array}{ll} |\tau| \leq 1 & \\ \text{and} & |\tau| < 1 \text{ implies } \dot{u} = 0 \\ \text{while} & |\tau| = 1 \text{ implies } \text{sign}(\dot{u}) = \text{sign}(\tau) \end{array} \quad (4.1-3)$$

where \dot{u} is the sliding velocity.

- In static analysis, the sliding velocity is calculated by dividing the incremental sliding displacement by the time increment. Hence, time is not a dummy variable in static frictional contact problems.
- When (Coulomb) friction is used, the friction coefficient can be constant or calculated from one of several predefined friction laws.
- The possible states of the contactor nodes and/or segments are

No contact: the gap between the contactor node and target segment is open.

Sliding: the gap between the contactor node and the target segment is closed; a compression force is acting onto the contactor node and the node kinematically slides along the target segments (either due to frictionless contact to a frictional restrictive force less than the limit Coulomb force).

Sticking: as long as the tangential force on the contactor node that initiates sliding is less than the frictional capacity (equal to the normal force times the Coulomb friction coefficient), the contactor node sticks to the target segment.

4.2 Contact algorithms for implicit analysis

- ADINA offers three contact solution algorithms (set via the ALGORITHM flag):
 - ▶ Constraint-function method,

- ▶ Lagrange multiplier (segment) method, or
- ▶ Rigid target method
- Each contact group must belong to one of these three contact algorithms. However, different contact groups can use different algorithms.
- All 3 contact algorithms can be used with or without friction.

4.2.1 Constraint-function method

- In this algorithm, constraint functions are used to enforce the no-penetration and the frictional contact conditions.
- The inequality constraints are replaced by the following normal constraint function:

$$w(g, \lambda) = \frac{g + \lambda}{2} - \sqrt{\left(\frac{g - \lambda}{2}\right)^2 + \varepsilon_N}$$

where ε_N is a small user-defined parameter. The function is shown in Fig. 4.2-1. It involves no inequalities, and is smooth and differentiable.

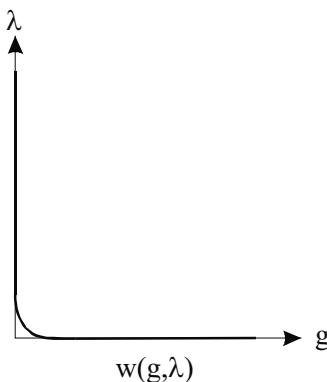


Figure 4.2-1 Constraint function for normal contact

- The constraint function method regularizes the stick-slip transition of Eq. (4.1-3). This results in a smooth transition from stick to slip and vice versa, and it also results in a smooth differentiable friction law that is less likely to cause convergence difficulties.

The frictional constraint function $v(\dot{u}, \tau)$ is defined by

$$\tau + v - \frac{2}{\pi} \arctan\left(\frac{\dot{u} - v}{\varepsilon_T}\right) = 0$$

Here ε_T is a small parameter which can provide some "elasticity" to the Coulomb friction law as shown in Fig. 4.2-2.

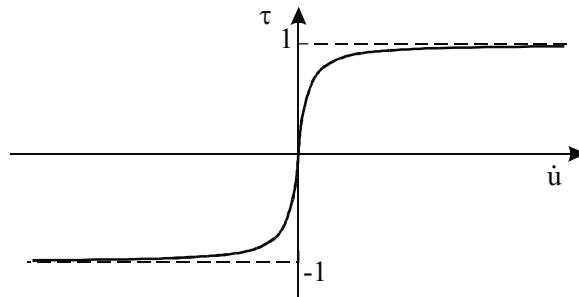


Figure 4.2-2: Constraint functions for tangential contact $v(\dot{u}, \tau)$

4.2.2 Lagrange multiplier (segment) method

- In this method, Lagrange multipliers are used to enforce the contact conditions of Eq. (4.1-1). The kinematic conditions are enforced at the contactor nodes, and the frictional conditions are enforced over the contact segments.
- This method involves distinct sticking and sliding states. It also calculates this state for each contactor node based on the contact forces on the target segment.
- This method should not be used with a force or energy/force convergence criterion.

4.2.3 Rigid target method

- This is a significantly simplified contact algorithm intended primarily for metal forming applications.
- The target surface must be rigid, and contact is enforced using a penalty-based approach. The normal modulus is set via the NORMAL-STIFFNESS parameter. In addition, two major simplifying assumptions are used which facilitate convergence, but at the same time reduce the acceptable time step size. These simplifications are:
 - ▶ New contactor nodes can only come in contact at the first iteration of a load step. Contact is not enforced at any contactor node that comes into contact after the first iteration. These nodes may therefore penetrate the target.
 - ▶ Contactor nodes currently in contact can only be released also in the first iteration of a load step. Nodes trying to be released at intermediate iterations will be kept in tensile contact.
- These simplifications require that a small time step be used with the Rigid Target contact algorithm. Also the resulting penetrations and contact forces must be checked to ensure that the solution does not excessively violate the original contact conditions.
- If a tensile force (from the second simplification) is higher than a user-specified tolerance (RESIDUAL-FORCE parameter), the time step is reduced and the load step is repeated. Similarly, if the sum of all tensile contact forces is greater than a user-specified tolerance (LIMIT-FORCE parameter) the time step is reduced and the load step is repeated.

The manner in which the time step is reduced due to tensile forces is governed by the RT-SUBDIVIDE parameter in the MASTER command. It can be based on the magnitude of the tensile contact forces, or on the general subdivision parameters set for Automatic Time Stepping (ATS) in Section 7.2.

- Rigid target penetration checking can be selected (via the RTP-CHECK parameter) to ensure that the penetrations (due to the first

simplification) do not exceed a preset level. If the penetration exceeds the tolerance, the time step is reduced by the program and the load step is repeated.

- Friction in the Rigid Target algorithm is also treated using a penalty-based approach. The tangential modulus is set via the TANGENTIAL-STIFFNESS parameter.

4.2.4 Selection of contact algorithm

- Our experience is that in most frictionless contact problems the constraint function method is more effective than the segment method. The constraint function method is the default.
- For friction contact dominated contact problems, the performance of the constraint function and segment methods are comparable. The default is still the constraint function method.
- The rigid target method requires much smaller time steps and has several input parameters which need to be properly adjusted. It should only be used for metal forming type problems.
- Note that the target surface can be rigid in all three contact algorithms. The presence of a rigid target does not mean that the rigid target algorithm should be used.

4.3 Contact algorithms for explicit analysis

- ADINA offers three contact solution algorithms for explicit analysis:
 - ▶ Kinematic constraint method,
 - ▶ Penalty method, or
 - ▶ Rigid target method
- Each contact set must belong to one of these three contact algorithms. However, different contact sets can use different algorithms.

- All these contact algorithms can be used with or without friction.

4.3.1 Kinematic constraint method

- This is the default explicit contact algorithm.
- A predictor step is first done without applying contact constraints or forces. Then displacements are evaluated and penetration is detected and corrected. The exact correction of displacements requires the solution of a non-diagonal system of equations. Instead, a good approximation is done. In this case, for each penetrating contactor node, a penetration force

$$\mathbf{F}_C^N = M_C \mathbf{a}_C^{N*} = M_C \frac{\delta_N}{\Delta t^2} \mathbf{N}$$

is calculated. This is the force required to remove the penetration at the contactor node. However, not all the penetration will be removed by moving the contactor. The target will get some motion depending on its mass relative to the contactor and how many contactor nodes are touching it. So, the \mathbf{F}_C^N force above is projected to the target segment nodes:

$$\mathbf{F}_{T_i}^N = N_i \mathbf{F}_C^N$$

where N_i is the shape function relating the contactor displacement to that of each target node. Similarly, the mass of the contactor node is projected to the target in the same way:

$$M_{T_i} = N_i M_C$$

and this mass is added to that of the target node itself. Then the acceleration of the target node is determined as

$$\mathbf{a}_T^N (M_T + \sum M_{T_i}) = \sum \mathbf{F}_{T_i}^N$$

This correction is then used to update the target displacements. The contactor acceleration is

$$\mathbf{a}_C^N = \mathbf{a}_C^{N*} - \sum \mathbf{a}_T^N N_i$$

- For friction, a similar approach is used. A correction force is calculated

$$\mathbf{F}_C^{T*} = M_C \frac{\mathbf{v}_T}{\Delta t}$$

where \mathbf{v}_T is the tangential sliding velocity. However, this force cannot exceed the limit force based on the normal force and the coefficient of friction

$$\mathbf{F}_C^T = \min(\mu \mathbf{F}_C^N, \mathbf{F}_C^{T*})$$

The rest of the procedure is very similar to the case of normal contact. The form of the equations is different if there is damping, and is also different if the previous and current time steps are not the same.

- A modification is also required for rigid targets, which are common in contact. The form of the equations in this case depends on whether the rigid target has natural or essential boundary conditions.

4.3.2 Penalty method

- In this algorithm, contact conditions are imposed by penalizing the inter-penetration between contacting surfaces. When a penetration is detected, a normal force of

$$\mathbf{F} = A \quad \mathbf{P} = A \quad \mathbf{N} \quad (K_N \delta_N + K_D \dot{\delta}_N)$$

is applied to the contactor node, where K_N is the normal stiffness, K_D is a normal rate stiffness, δ_N is the penetration, $\dot{\delta}_N$ is the penetration rate, \mathbf{N} is the normal vector pointing towards the contactor, A is the contact area and \mathbf{P} is the normal contact traction. An opposing force is distributed to the target nodes.

- Similarly, in the presence of friction, the relative sliding

velocity between the two bodies is penalized as follows:

$$\mathbf{F} = A \ K_T \mathbf{v}_T$$

where \mathbf{v}_T is the tangential sliding velocity.

- The normal and tangential penalty stiffnesses K_N and K_T can be selected by the user, or determined automatically by the program. The penalty rate stiffness K_D can be explicitly selected by the user, or determined by the program as a ratio of critical damping for the contact node.
- When penalty stiffnesses are automatically determined they are chosen based on the masses of the contactor nodes and the time step. They are selected such that they have a minimal effect on the existing time step.

Note that unduly small penalty stiffnesses will lead to excessive penetrations, and while unduly large penalty stiffnesses will lead to excessive oscillations or unstable time integration.

4.3.3 Rigid target method

- This algorithm should only be used when the analysis might involve restarts to or from an implicit analysis involving the implicit rigid target contact algorithm of Section 4.2.3. The normal contact conditions are imposed exactly in this case, using an algorithm similar to the kinematic constraint method. Friction conditions are imposed using the same penalty approach as in the implicit rigid target algorithm.
- The stable time step size is affected by the frictional stiffness parameter. However, this effect is not taken into account in the calculation of the stable time step. The user should select an appropriate value of the penalty stiffness.

4.3.4 Selection of contact algorithm for explicit analysis

- The kinematic constraint method is the default.
- The penalty method is the simplest and fastest of the explicit

contact algorithms. It can also handle rigid contactor and target surfaces. It also allows a contactor node to be in contact with multiple targets simultaneously.

- The main disadvantage of the penalty method is that contact conditions are not exactly satisfied and it usually shows oscillations in contact forces. These oscillations can usually be removed by using penalty damping. It is also sensitive to the choice of the penalty stiffness. If that stiffness is too large it leads to instability and oscillations, and if it is too small it leads to excessive penetrations.
- The default penalty stiffness selected by the program is, in most cases, a suitable compromise.
- The explicit rigid target algorithm should only be used if the purpose is there are restarts to or from implicit analyses that use the implicit rigid target algorithm.

4.4 Contact group properties

This section describes the main options available to contact groups.

- **Contact surface offsets**

Penetration of a contact surface occurs when the plane or line defined by the contact segment nodes is penetrated. However, an offset distance can be specified which causes the actual contact surface to be offset from the plane defined by the contact surface nodes. In the case of double-sided contact, the offset creates two separate surfaces above and below the reference surface. Note that if the contact surface is on a shell (in rigid target method only), then half the shell thickness can automatically be used as the offset. Fig. 4.4-1 shows the possibilities for single and double-sided contact. Note that the offset distance should be small compared to the contact surface length. Offsets for a whole contact group are specified in the CGROUP commands. Offsets for a specific contact surface can also be defined. If one of the contact surfaces has a defined offset, it will overwrite the contact group offset.

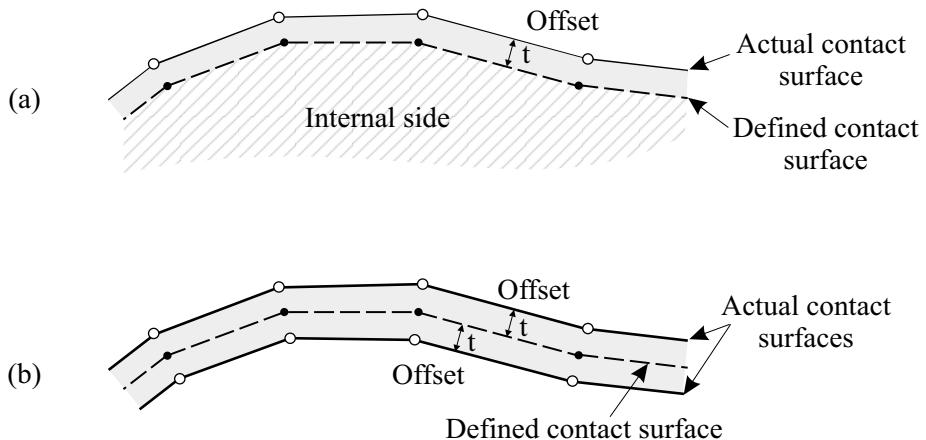


Figure 4.4-1: Contact surface offsets for: (a) single-sided contact, and (b) double-sided contact

- **Continuous normals (not available in explicit analysis)**

The normal direction to a contact segment will in general not be continuous between segments as illustrated in Fig. 4.4-2. This sometimes causes convergence difficulties due to the non-unique normals at nodes and segment edges. The continuous normals feature first calculates nodal normals as averages of all the normals from the attached segments, and then interpolates these nodal normals across the segment. This leads to a uniformly varying normal direction. Continuous normals is the default setting (Fig. 4.4-2(b)).

In modeling target surfaces with sharp corners, either use discontinuous normal vectors, or use small segments near the corners, in order that the normal vectors for segments near the corners be computed correctly. See Section 4.8.2 for modeling tips related to this feature.

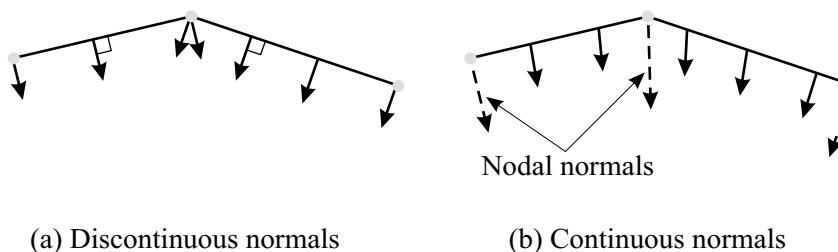


Figure 4.4-2: Contact surface normals

- **Contact surface depth**

By default, the contact region extends for an infinite distance below the contact surface (for single-sided contact). However, a contact surface depth can be defined (by setting the DEPTH parameter), below which the contact surface is no longer active. The default DEPTH=0.0 results in an infinite contact depth extension. Fig. 4.4-3 shows some of the possibilities.

Contact surface is used in self-contact.

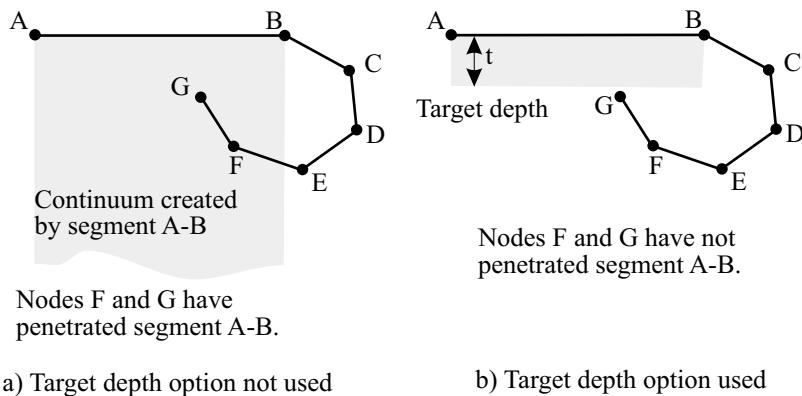


Figure 4.4-3: Contact surface depth

- **Initial penetration**

The treatment of initial penetrations in ADINA is governed by the INITIAL-PENETRATION parameter. By default, if there is initial overlap (penetration) between a contact node and a target segment in the first solution step, ADINA attempts to eliminate the overlap. ADINA can eliminate the overlap at the first step or over a user-specified time using the TIME-PENETRATION parameter. This

feature is useful if the initial penetrations are too large to be eliminated in a single step.

The program can also calculate initial penetrations at the start of solution and ignore them in future steps. In this case the program does not detect penetration for a contactor node if the amount of penetration is less than or equal to the recorded amount. Fig. 4.4-4 shows some of the possibilities. See Section 4.8.2 for modeling tips related to this feature.

- **Contact surface extension**

The target surface can be enlarged beyond its geometric bounds, so that contactor nodes that slip outside the target can still be considered in contact (via the CS-EXTENSION parameter). This feature is useful where the edge of the contactor and target surfaces coincide, as shown in Fig. 4.4-5.

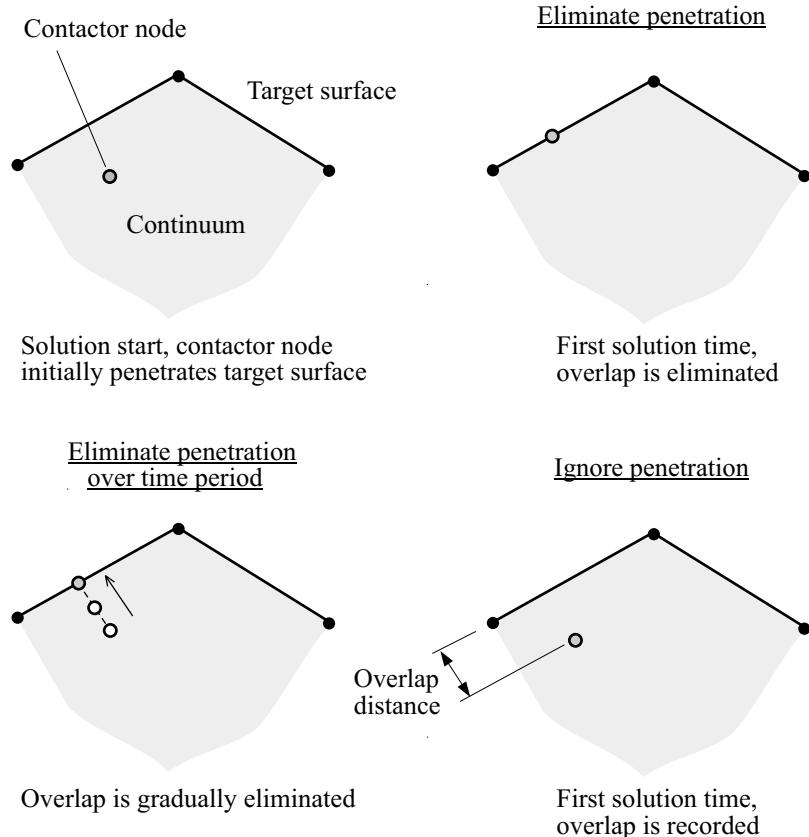


Figure 4.4-4: Initial penetration options

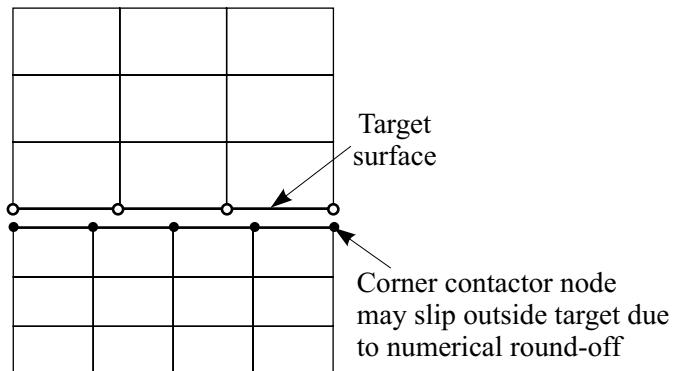


Figure 4.4-5: Contact requiring contact surface extension

- **Contact surface compliance***

Contact surface compliance is set via the CFACTOR1 parameter and is only available with the constraint function algorithm. Contact surfaces are commonly assumed to be rigid meaning that no interpenetration is allowed. In this case, the contact surface compliance is set to 0.0. However, this feature can be used to simulate soft or compliant surfaces. The amount of allowed interpenetration between the contacting surfaces in this case is

$$\text{penetration} = \varepsilon_p \times \text{normal contact pressure} \quad (4.4-1)$$

The constraint function in the presence of a compliance factor is modified as shown in Fig. 4.4-6.

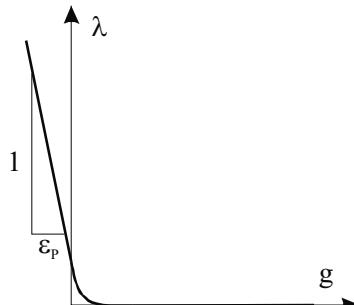


Figure 4.4-6: Constraint function for compliant contact

- **Consistent contact stiffness***

The consistent contact stiffness feature is set via the CONSISTENT-STIFF flag. Changes in the direction of the contact normal provide an additional contribution to the stiffness matrix that is proportional to the value of the contact force and the change in the normal direction. Therefore, higher convergence rates (closer to quadratic) can sometimes be obtained by selecting the consistent contact stiffness option which accounts for these additional stiffness contributions. This results, however, in an increase in the size of the stiffness matrix which is detrimental for large problems. This option is more beneficial when discontinuous contact normals are selected.

*not available in explicit analysis

- **Contact birth/death**

The contact birth feature activates a contact group at a specific time, while the contact death feature disables a contact group at a specific time. They are set via the TBIRTH and TDEATH parameters. A 0.0 birth time means that the contact group starts active at the beginning of the analysis, and a death time less than or equal to the birth time means that the contact group does not die.

- **Contact slip***

Contact slip allows a tangential or slip velocity to be added component between contact surfaces. With this feature, you can model out-of-plane slip in 2-D axisymmetric analysis. This feature is considered to be a load added to the model. See Section 5.11 for further details.

- **Contact surface output**

You can request that ADINA save contact tractions and/or forces. Different outputs are obtained for the old and new contact surface representations.

Traction:

Old contact surface representation

Traction output is available only on contactor surfaces, and only if the surface is segment-based (not nodal based). Each contactor contact segment outputs, at its center, the following output information to the porthole file. This information is accessible in the AUI using the given variable names.

2-D contact: SEGMENT_LENGTH, NORMAL-Y, NORMAL-Z,
NORMAL_TRACTION, TANGENTIAL_TRACTION

3-D contact: SEGMENT_AREA, NORMAL-X, NORMAL-Y,
NORMAL-Z, NORMAL_TRACTION-X,
NORMAL_TRACTION-Y, NORMAL_TRACTION-Z,
TANGENTIAL_TRACTION-X, TANGENTIAL_TRACTION-Y,
TANGENTIAL_TRACTION-Z

*not available in explicit analysis

In addition, the following information is output for solitary nodes:
SOLITARY_CONTACT_FORCE-X (3-D contact only),
SOLITARY_CONTACT_FORCE-Y,
SOLITARY_CONTACT_FORCE-Z,
SOLITARY_CONTACT_FORCE-A (3-D contact only),
SOLITARY_CONTACT_FORCE-B,
SOLITARY_CONTACT_FORCE-C.

New contact surface representation

Only nodal contact quantities are output for the new contact surface representation. Output is restricted to the contactor nodes. The output data is available to the AUI using the variable names given below. Note that “traction” output in this case also produces some non-traction surface data such as contact status, gap and slip velocity.

```
NODAL_CONTACT_GAP  
NODAL_CONTACT_STATUS (see note below)  
NODAL_NORMAL_TRACTION (magnitude of normal traction)  
NODAL_NORMAL_TRACTION-{XYZ}  
NODAL_TANGENTIAL_TRACTION (magnitude of tangential traction)  
NODAL_TANGENTIAL_TRACTION-{XYZ}  
NODAL_SLIP_VELOCITY (magnitude of slip velocity)  
NODAL_SLIP_VELOCITY-{XYZ}
```

All of these variables are output at the nodes. As a convenience, band plots still allow the variable names

```
NORMAL_TRACTION  
NORMAL_TRACTION-{XYZ}  
TANGENTIAL_TRACTION  
TANGENTIAL_TRACTION-{XYZ}
```

See Section 11.1.1 for the definitions of those variables that are not self-explanatory.

Forces:

Force output is available for all contact nodes under all contact conditions. The AUI variables are: CONTACT_FORCE-X (3-D contact only), CONTACT_FORCE-Y, CONTACT_FORCE-Z,

CONTACT_FORCE-A (3-D contact only), CONTACT_FORCE-B, CONTACT_FORCE-C. These forces are sometimes called "consistent contact forces".

4.5 Friction

ADINA has a general Coulomb type friction model, where the coefficient of friction μ can be a constant or calculated based on several user-defined friction laws. Explicit analysis however, only supports standard Coulomb friction.

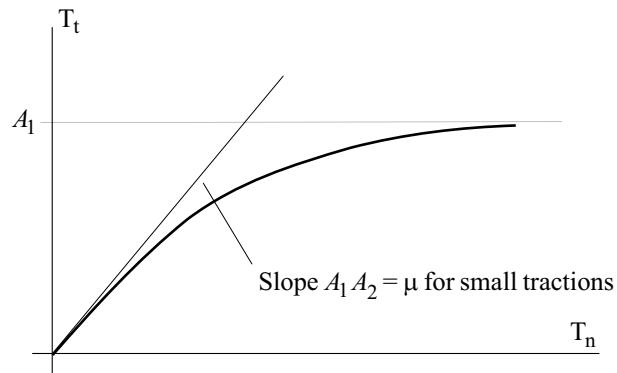
4.5.1 Basic friction models

By default, a constant coefficient friction is used. The Rigid Target algorithm can only use a constant Coulomb friction coefficient.

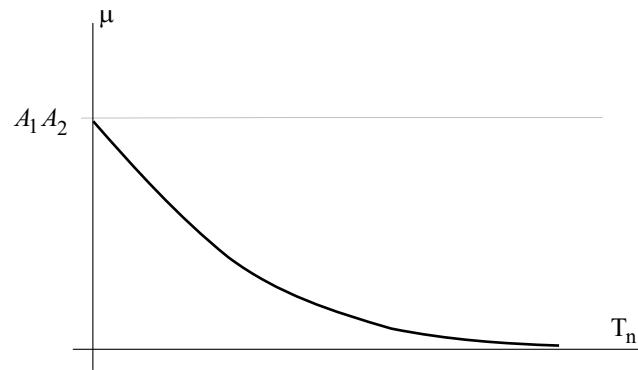
Two other basic models in which the coefficient of friction depends on the contact tractions are

Model 1: $T_t = A_1 \left(1 - \exp(-A_2 T_n)\right)$, where A_1, A_2 are specified constants. This implies that the instantaneous coefficient of friction is $\mu = \frac{1 - \exp(-A_2 T_n)}{T_n / A_1}$. Model 1 is graphically shown in Fig. 4.5-1.

Model 2: $T_t = [A_2 + (A_1 - A_2) \exp(-A_3 T_n)] T_n$, where A_1, A_2, A_3 are specified constants. This implies that the instantaneous coefficient of friction is $\mu = A_2 + (A_1 - A_2) \exp(-A_3 T_n)$. Model 2 is graphically shown in Fig. 4.5-2.

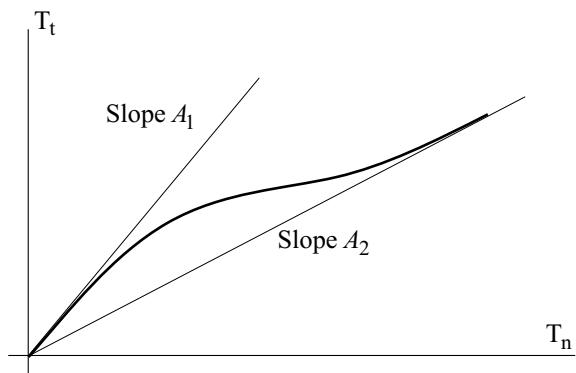


a) Maximum tangential traction vs normal traction

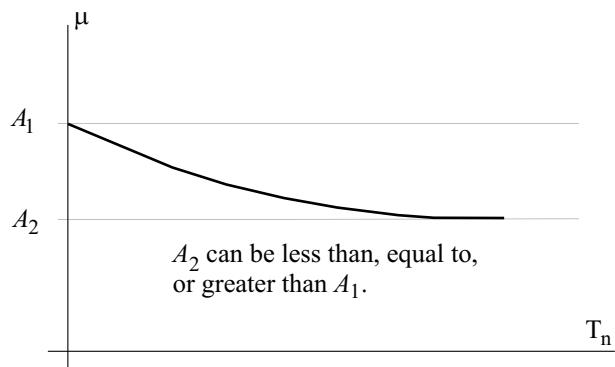


b) Coefficient of friction vs normal traction

Figure 4.5-1: Frictional contact using model 1



a) Maximum tangential traction vs normal traction



b) Coefficient of friction vs normal traction

Figure 4.5-2: Frictional contact using model 2

4.5.2 User-supplied friction models (not available in explicit analysis)

User supplied friction models are provided in subroutine FUSER. They offer you more flexibility in implementing your own friction models. They can be used with 2D and 3D contact surfaces, and can be used with both segment and constraint-function methods. They can also be used with the default node-to-segment contact and with node-node contact. They cannot be used with the rigid target contact algorithm.

Some algorithmic considerations

- The subroutine FUSER in file ovlusr.f accepts two user-specified arrays of constants, an integer array and a real array. In the current implementation, the first integer constant I_1 is used to select a friction law. The real constants starting with A_1 are used as parameters for the specified friction law.
- The input parameters passed to subroutine FUSER are: contact traction T_n , consistent contact force F_n , normal direction, direction of sliding \mathbf{v} , other tangential direction, original and current coordinates of contactor nodes, normal constraint-function parameter ε_N , friction constraint-function parameter ε_T , magnitude of relative sliding velocity \dot{u} , frictional force, user-selected parameters (real and integer), number of user-selected parameters, contact pair number, time, load step number, iteration number, initial temperature of contactor, current temperature of contactor θ , flag to show if temperatures are present, algorithm used (constraint function or segment method), contact type (node-to-segment, node-to-node) and contactor node number.
- The output parameters from subroutine FUSER are the coefficient of friction SCF, an error flag IERR, and an error string ERMES.
- Not all input parameters are available with all contact algorithms, for example contact tractions are not available with node-node contact. More details are provided in the documentation of the FUSER subroutine.
- Several friction laws are already implemented in subroutine FUSER.

Existing user-supplied friction models

- Constant coefficient of friction ($I_1 = 1$)

$$\mu = A_1$$

- Different static and dynamic friction coefficients ($I_1 = 4$)

$$\mu = \begin{cases} A_1 & \text{if } \dot{u} \leq A_3 \\ A_2 & \text{if } \dot{u} > A_3 \end{cases}$$

- Friction coefficient varying with sliding velocity ($I_1 = 5$)

$$\mu = \begin{cases} A_1 + \frac{\dot{u}}{A_2}(A_3 - A_1) & \text{if } \dot{u} < A_2 \\ A_3 & \text{if } \dot{u} > A_2 \end{cases}$$

- Anisotropic friction model ($I_1 = 6$)

$$\mu = \begin{cases} \sqrt{(A_1 \mathbf{v}_{(1)})^2 + (A_2 \mathbf{v}_{(2)})^2 + (A_3 \mathbf{v}_{(3)})^2} & \text{if } \dot{u} > A_5 \\ A_4 & \text{if } \dot{u} \leq A_5 \end{cases}$$

where $\mathbf{v}_{(1)}$, $\mathbf{v}_{(2)}$ and $\mathbf{v}_{(3)}$ are the x, y and z components of the sliding direction.

- Friction coefficient varying with consistent contact force ($I_1 = 7$)

$$\mu = A_1 + A_2 F_n, \quad 0 \leq \mu \leq 1$$

- Time varying friction model ($I_1 = 8$)

$$\mu = \begin{cases} A_1 + \frac{t}{A_2}(A_3 - A_1) & \text{if } t < A_2 \\ A_3 & \text{if } t > A_2 \end{cases}$$

- Coordinate-dependent friction model ($I_1 = 9$)

$$\mu = \begin{cases} A_1 + A_3 \mathbf{x}_{(2)} + A_4 \mathbf{x}_{(3)} & \text{in 2D} \\ A_1 + A_3 \mathbf{x}_{(1)} + A_4 \mathbf{x}_{(2)} + A_5 \mathbf{x}_{(3)} & \text{in 3D} \end{cases}, \quad 0 \leq \mu \leq A_2$$

- Temperature-dependent friction model ($I_1=10$)

$$\mu = \begin{cases} A_2 & \text{if } \theta < A_1 \\ A_2 \frac{A_3 - \theta}{A_3 - A_1} + A_4 \frac{\theta - A_1}{A_3 - A_1} & \text{if } \theta < A_3 \\ A_4 & \text{if } \theta \geq A_3 \end{cases}$$

- Friction model 1a ($I_1 = 2$)

$$\mu = \frac{1 - \exp(-A_2 T_n)}{T_n / A_1}$$

- Friction model 1b ($I_1 = 12$)

$$\mu = \frac{1 - \exp(-A_2 F_n)}{F_n / A_1}$$

- Friction model 2a ($I_1 = 3$)

$$\mu = A_2 + (A_2 - A_1) \exp(-A_3 T_n)$$

- Friction model 2b ($I_1 = 13$)

$$\mu = A_2 + (A_2 - A_1) \exp(-A_3 F_n)$$

4.5.3 Frictional heat generation

The heat generation resulting from frictional contact can be accounted for in a thermomechanically coupled analysis. The user can select the fraction of this heat that is lost, as well as the fractions going into the contactor and target surfaces. Details are provided in Chapter 11 of the ADINA-T Theory and Modeling Guide.

4.6 Contact definition features

4.6.1 Analytical rigid targets

There is the possibility of describing target contact surfaces as smooth analytically described surfaces for some simple types of geometry. These analytically described surfaces are termed "analytical rigid targets" (ARTs).

The smooth surface representation with ARTs allows for more accurate and more easily converging solutions, especially when friction is present. Most of the applications of ARTs are in metal forming problems.

The possible geometries are infinite planes, spheres, circles, infinite cylinders or strips (the intersections of infinite cylinders with the YZ plane for 2-D problems).

Each ART is connected to one reference node, which defines the motion of the ART. ARTs can move translationally and rotate with the reference node as a rigid body, but cannot deform. Usually the motion of the reference node is controlled with prescribed displacements. In general the reference node can be any node in the model, free to move or with boundary conditions or constraints.

There can be several ARTs in the model, but only one ART in each contact group. The usual node-to-node contact can be used with an ART in a contact group. Contact can either be frictionless or with friction, and all types of static and dynamic analyses in ADINA are possible when ARTs are present.

There are the following limitations. The ARTs are not plotted by the AUI and are therefore invisible. Contact tractions can't be computed with ART contact.

4.6.2 Contact body

Unlike the contact surface, the contact body consists of a group of nodes that are not connected through segments (see Fig. 4.6-1). The contact body can only be the contactor in a contact pair, and the target has to be a contact surface (i.e. the target cannot be a contact body). This is useful for modeling metal cutting applications where the internal nodes of a body can also come in contact once the outer ones are removed. This option can also be

used to model 3D contact problems involving beams (since the beams cannot define a 3D contact surface).

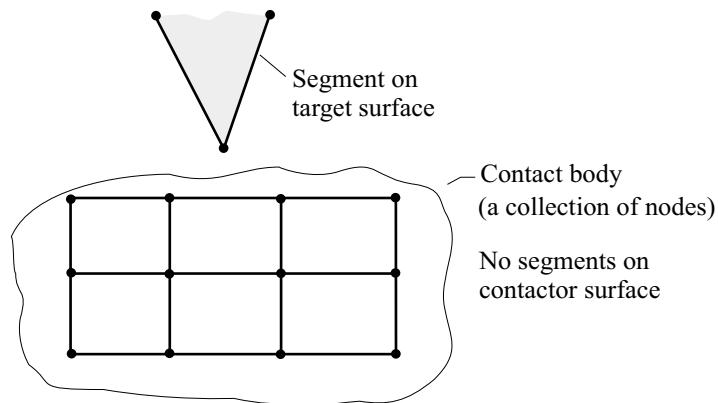


Figure 4.6-1: Contact body

4.6.3 Draw beads

Draw beads are used in metal stamping processes to eliminate defects (as wrinkling or buckling) or to reduce material consumption, see Fig. 4.6-2. Draw beads restrict and control metal flowing into the die cavity by forcing it to bend and unbend before it enters the cavity. The Rigid Target algorithm must be used in conjunction with draw beads. The effect of bending and unbending of the metal sheet is replaced by a restraining force \mathbf{R}_r (acting on shell elements passing through the draw bead line) and an uplifting force \mathbf{R}_u (acting on a pair of tools).

The ADINA draw bead model is available only for MITC4 elements.

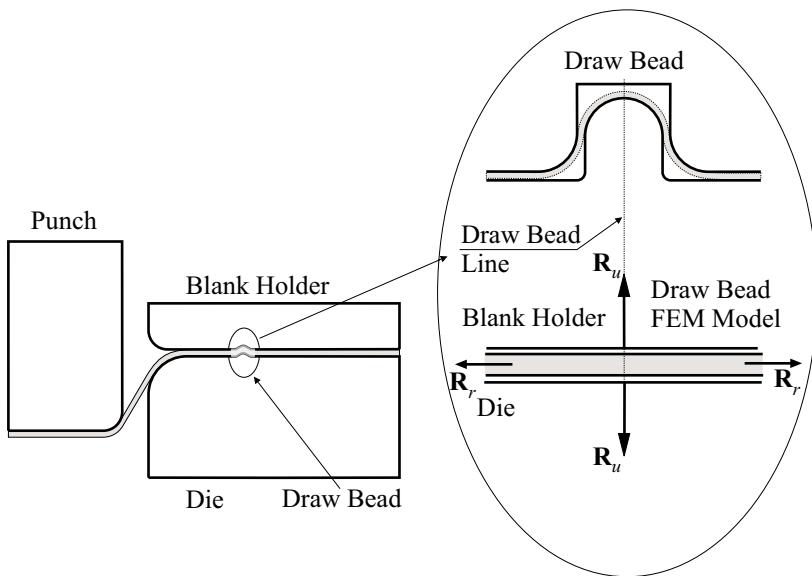


Figure 4.6-2: Draw bead modeling

4.7 Contact analysis features

4.7.1 Implicit dynamic contact/impact

- Oscillations in velocities and accelerations can sometimes be present in implicit dynamic contact analysis especially for high speed impact problems. These oscillations can be reduced by
 - applying post-impact corrections,
 - setting the Newmark parameter $\alpha = 0.5$,
 - adding compliance to the contact surfaces.
- In post-impact corrections, the velocities and accelerations of the contactor and target can be forced to be compatible during contact (only in the normal contact direction). This is achieved by modifying the velocities and accelerations of the contact nodes once convergence is reached such that they satisfy conservation of linear and angular momentum.

- Setting the Newmark $\alpha = 0.5$ instead of the default $\alpha = 0.25$ (trapezoidal rule — see Section 7.3) results in an accurate solution of rigid body impact problems, and frequently has a positive effect on reducing numerical oscillations in flexible body contact.
- Adding compliance to the contact surface can also significantly reduce the numerical oscillations that result from dynamic time integration. In this case, the compliance factor must be selected based on Eq. (4.4-1) such that the contact pressures do not cause excessive penetration. Allowing penetration to the order of 1% of the element size usually eliminates numerical oscillations.
- Only the post-impact correction option requires additional memory and computations.
- The post-impact correction feature should not be used together with any of the other two oscillation suppression features.
- If post-impact correction is activated, all target nodes, except those with all degrees of freedom fixed or enforced displacements, must have a positive non-zero mass. The contactor nodes can have zero mass.

4.7.2 Explicit dynamic contact /impact

- Oscillations in velocities and accelerations can sometimes be present in explicit dynamic contact analysis especially for high speed impact problems. These oscillations are more common with the penalty contact algorithm. In that case, they can be reduced by
 - reducing the normal penalty stiffness,
 - adding penalty contact damping.

See Section 4.3.2 for details on the explicit penalty contact algorithm.

In addition, other sources of damping such as Rayleigh damping can reduce contact oscillations by damping the high frequency modes that generate them.

- Oscillations in results can also occur when using the kinematic constraint algorithm. These oscillations can be due to a mismatch in the masses of the two contacting surfaces. See section 4.7.3 for more details.

4.7.3 Contact detection

- As explained earlier in this chapter, the contact conditions prevent the contactor nodes from penetrating the target segments. During each equilibrium iteration, the most current geometry of the contactor and target surfaces is used to determine and eliminate the overlap at the contactor nodes.
- For single-sided contact, the calculation of overlap at a contactor node k consists of a contact search, followed by a penetration calculation. The contact search starts by identifying all possible target surfaces where node k can come into contact. For each of these target surfaces:
 - Find the closest target node n to node k .
 - Find all the target segments attached to node n .
 - Determine if node k is in contact with any of these segments.
 - If node k is in contact, update the information.
 - If the new contact surfaces are used, and no appropriate target segment is detected, the contact search is expanded beyond the target segments attached to node n .
- For double-sided contact, the contact search algorithm uses time tracking and checks whether the contactor node penetrated a target segment between times t and $t + \Delta t$.
- Note that the contactor and target bodies must lie on opposite sides of the interface (see Fig. 4.7-1). Thus for admissible conditions of contact the dot product of the normals to the

contactor surface and the target surface must be a negative number.

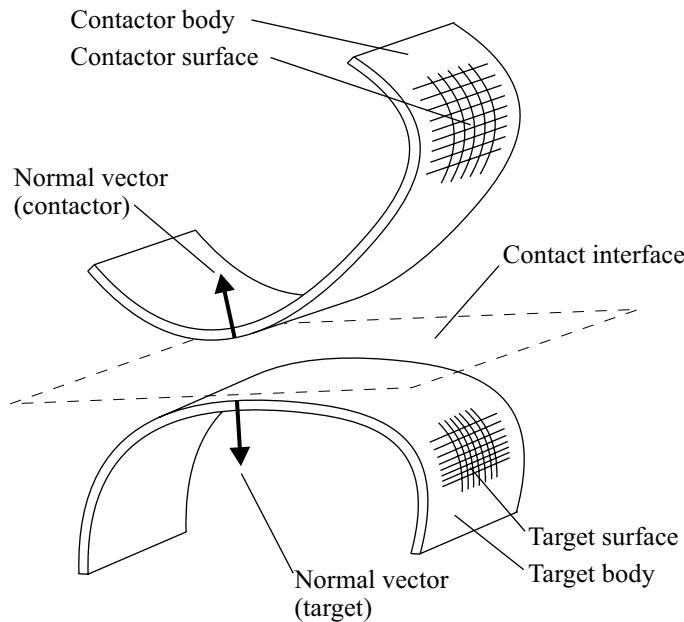


Figure 4.7-1: Admissible conditions of contact (dot product of normals to contactor surface and target surface is a negative number)

4.7.4 Suppression of contact oscillations (not available in explicit analysis)

- In some problems contactor nodes may oscillate during equilibrium iterations between several (usually two) neighboring target segments. Frequently, both solutions are acceptable. A special procedure can be used to prevent such oscillations. This is done by selecting a non-zero NSUPPRESS parameter in the CONTACT-CONTROL command. In this case, the program records the pairing target segment for each contactor node in the previous NSUPPRESS iterations. Once this array is full, and the contactor node is still in contact, and the pairing target segment is one of those recorded in previous iterations, the suppression feature is activated. The contactor node from this iteration onwards is associated with only that target segment. It may remain in contact

with the segment, or in contact with an infinite plane passing through the segment, or it can separate from contact completely. The node is released from its restrictions once iteration ceases, either because convergence is reached, or due to non-convergence.

- If this oscillation suppression feature is used, it is recommended that NSUPPRESS be set greater or equal to 5 and at least 5 less than the maximum number of iterations.
- Note that there is memory overhead associated with this feature, where an integer array of size NSUPPRESS is defined for all contactor nodes.

4.7.5 Restart with contact

- Changes in contact parameters are allowed between restarts, with some exceptions. Some restrictions exist, such as no restart from friction to frictionless and vice versa.
- The contact algorithm itself for a certain contact group can also change in a restart. For this purpose, the contact algorithms can be divided into two groups. The first group includes the constraint function (implicit), Lagrange multiplier segment (implicit), kinematic constraint (explicit), and penalty (explicit). Restarts are possible between different algorithms in this group.

The second group includes the implicit and explicit Rigid Target algorithms. Restarts are possible between these algorithms. However, restarts are not allowed between the two groups.

4.8 Modeling considerations

4.8.1 Contactor and target selection

- For some contact problems, the contactor and target surfaces in a contact pair can be interchanged without much effect on the solution. However, for many cases, one of the two alternatives is better.
- If it is more important for the nodes of one surface not to

penetrate the other, then that surface should be the contactor. This factor is usually important when one surface has a much coarser mesh than the other as shown in Fig. 4.8-1. The coarse surface should preferably be the target in this case. A related condition occurs around corners or edges as shown in Fig. 4.8-2. The upper surface should preferably be the contactor in this case.

- If one of the surfaces has mostly dependent degrees of freedom, it should be the target. This dependency can be due to boundary conditions, constraints or rigid elements. The surface can also be rigid if its nodes are not attached to any elements. In that case too it has to be the target (except in the explicit penalty algorithm where this is permitted).
- If one surface is significantly stiffer than the other, it should preferably be the target, unless one of the two conditions above also exist.

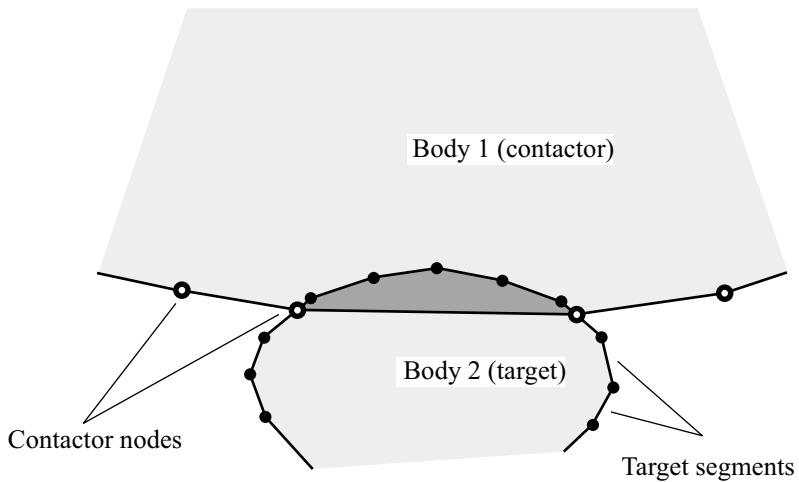


Figure 4.8-1: Effect of incorrect contactor-target selection due to mesh density

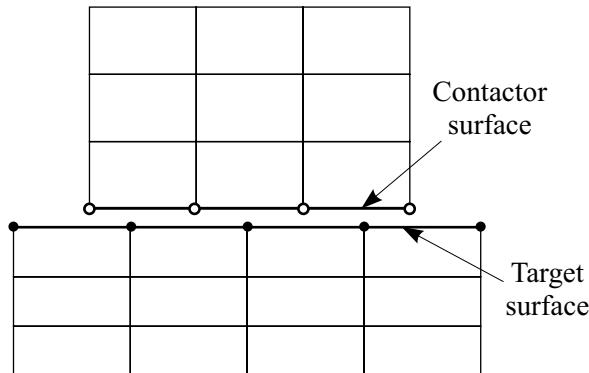


Figure 4.8-2: Target selection for surfaces of different sizes

4.8.2 General modeling hints

- ADINA automatically determines the direction of the contact surfaces on the faces of solid elements. For contact surfaces defined on structural elements (beams, trusses and shells) the user has to ensure that the correct direction is defined (except in double-sided contact).
- Rigid target surfaces can be modeled using nodes with no degrees of freedom or nodes with enforced displacements for all active degrees of freedom. As a result, a fine discretization of a complex rigid surface geometry is possible with only a small increase in the solution cost.
- If the contact surface is rigid it can be a slave to a master node that will control the motion of the rigid surface. Rigid links are created between the master node and all the nodes on the rigid surface.
- It is acceptable for the nodes on the contactor and target surfaces to be coincident (have identical coordinates). In this case, it is important to ensure that the two surfaces do not share the same nodes.
- In general it is recommended that the lengths of segments on the contactor and target surfaces be approximately equal. This is particularly important if multiple contact surface pairs are

considered in the analysis or if the contact surface geometries are complex.

- If required, a contactor surface can be modeled as almost rigid by choosing a reasonably high Young's modulus for the finite elements modeling the contactor surface. However, the stiffness of the surface elements should not be excessively high and make the model ill-conditioned.
- If the degrees of freedom of a node on a contactor surface are used in constraint equations or attached to a rigid element (see Section 5.15.2), the contactor node degrees of freedom should preferably be independent.
- If a contactor node has all of its translational degrees of freedom dependent, the node is dropped from the contact surface.
- If the contact surfaces are smooth (i.e., the coefficient of friction is small), the frictionless model is recommended as it is less costly to use. It is also recommended that prior to any contact analysis involving friction, the frictionless solution is first obtained, whenever possible.
- It is not recommended that contact pairs with friction coexist with contact pairs without friction in the same contact group.
- A contactor node should preferably not belong to more than one contact surface in a contact group, otherwise the contactor node may be over-constrained.
- Restarting from frictionless contact to contact with friction and vice versa is not possible. However, it can be done if the frictionless analysis is replaced by a frictional analysis with a very small friction coefficient.
- Ignoring initial penetrations is a useful option when these penetrations are just a product of the finite element discretization, meaning that they do not exist in the physical model. Fig. 4.8-3 illustrates one such case involving contact between concentric cylinders. In this situation, if initial penetrations are eliminated, the contact algorithm will try to push the penetrating contactor nodes

to the target surface segments in the first step, creating initial prestressing. These initial penetrations and any prestressing that they might cause are unrealistic. Ignoring them is useful in this case. Note however, that if either cylinder is significantly rotated the initial penetrations calculated at each contactor node (in the initial configuration) will no longer be valid. In this case, the best alternative would be to use a much finer mesh.

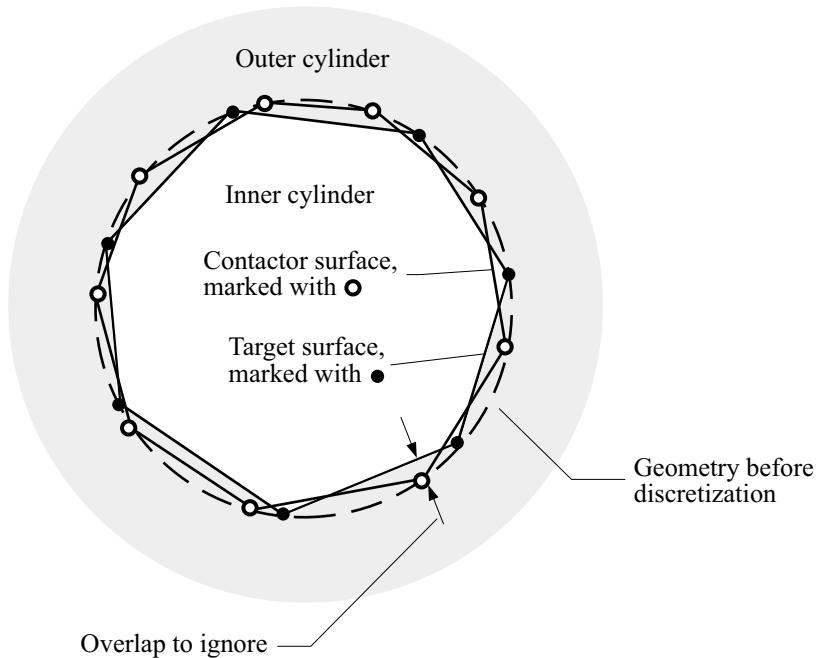


Figure 4.8-3: Analysis of contact between concentric cylinders,
initial penetration is ignored

4.8.3 Modeling hints for implicit analysis

- It is recommended that the ATS method be used in contact analysis (see Section 7.2.1).
- Line search can sometimes be beneficial for contact problems.
- Frictional contact problems using the constraint function algorithm can be sensitive to the choice of frictional regularization constant. For most problems, this parameter should be one or two

orders of magnitude smaller than the expected sliding velocity. Using an excessively large value leads to a smoother friction law, which generally converges faster but results in smaller frictional forces or more sliding. Using an excessively small value enforces the Coulomb law more accurately but is more likely to experience convergence difficulties.

- Friction is not regularized or smoothed in the Lagrange multiplier segment algorithm. This results in accurate enforcement of stick and slip, but is more likely to experience convergence difficulties.
- Friction in the Rigid Target algorithm is regularized by replacing the perfectly rigid (no-slip) stick friction state by an elastic stick state via the tangential stiffness parameter. Using an excessively large value leads to a smoother friction law, which generally converges faster but results in smaller frictional forces or more sliding. Using an excessively small value enforces the Coulomb law more accurately but is more likely to experience convergence difficulties.
- Geometric and material nonlinearities can highly depend on the sequence of load application. Thus, for problems involving such features, the load steps should be small. The time step Δt should also be small in dynamic analysis and when time dependent material constitutive relations (e.g., creep) are used.
- If a contact surface with corners or edges is modeled with continuous contact normals, the normal vectors may be inaccurate as shown in Fig. 4.8-4(a). In this case, switch to discontinuous normals, use different contact surfaces for each smooth part (Fig. 4.8-4(b)) or use a fine mesh close to the corners or edges (Fig. 4.8-4(c)).

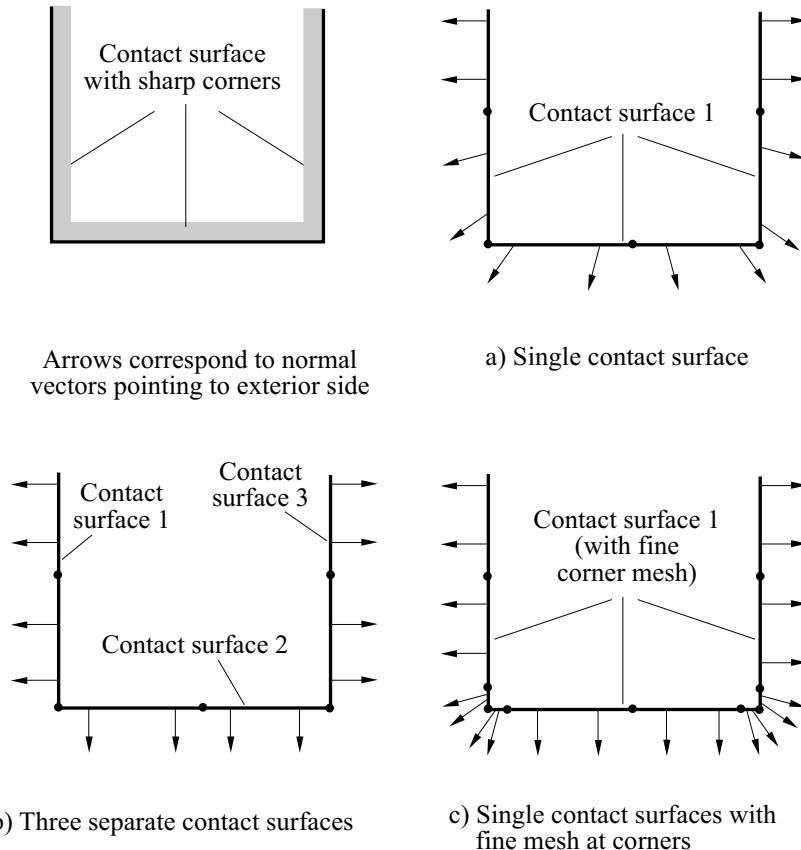


Figure 4.8-4: Defining contact surfaces (with continuous normal vectors) in the presence of corners

4.8.4 Modeling hints for explicit analysis

- The penalty algorithm is preferred when both surfaces are rigid or have many fixed or prescribed nodes.
- Large oscillations in the contact forces may occur when using the penalty method even though the model is stable. These can be reduced by reducing adding a penalty damping term and/or reducing the penalty stiffness.
- When using the penalty contact algorithm it is important to check that the contact stiffnesses are properly selected. Unduly

small penalty stiffnesses will lead to excessive penetrations, while unduly large penalty stiffnesses will lead to excessive oscillations or unstable time integration.

- Large mismatches between the masses of contacting surfaces should be avoided when using the kinematic constraint method. This mismatch is common when contact involves a rigid surface with a small mass and an applied force, as shown in Fig. 4.8-5. The best solution in such cases is to minimize the mismatch by increasing the mass of the rigid surface.

The inaccuracy in this case results from the way the contact is enforced. The kinematic constraint method first predicts displacements without contact then applies a contact correction. The contact conditions are satisfied more accurately when the penetrations in the predicted configuration are small which is usually the case due to the small time step size of explicit analysis. However, some cases such as that mentioned above lead to large projected penetrations which results in incorrect contact conditions and tensile contact forces.

- Large mismatches between the masses of contacting surfaces can also lead to problems when using the penalty method. In this case, the normal penalty stiffness required to avoid instability (without reducing the time step) can be unduly small leading to excessive penetrations. The best solution in such cases is to minimize the mismatch by increasing the mass of the rigid surface, or increase the penalty stiffness by setting it manually or by reducing the time step.

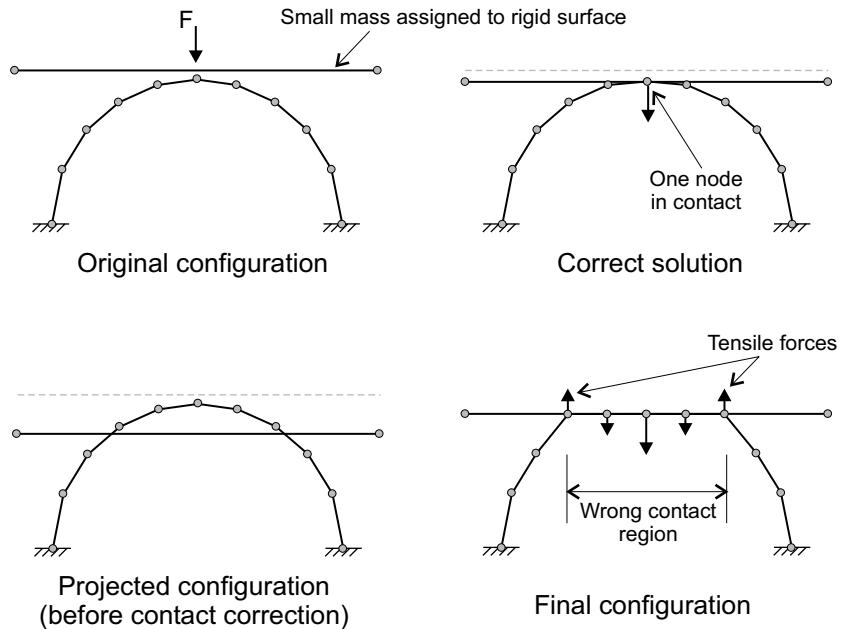


Figure 4.8-5: Performance of kinematic contact algorithm when contact surfaces have a large mass mismatch.

4.8.5 Convergence considerations (implicit analysis only)

- When ADINA fails to converge during the incremental analysis, the intermediate printout given by ADINA in the output listing can provide some useful information (see Fig. 4.8-6).
- Three non-contact related norms are given: first for the energy convergence criterion, the displacement and rotation convergence criterion (boxes b and c), and the force and moment convergence criterion (boxes d and e). Each box has 3 lines of output with the top one giving the norm of the quantity, the second one giving the equation number corresponding to the maximum value, and the third line giving the maximum value. See Chapter 7 for definitions and more details on these norms.
- Box f of Fig. 4.8-5 shows the contact related norms. Parameter CFORCE indicates the norm of the change in the contact forces (between two iterations), and parameter CFNORM gives the norm of the contact force vector.

- The following additional convergence criterion is used since contact is present:

$$\text{RCTOL} = \frac{\text{CFORCE}}{\max(\text{CFNORM}, \text{RCONSM})}$$

where RCTOL is the contact force convergence tolerance and RCONSM is the reference contact force to prevent possible division by zero in the contact convergence criterion above.

- When the maximum number of iterations is reached without convergence, and all norms are decreasing, the maximum number of iterations should be increased.

```

I N T E R M E D I A T E   P R I N T O U T   D U R I N G   E Q U I ...
OUT-OF-          NORM OF
BALANCE          OUT-OF-BALANCE      NORM OF INCREMENTAL ...
ENERGY          FORCE (EQ MAX)    MOMENT (EQ MAX)    DISP. (EQ MAX)    ROTN. (EQ MAX)    CFORCE CFNORM ...
                   VALUE           VALUE           VALUE           VALUE           VALUE           ...
FOR ITE=0        box b       box c       box d       box e       box f
.95E+04
ITE=1  -.49E-09  .17E-08  .00E+00  .66E+01  .00E+00  .67E+04  ...
          ( 85 )  ( 0 )  ( 6 )  ( 0 )  .00E+00  .67E+04  ...
          -.81E-09  .00E+00  .12E+01  .00E+00
ITE=2  .76E+04  .15E+05  .00E+00  .17E+01  .00E+00  .22E+05  ...
          ( 67 )  ( 0 )  ( 81 )  ( 0 )  .00E+00  .21E+05  ...
          .97E+04  .00E+00  .31E+00  .00E+00
ITE=3  -.24E-10  .42E-09  .00E+00  .56E+00  .00E+00  .25E+05  ...
          ( 65 )  ( 0 )  ( 9 )  ( 0 )  .00E+00  .19E+05  ...
          -.17E-09  .00E+00  .14E+00  .00E+00

```

```

... I N G   E Q U I L I B R I U M   I T E R A T I O N S
... CONVERGENCE RATIOS   CONVERGENCE RATIOS   OUT-OF-BALANCE LOAD
... FOR OUT-OF-BALANCE   FOR INCREMENTAL   VECTOR CALCULATION
... ENERGY   FORCE   DISP.   CFORCE   BETA   RATIO
...                   MOMENT   ROTN.   (ITERNS)
... COMPARE WITH       COMPARE WITH
... ETOL     RTOL     DTOL     RCTOL
...                   (NOT USED)
...  -.51E-13  .57E-13  .00E+00  .10E+01  .10E+01  .93E-13
...  .00E+00  .00E+00
...  .80E+00  .49E+00  .00E+00  .10E+01  .10E+01  -.43E+14
...  .00E+00  .00E+00
...  -.25E-14  .14E-13  .00E+00  .13E+01  .10E+01  -.77E-14
...  .00E+00  .00E+00

```

Figure 4.8-6: ADINA Output listing of convergence criteria during equilibrium iterations

- When the norms are rapidly changing before convergence fails, it is commonly caused by applying the load too quickly or using a large time step.
- When CFNORM is stable but CFORCE changes rapidly during equilibrium iterations, the contact can be oscillating between 2 or more close solutions. In this case, try to change the time stepping, or turn on the suppression of contact oscillations feature. When CFNORM varies rapidly, usually the other three norms also vary.
- The segment contact algorithm should not be used with force or energy/force convergence criteria.

5. Boundary conditions/applied loading/constraint equations

5.1 Introduction

The objective of this chapter is to present the various options available in ADINA for the description of boundary conditions, applied loads and constraint equations.

- As discussed in ref. KJB, Section 3.3.2, there are two classes of boundary conditions: essential boundary conditions, such as prescribed displacement (and rotation) boundary conditions, comprise the first class; and natural boundary conditions, such as applied force and moment boundary conditions, comprise the second class.
- Displacement boundary conditions include nodal degree of freedom fixities, prescribed displacements and constraint equations.
- Force and moment boundary conditions include various types of applied loading.
- All displacement and force boundary conditions can be referred to the global Cartesian system or to predefined skew systems. For more information about skew systems, see Ref KJB, Section 4.2.2.
- The externally applied load vector used in the governing equilibrium equations is established using contributions from the various applied loads (see Chapter 7).
For concentrated loads (Section 5.2) and user-supplied loads (Section 5.14), the contributions of these nodal loads are directly assembled into the externally applied load vector.

For pressure loading, distributed loading, centrifugal loading and mass proportional loading, pipe internal pressure loading, and electromagnetic loading, ADINA first calculates the corresponding consistent nodal load vectors (consistent in the sense that the principle of virtual work is used) and then assembles these load vectors into the externally applied load vector. The evaluations of the consistent nodal load vectors for the various types of loading are described in the following sections.

- The options of prescribing nodal displacements, temperatures and temperature gradients are provided to enable you to specify the time variations of these quantities throughout the solution period.

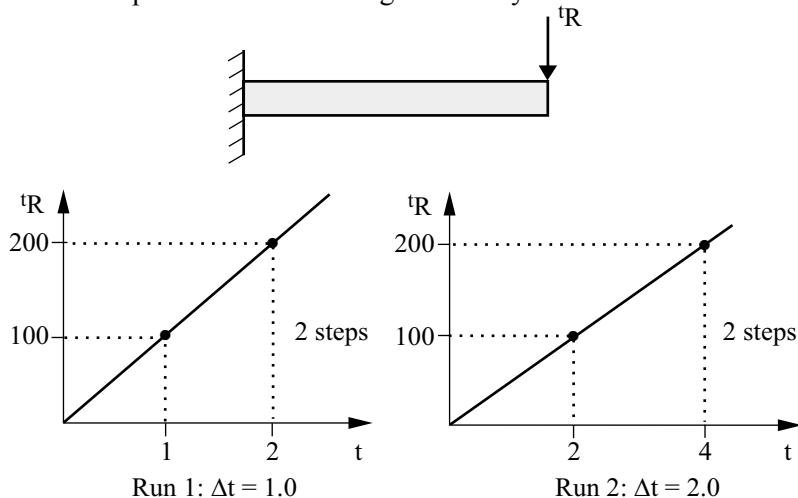
Temperatures and temperature gradients in ADINA are used in conjunction with material models which include temperature effects, see Sections 3.5 to 3.7 and 3.11.

- The definition of the time variation of the externally applied load vector for the various time steps in the solution period depends on whether automatic step incrementation is used or not.

Time variation of externally applied loads when automatic step incrementation is not used: Each applied load is associated with a time function which defines the time variation of the load throughout the solution period.

- ▶ In a static analysis in which time-dependent effects (such as creep, viscoplasticity or friction) are not included in the material models, time is a "dummy" variable which is used, via the associated time function of each applied load, to define the load intensity at a step. Thus, the time step increment directly establishes the load increments. In the example shown in Fig. 5.1-1, note that the same solution is obtained regardless of the size of the time step increment.
- ▶ In a dynamic analysis or if time-dependent effects are included in the material models in a static analysis, time is used in a similar way to define the load intensity of an applied load at a step. However, in these cases, time is a "real" variable because the time step increment is employed in the actual integration of the equations of motion in a dynamic analysis, and in the integration of the element stresses in a creep or

viscoplastic analysis. Hence, in these cases the choice of the time step increment is no longer arbitrary.



Note: identical results are obtained in Run 1 and Run 2
for a linear static analysis.

Figure 5.1-1: Example of the definition of load variation using time

Time variation of externally applied loads when automatic step incrementation is used: Two options are available:

- ▶ Using the automatic-time-stepping (ATS) procedure (see Section 7.2), the loads are defined for all times $\Delta t, 2\Delta t, \dots$ as for no automatic step incrementation. In addition, when the algorithm subdivides a time (load) step, the load vector is established by linear interpolation of the load vectors at times t and $t + \Delta t$.
- ▶ Using the load-displacement-control (LDC) procedure (see Section 7.2), the applied loads are not associated with any time function and the time variation of the loads cannot be specified by the user. The contributions from all the loads are assembled into a constant load vector denoted as the reference load vector. During the response calculation, this reference load vector is scaled proportionally using a load multiplier (in general different from one step to the next) automatically computed by the program.

- A time function is defined as a series of points $(t, f_i(t))$ in which t is time and $f_i(t)$ is the value of time function i at that time. Between two successive times, ADINA uses linear interpolation to determine the value of the time function.
- ADINA includes special time function multiplying functions that alter the time function value obtained using the $(t, f_i(t))$ input data. In the following, let $f^*(t)$ be the value of the time function obtained from the $(t, f_i(t))$ input data and let $f(t)$ be the value of the time function actually used by ADINA. Then the following multiplying functions are available:

Constant: $f(t) = f^*(t)$

Sinusoidal: $f(t) = f^*(t) \cdot \sin(\omega t + \phi)$ where ω is the angular frequency in degrees/unit time and ϕ is the phase angle in degrees.

Short circuit (type 1): $f(t) = f^*(t) \cdot \sqrt{a + b \exp(-t/\tau)}$ in which a and b are constants and τ is the time constant (unit of time).

Short circuit (type 2):

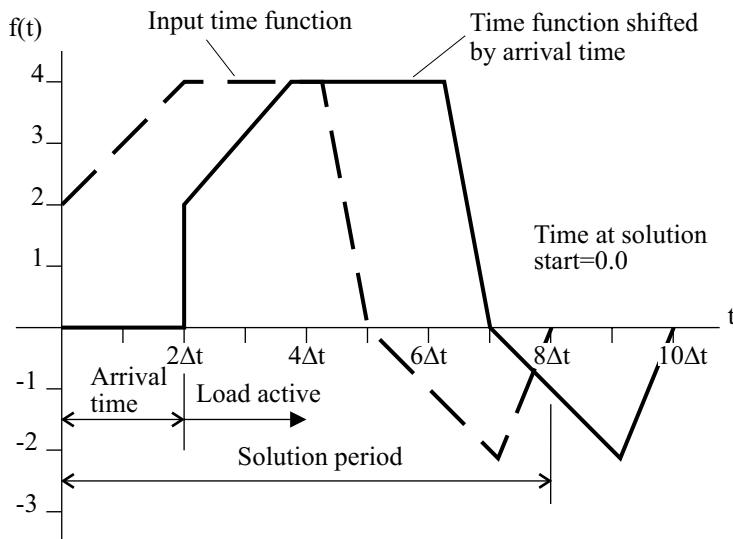
$$f(t) = f^*(t) \cdot \left(\sqrt{\frac{\mu_0}{4\pi}} \cdot \sqrt{2} \cdot I \cdot (\sin(\omega t - \phi + \alpha) + \exp(-t/\tau) \cdot \sin(\phi - \alpha)) \right)$$

in which μ_0 is the magnetic permeability in (volt-seconds)/(meters-amperes), I is the RMS value of the short circuit current, ω is the angular frequency in degrees/unit time, ϕ is the phase angle in degrees, α is the impedance angle in degrees and τ is the time constant (unit of time).

- If time functions are used to determine the time variation of the applied loads, then the activation of the various applied loads can be delayed through the use of the arrival time option. The arrival time option does not apply, however, to centrifugal and mass-proportional loading, see Section 5.4.

The specification of a nonzero arrival time corresponds to a shifting of the associated time function forward in time. Note that the time function multiplier is zero for all times t smaller than the arrival time; see illustrations given in Fig. 5.1-2(a) and (b).

Time functions:

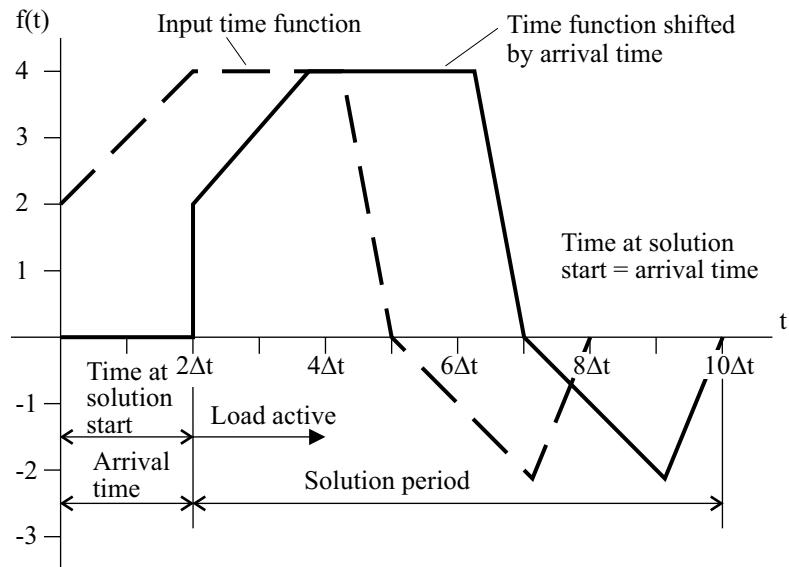


Resulting load values for $t_R = 10f(t)$:

Time step	1	2	3	4	5	6	7	8
$n\Delta t_R$	0	0	30	40	40	40	0	-10

Figure 5.1-2(a): Example on the use of the arrival time option

Time functions:

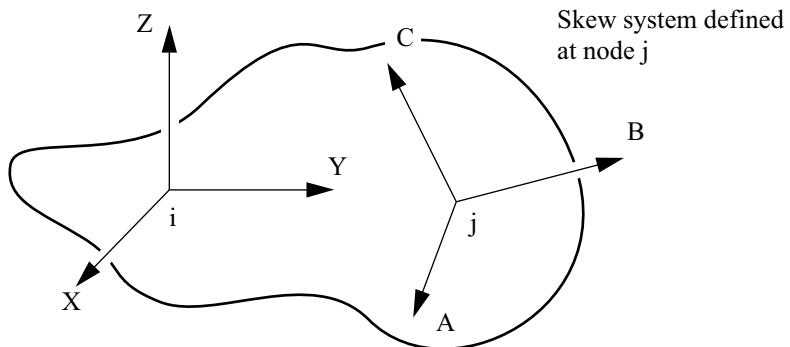
Resulting load values for $t^R = 10f(t)$ and $t_0 = 2\Delta t$:

Time step	1	2	3	4	5	6	7	8
$t_0 + n\Delta t R$	30	40	40	40	0	-10	-20	0

Figure 5.1-2 (b): (continued)

5.2 Concentrated loads

- Concentrated loads are nodal point forces or moments which are applied at the specified nodes and act in the specified degrees of freedom.
- The direction in which a concentrated load acts depends on the coordinate system(s) assigned to the node of load application and upon whether the concentrated load is assigned to translational or rotational degrees of freedom; see Fig. 5.2-1.



Node	Degree of freedom (direction)	Corresponding load component
i	1	X force
	2	Y force
	3	Z force
	4	X moment
	5	Y moment
	6	Z moment
j	1	A force
	2	B force
	3	C force
	4	A moment
	5	B moment
	6	C moment

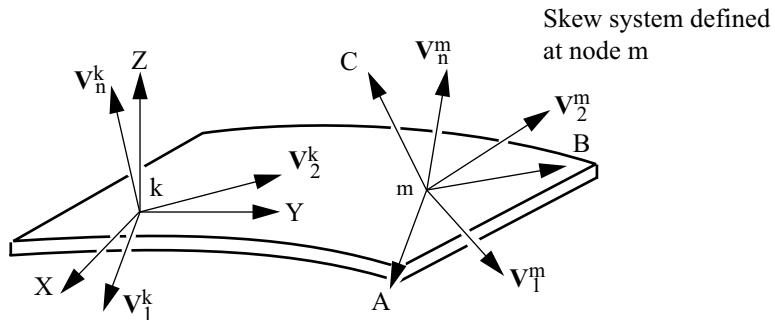
(a) Case of nodes with six degrees of freedom

Figure 5.2-1: Reference systems for application of concentrated loads

Translational degrees of freedom:

- If no skew system is defined at the node, then the concentrated loads applied to these degrees of freedom correspond to nodal forces acting in the global X, Y, and Z directions.

- If a skew system (A, B, C) is assigned at the node, then the concentrated loads applied to these degrees of freedom correspond to nodal forces acting in the A, B and C directions.



Node	Degree of freedom (direction)	Corresponding load component
k	1	X force
	2	Y force
	3	Z force
	4	V_1 moment
	5	V_2 moment
	6(*)	V_n moment
m	1	A force
	2	B force
	3	C force
	4	V_1 moment
	5	V_2 moment
	6(*)	V_n moment

(*) Note that this degree of freedom is automatically deleted by the program and moments applied at this degree of freedom are ignored.

(b) Case of nodes with five degrees of freedom

Figure 5.2-1: (continued)

Rotational degrees of freedom:

- ▶ In this case a distinction must be made between shell midsurface nodes using five degrees of freedom and the shell midsurface nodes using six degrees of freedom.
- ▶ If six degrees of freedom are used, and if no skew system is defined at the node, the concentrated loads applied to these degrees of freedom correspond to nodal moments acting about the global X, Y, and Z directions. If a skew system (A, B, C) is assigned at the node, then the concentrated loads correspond to nodal moments acting about the A, B, and C directions.
- ▶ If five degrees of freedom are used, the concentrated loads applied to these degrees of freedom correspond to nodal moments acting about the V_1 , V_2 and V_n directions of the midsurface system described in Section 2.7.
- Concentrated loads can also be specified as follower loads. In this case the direction in which a load (force or moment) acts is given by the vector pointing from an auxiliary node to the node of load application. This direction is updated during the response solution by evaluating the most current coordinates of both the node of the load application and the auxiliary node.
Note that follower moments are not allowed at shell midsurface nodes using five degrees of freedom.
- Follower loads should only be applied in a large displacement analysis. Equilibrium iterations (see Chapter 7) should, in general, be performed if follower loads are present.
- The direction of a follower load can be established with respect to a one-dimensional structure (such as a beam structure) using nonlinear rigid links (see Section 5.15.2). An example is given in Fig. 5.2-2.

5.3 Pressure and distributed loading

- Pressure loading can be applied to the surfaces of the following types of elements (see Fig. 5.3-1): **2-D solid**, **2-D fluid**, **3-D solid**,

3-D fluid, plate/shell and shell. Considering pressure loads on 2-D elements, the area over which a pressure load acts includes the element thickness.

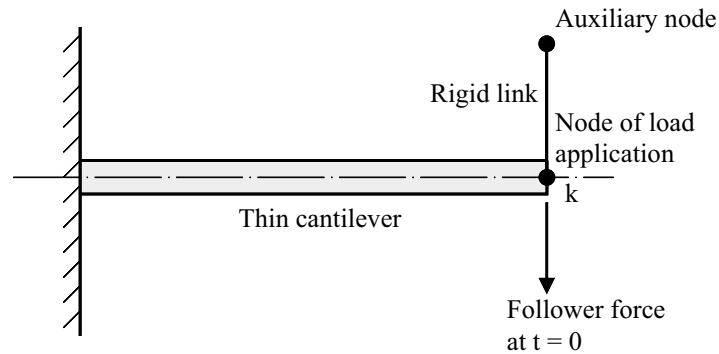
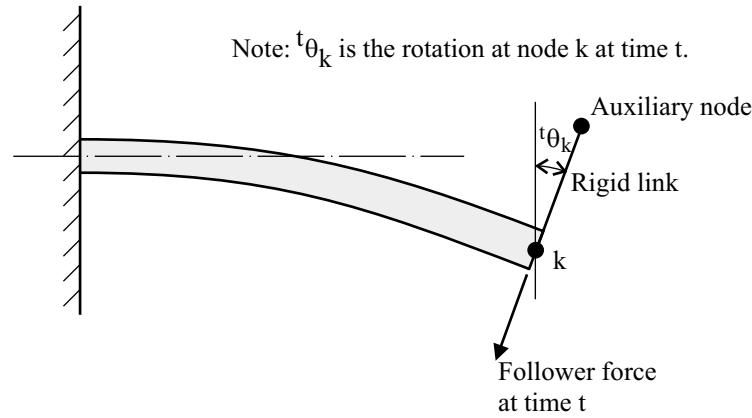
a) Configuration at time $t=0$ b) Configuration at time t

Figure 5.2-2: Example of the use of rigid link to establish the follower load direction

- Distributed loading can be applied along the longitudinal axis of the following types of elements (see Fig. 5.3-2): **beam, iso-beam** (including axisymmetric shells), **pipe, shell** (along edges).

Distributed loads do not include the element cross-section dimensions or thickness. The distributed load is applied along the neutral axis of beam/iso-beam/pipe elements, and along the midsurface line of shell elements. For example, if a shell element with original thickness of 2 undergoes uniaxial tension due to a

distributed load applied to an edge of the shell element, the internal stress is one-half the magnitude of the line load.

- For each pressure/distributed load surface specified, a consistent nodal load vector is calculated to represent the pressure/distributed loading. The formulations used in the consistent load vector calculations for the various loading surfaces are given in the following sections.

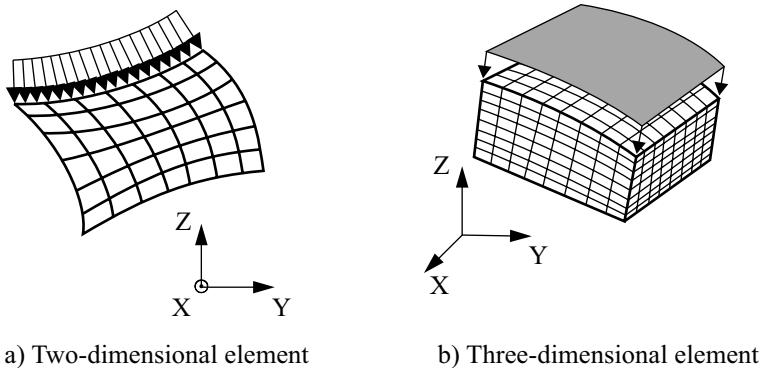


Figure 5.3-1: Examples of pressure loading

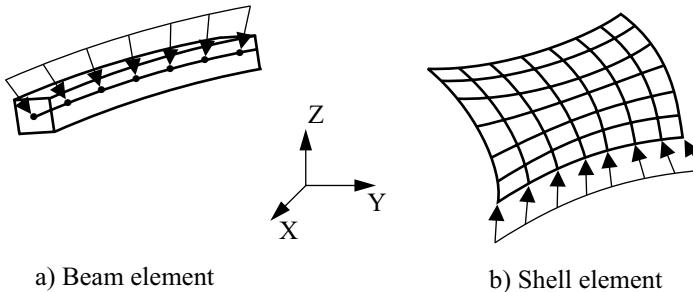


Figure 5.3-2: Examples of distributed loading

- In a large displacement analysis, the pressure/distributed loading can be specified as deformation dependent for all element types. In this case, the calculations of the consistent load vectors are based on the latest geometry and configuration of the loading surface.

- Deformation dependent loading should only be used in a large displacement analysis. Equilibrium iterations (see Chapter 7) should in general be performed if deformation dependent loading is present.
- For pressure loads, you can optionally specify the direction of loading, as follows:
 - a) Pressure acts normal to the surface or line to which it is applied (the default).
 - b) Use x, y or z-component of a).
 - c) Pressure represents a traction in the x, y or z directions.
 - d) Pressure represents a traction tangential to the line (this option is only valid for 2-D element loadings).
 - e) Pressure represents a traction load in a user-specified direction tangent to the surface (this option is only valid for 3-D, plate/shell and shell element loadings).
- For distributed loads, you can optionally specify the direction of loading, as follows:
 - a) Distributed load acts normal to the line to which it is applied (the default).
 - b) Use x, y or z-component of a).
 - c) Distributed load represents a traction in the x, y or z directions.
 - d) Distributed load represents a traction tangential to the line.

5.3.1 Two- and three-dimensional pressure loading

- For two- and three-dimensional pressure loading, the consistent load vector consists of nodal forces acting in the translational degrees of freedom. The calculation of this load vector is given in ref. KJB, Section 4.2.1.

- Deformation-dependent pressure loads that act onto the edges of plane stress elements do not take into account the change in element thickness due to in-plane deformations. For example, a plane stress element that undergoes uniaxial tension due to a deformation-dependent pressure load has internal stresses larger than the pressure load by the ratio (current thickness)/(original thickness).

Load penetration option: If a finite element "dies" due to material rupture or due to a predefined death time, the pressure loads applied to that element are transferred to the neighboring "alive" elements.

In the case when multiple pressure loads are applied to the finite element model, the total pressure applied inside a cavity created by "dead" elements is equal to the sum of the pressure loads applied to the cavity opening divided by the number of pressurized element faces at the opening (see Figure 5.3-3).

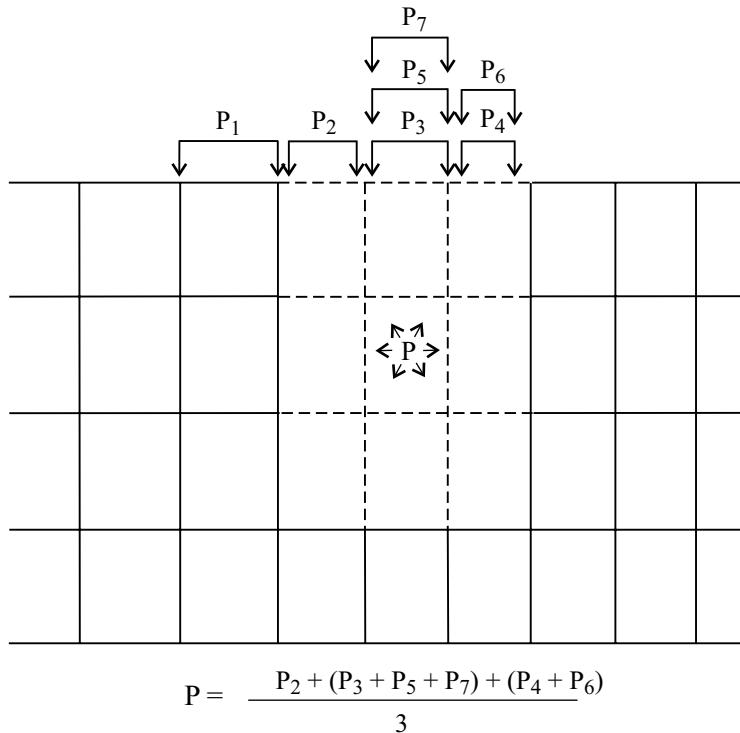


Figure 5.3-3: Pressure applied inside a domain created by "dead" elements

Figure 5.3-4 shows some examples of situations arising when the load penetration option is activated.

Load penetration can be selectively activated for designated element groups only (thus reducing the computational time necessary to perform the analysis).

In the case that a part-through crack or void is being created during the finite element analysis, load penetration should be used with caution and judgement. You must make sure that the external pressure loading is consistent at all times of the analysis with the physics of the problem solved.

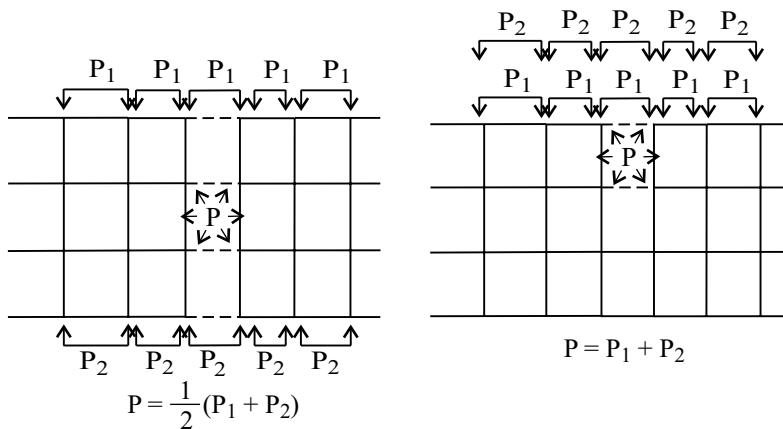


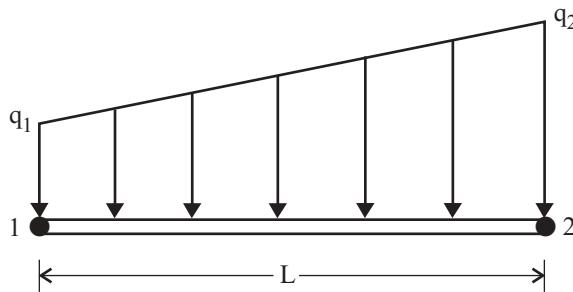
Figure 5.3-4: Some of the situations arising with load penetration

Load penetration cannot be used with deformation independent pressure loading: pressure loading must be deformation dependent.

Note that load penetration is not applicable in case of dynamic analysis using explicit time integration. Load penetration is also not applicable when the automatic LDC method is used.

5.3.2 Hermitian (2-node) beam distributed loading

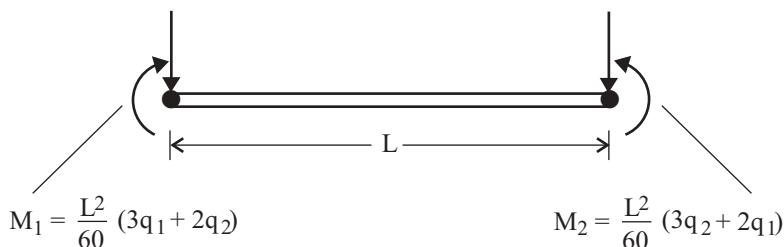
- The distributed loading on a beam element surface is represented by the equivalent concentrated forces and moments acting at the beam nodes, see Fig. 5.3-5. These equivalent forces and moments are equal in magnitude to the fixed-end forces and moments but act in the opposite directions (the fixed-end forces and moments are the reaction forces and moments at the nodes when the beam element is subjected to the distributed load and the nodal displacements and rotations are enforced to be zero). The consistent load vector for beam distributed loading consists of these fixed-end forces and moments.



(a) Beam distributed loading

$$F_1 = \frac{L}{20} (7q_1 + 3q_2)$$

$$F_2 = \frac{L}{20} (7q_2 + 3q_1)$$

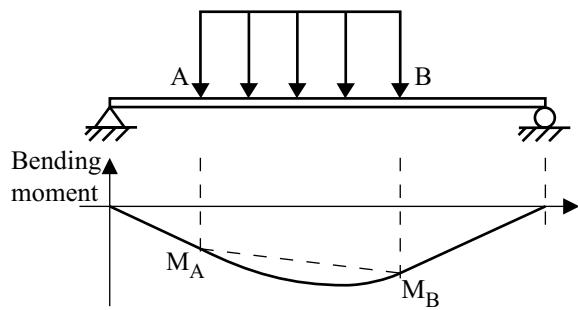


(b) Fixed-end forces/moment representation

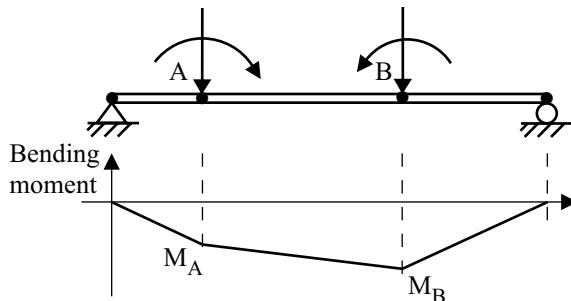
Figure 5.3-5: Representation of beam distributed loading

- Displacements and stresses in the model are calculated by representing the actual distributed loading using the consistent load vector defined above. Hence, the calculated solution corresponds only to these equivalent concentrated nodal forces and moments, and may not correspond entirely to beam theory results taking account of the distributed loading more accurately, as in Figs. 5.3-6 and 5.3-7. Hence, when subjected to distributed loading, the beam element is formulated as in standard finite element analysis of 2-D and 3-D continua. However, an option is available to correct the force and moment results in linear, static analysis to obtain the beam theory solution. For more information on this fixed-end force correction, see the following reference:

ref: W. McGuire and R.H. Gallagher, *Matrix Structural Analysis*, John Wiley & Sons, New York, 1979.



a) Actual beam structure bending moment diagram



b) Numerical results using 3 beam elements

Figure 5.3-6: Schematic beam element bending moments when subjected to distributed loading

5.3.3 Iso-beam and pipe distributed loading

*ref. KJB
Section 4.2.1*

- The distributed loading on an iso-beam or pipe element surface is represented by concentrated forces acting at all nodes on the element longitudinal axis. These forces are calculated using the interpolation functions for the nodal displacements as given in ref. KJB.

Since the nodal translations and rotations are interpolated independently for the iso-beam and pipe elements, the consistent load vector for the distributed loading consists of nodal forces only (no moments).

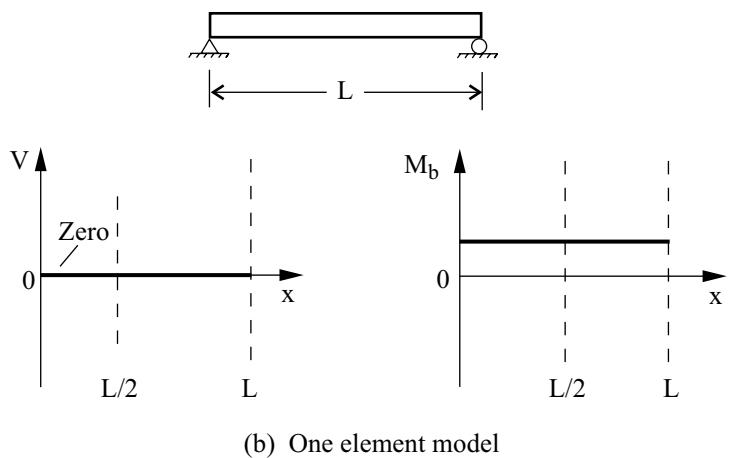
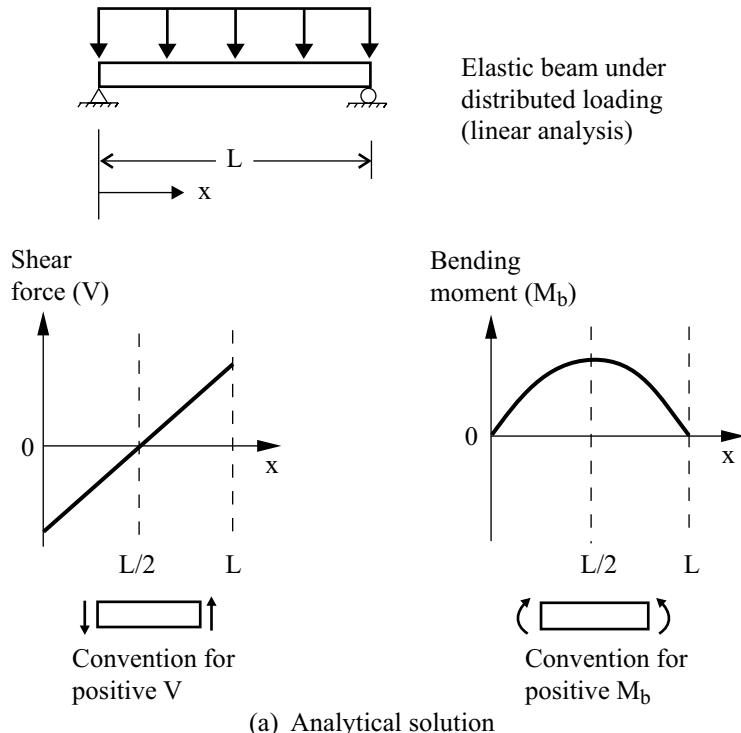
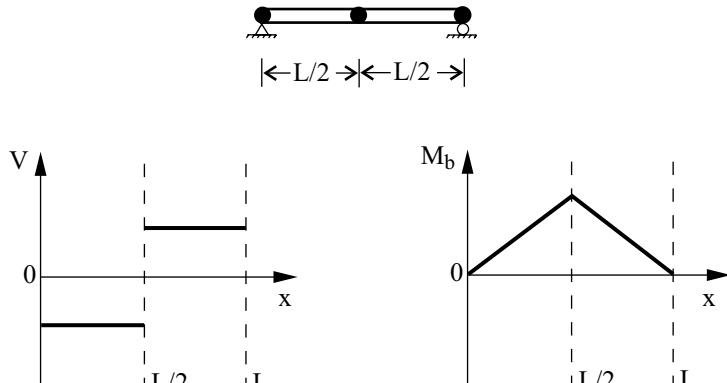
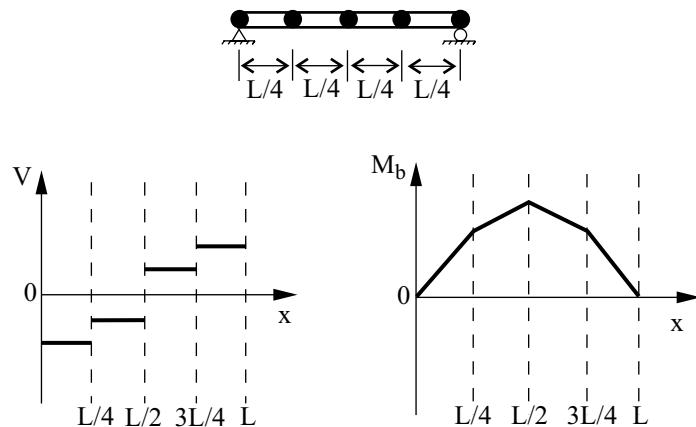


Figure 5.3-7: Schematic beam element forces/moment when subjected to distributed loading



(c) Two element model



(d) Four element model

Figure 5.3-7: (continued)

- For iso-beam and pipe elements subjected to distributed loading, the element nodal forces/momenta and stresses calculated in ADINA are due to the consistent nodal loads.

5.3.4 Plate/shell pressure loading

- The pressure loading on a plate/shell element surface is represented by concentrated forces acting at the three corner nodes. These forces are calculated assuming a linear interpolation of the

transverse displacements between nodes and hence this consistent load representation of the pressure loading is only approximate.

5.3.5 Shell loading

- The pressure loading on a shell element surface is represented by concentrated forces acting at the nodes on the element surface. These forces are calculated using the interpolation functions for the nodal displacements as given in ref. KJB, Section 4.2.1.

Since the nodal translations and rotations are interpolated independently for the shell element, the consistent load vector for the distributed loading consists of nodal forces only (no moments).

- Triangular pressure surfaces obtained by having the nodes 1, 4 and 8 coincident are spatially isotropic, see also Section 2.2.2.
- Shell elements can also be subjected to a distributed line loading acting on the edge of the elements (see Fig. 5.3-2). In this case an auxiliary node needs to be used to define the plane in which the load is acting.

5.4 Centrifugal and mass proportional loading

- Centrifugal and mass proportional loading can be used to model the effect of body forces which arise from accelerations to which the structure is subjected.
- The mass matrix used in the calculation of centrifugal and mass proportional loading can be lumped or consistent. Note that the computational effort involved in the evaluation of a lumped mass matrix is, in general, much less than the effort for a consistent mass matrix.
- Centrifugal and mass proportional loading can both be present in a static or dynamic analysis. However, in a dynamic analysis, the type of mass matrix employed in the load calculation and in the dynamic response calculation must be the same.
- When the element birth and death option is active (see Section 11.4), the consistent load vector consists (at all times) only of the

contributions from the elements currently alive.

- Note that the arrival time option (see the beginning of this chapter) is not applicable for centrifugal and mass proportional loading.

Centrifugal loading

- The consistent load vector for centrifugal loading is computed using the mass matrix of the entire finite element system, the radial distance from the axis of rotation, and the specified angular velocity (see Fig. 5.4-1), as follows:

$$^t\mathbf{R} = ^t\mathbf{M} \left(^t\boldsymbol{\omega} \times \left(^t\boldsymbol{\omega} \times ^t\mathbf{r} \right) \right) \quad (5.4-1)$$

where $^t\boldsymbol{\omega} = \left(\text{OMEGA} \sqrt{\text{FACTOR} \cdot f(t)} \right) \mathbf{n}$, $^t\mathbf{r}$ is the radial vector from the axis of rotation to the node, OMEGA and FACTOR are specified in the load definition, $f(t)$ is the time function and \mathbf{n} is the unit vector parallel to the axis of rotation.

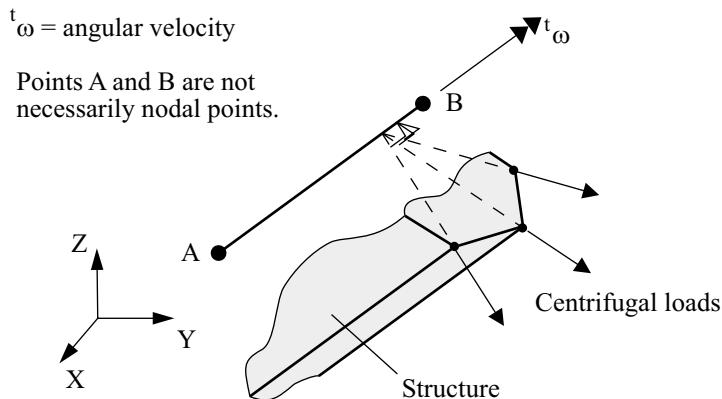


Figure 5.4-1: Convention used for centrifugal loading

Note that the time function associated with the centrifugal loading directly determines the time variation of the centrifugal

forces and not the angular velocity.

Note that the consistent load vector consists of nodal point forces applied to all the nodes of the finite element model. But loads applied to deleted degrees of freedom do not have any effects in the solution.

- When centrifugal loading is used in nonlinear analysis, additional nonlinear terms are added to the stiffness matrix and the load vector. These additional nonlinear contributions are described in the following. Let the equilibrium of the finite element system be calculated at time $t + \Delta t$, iteration i . Then

$${}^{t+\Delta t} \mathbf{M} {}^{t+\Delta t} \ddot{\mathbf{U}}^{(i)} + {}^{t+\Delta t} \mathbf{K}^{(i-1)} \Delta \mathbf{U}^{(i)} = {}^{t+\Delta t} \mathbf{R}^{(i)} - {}^{t+\Delta t} \mathbf{F}^{(i-1)} \quad (5.4-2)$$

where \mathbf{M} = mass matrix, \mathbf{K} = stiffness matrix = $(\mathbf{K}_L + \mathbf{K}_{NL_0})$, $\dot{\mathbf{U}}$ = acceleration vector, $\Delta \mathbf{U}$ = incremental displacement vector, \mathbf{R} = external load vector, \mathbf{F} = nodal point forces corresponding to stresses, (i) = iteration i , and $t + \Delta t$ = time $t + \Delta t$.

The complete expression of , including all nonlinear effects is

$${}^{t+\Delta t} \mathbf{R}^{(i)} = {}^{t+\Delta t} \mathbf{M} \left(\boldsymbol{\omega} \times \left(\boldsymbol{\omega} \times \left({}^0 \mathbf{r} + {}^{t+\Delta t} \mathbf{U}^{(i-1)} + \Delta \mathbf{U}^{(i)} \right) \right) \right) \quad (5.4-3)$$

where ${}^0 \mathbf{r}$ = vector of initial nodal distances to rotation axis and $\boldsymbol{\omega}$ = rotation vector. From the expression (5.4-3), it can be seen that an additional nonlinear contribution \mathbf{K}_{NL_1} to the stiffness matrix is present, which is given by

$${}^{t+\Delta t} \mathbf{K}_{NL_1}^{(i-1)} \Delta \mathbf{U}^{(i)} = {}^{t+\Delta t} \mathbf{M} \left(\boldsymbol{\omega} \times \left(\boldsymbol{\omega} \times \Delta \mathbf{U}^{(i)} \right) \right)$$

and that a deformation-dependent load is present, given by

$${}^{t+\Delta t} \mathbf{R}_{NL}^{(i)} = {}^{t+\Delta t} \mathbf{M} \left(\boldsymbol{\omega} \times \left(\boldsymbol{\omega} \times {}^{t+\Delta t} \mathbf{U}^{(i-1)} \right) \right)$$

- In ADINA, nonlinear centrifugal loading can be used in static analysis, implicit dynamic analysis, and frequency analysis. It cannot be used in explicit dynamic analysis.
- The correction to the stiffness matrix and the correction to the loading are made when a deformation dependent loading is requested.

Mass-proportional loading

- The consistent load vector for mass proportional loading in direction i is computed using the mass matrix of the entire finite element system and the specified accelerations (only in the translational degrees of freedom), as follows:

$${}^t\mathbf{R}_i = {}^t\mathbf{M} \mathbf{d}_i {}^t\mathbf{a}_i \quad (5.4-4)$$

where \mathbf{d}_i is a direction vector with "1" in the portions of the translational degrees of freedom acting into the direction i and "0" in the other portions.

The values of the accelerations are specified as follows:

$${}^t\mathbf{a}_i = \text{MAGNITUDE} \cdot f(t) \cdot A_i, \quad i = x, y, z$$

where MAGNITUDE and A_i are specified in the load definition and $f(t)$ is the time function.

Note that the consistent load vector includes nodal point forces applied to all nodes of the finite element model.

- Mass proportional loading is commonly used to model gravity loading. For gravity loading, ${}^t\mathbf{a}_i$ is the acceleration vector due to gravity. For example, for the z coordinate in the vertical direction (increasing z corresponds to movement away from the ground), enter ${}^t\mathbf{a}_z = -g$, where g is the (positive) acceleration due to gravity.
- Mass-proportional loads are frequently used to model ground motions. The basis for using mass-proportional loads in modeling

ground motions is given briefly now. The equations of motion for linear dynamics, not including damping but including ground motions, are

$$\begin{bmatrix} \mathbf{M}_{11} & \mathbf{M}_{12} \\ \mathbf{M}_{12}^T & \mathbf{M}_{22} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{x}}_1 \\ \ddot{\mathbf{x}}_2 \end{bmatrix} + \begin{bmatrix} \mathbf{K}_{11} & \mathbf{K}_{12} \\ \mathbf{K}_{12}^T & \mathbf{K}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{R}_1 \\ \mathbf{R}_2 \end{bmatrix} \quad (5.4-5)$$

where \mathbf{x}_1 is the vector of nodal point displacements for nodes not attached to the ground and \mathbf{x}_2 is the vector of nodal point displacements for nodes attached to the ground. \mathbf{R}_1 , \mathbf{R}_2 are externally applied forces (for example, concentrated forces).

When the ground motions are the same at all nodes attached to the ground, $\mathbf{x}_1 = \mathbf{u}_1 + \mathbf{d}_{1i}x_{gi}$, $\ddot{\mathbf{x}}_1 = \ddot{\mathbf{u}}_1 + \mathbf{d}_{1i}\ddot{x}_{gi}$, $\mathbf{x}_2 = \mathbf{u}_2 + \mathbf{d}_{2i}x_{gi}$, $\ddot{\mathbf{x}}_2 = \ddot{\mathbf{u}}_2 + \mathbf{d}_{2i}\ddot{x}_{gi}$, where \mathbf{u}_1 is the vector of nodal point displacements relative to the ground for nodes not attached to the ground and \mathbf{u}_2 is the vector of nodal point displacements relative to the ground for nodes attached to the ground. Clearly, $\mathbf{u}_2 = \mathbf{0}$. Also \mathbf{d}_{1i} is the direction vector for the nodes not attached to the ground and \mathbf{d}_{2i} is the direction vector for the nodes attached to the ground, with “1” in the portions of the translational degrees of freedom acting into the direction i and “0” in the other portions.

The matrix equation of motion becomes

$$\begin{bmatrix} \mathbf{M}_{11} & \mathbf{M}_{12} \\ \mathbf{M}_{12}^T & \mathbf{M}_{22} \end{bmatrix} \left(\begin{bmatrix} \ddot{\mathbf{u}}_1 \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} \mathbf{d}_{1i}\ddot{x}_{gi} \\ \mathbf{d}_{2i}\ddot{x}_{gi} \end{bmatrix} \right) + \begin{bmatrix} \mathbf{K}_{11} & \mathbf{K}_{12} \\ \mathbf{K}_{12}^T & \mathbf{K}_{22} \end{bmatrix} \left(\begin{bmatrix} \mathbf{u}_1 \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} \mathbf{d}_{1i}x_{gi} \\ \mathbf{d}_{2i}x_{gi} \end{bmatrix} \right) = \begin{bmatrix} \mathbf{R}_1 \\ \mathbf{R}_2 \end{bmatrix}$$

$$(5.4-6)$$

Now $\begin{bmatrix} \mathbf{K}_{11} & \mathbf{K}_{12} \\ \mathbf{K}_{12}^T & \mathbf{K}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{d}_{1i} \\ \mathbf{d}_{2i} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}$ since the vector $\mathbf{d}_i = \begin{bmatrix} \mathbf{d}_{1i} \\ \mathbf{d}_{2i} \end{bmatrix}$ corresponds to a rigid body motion. The matrix equation of motion becomes

$$\begin{bmatrix} \mathbf{M}_{11} & \mathbf{M}_{12} \\ \mathbf{M}_{12}^T & \mathbf{M}_{22} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{u}}_1 \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} \mathbf{K}_{11} & \mathbf{K}_{12} \\ \mathbf{K}_{12}^T & \mathbf{K}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{u}_1 \\ \mathbf{0} \end{bmatrix} = \begin{bmatrix} \mathbf{R}_1 \\ \mathbf{R}_2 \end{bmatrix} - \begin{bmatrix} \mathbf{M}_{11} & \mathbf{M}_{12} \\ \mathbf{M}_{12}^T & \mathbf{M}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{d}_{1i}\ddot{x}_{gi} \\ \mathbf{d}_{2i}\ddot{x}_{gi} \end{bmatrix}$$

$$(5.4-7)$$

It is seen that the ground acceleration can be applied to the model

as a mass-proportional load, provided that the resulting nodal point motions are interpreted as motions relative to the ground.

Please note:

- ▶ To enter a positive ground acceleration \ddot{x}_{gi} , specify a negative mass-proportional load a_i .
- ▶ All fixities are relative to the ground. In other words, fixing a node attaches it to the ground.
- ▶ All prescribed displacements are relative to the ground.
- ▶ All single DOF springs are attached to the ground.

Damping can be used. However, concentrated dampers and single DOF damping spring elements are attached to the ground. Mass-proportional Rayleigh damping, acts relative to the ground motion.

Although we have illustrated the procedure only for linear dynamics, the procedure is also valid in nonlinear dynamics.

- In a nonlinear static analysis, the nonlinear response due to gravity loading alone can be obtained first and then the structure can be subjected to additional concentrated, surface pressure loading, etc. In this case, the total gravity loading is applied alone in the first N steps, see Fig. 5.4-2. The time function defines the load increments with which the gravity loading is applied. Other concentrated loads, pressure loading, etc. are applied in the load steps by using either the arrival time option or by the use of another time function. Hence, in this case during the last (NSTEP-N) steps, the response of the structure is due to both the maximum gravity loading and any additional concentrated, pressure loading, etc.

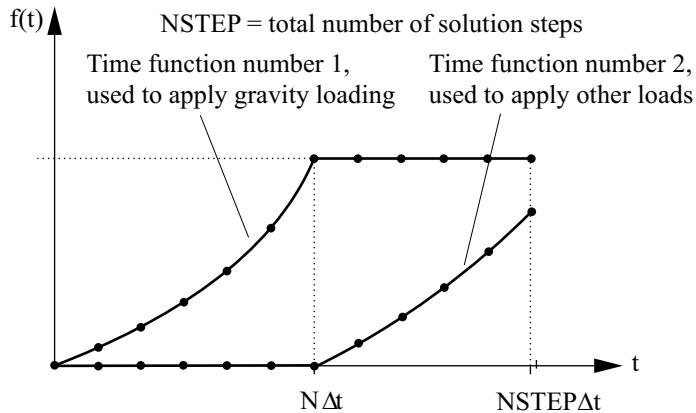


Figure 5.4-2: Use of gravity loading in nonlinear static analysis with other types of loading

5.5 Prescribed displacements

- In some analyses, it is necessary to prescribe displacements at specified degrees of freedom. In ADINA, the following relationship can be specified.

$$u_i = f(t)$$

where f is a general time function (see Section 5.1) which prescribes the nodal point displacement u_i . Prescribed displacement conditions can be used in static and dynamic analysis.

- Nodal point translations and rotations can be prescribed. The degree of freedom for which a displacement is prescribed depends on the coordinate system assigned at the node. The convention used is exactly the same as the convention used for concentrated loads, see Section 5.2.
- Ovalization and warping modes can be prescribed at the nodes of pipe-shell elements.
- A nodal point can be "fixed" by prescribing a zero displacement component for all degrees of freedom at this node. This is,

however, different from imposing a fixity boundary condition at the nodal degrees of freedom because the prescribed degrees of freedom are retained in the system matrices (i.e., equation numbers are assigned) whereas the degrees of freedom at which fixity conditions are imposed are deleted in the system matrices.

- A node which is "fixed" by having zero prescribed displacements at a step can displace due to nonzero prescribed displacements at the same degrees of freedom at another step. This technique can be used to model moving boundaries.
- Note that prescribed displacements are not allowed on contactor surfaces (see Chapter 4).
- In a restart run, there are two options available. One option is to specify the prescribed displacement relative to the original configuration (the default). The other option is to specify the prescribed displacement relative to the deformed configuration. When this option is used, the total displacement (of a degree of freedom onto which the displacement is applied) is calculated as follows. The total displacement is equal to the sum of the calculated displacement component from the previous run, evaluated at the starting solution time TSTART, and the new prescribed displacement value (which can be constant or time-dependent).
- Prescribed displacements can be unloaded (switched to prescribed forces) during the analysis. The unloading can be time or force driven. Time driven unloading requires a time value input at which a prescribed displacement is replaced by the corresponding reaction force acting on the same degree of freedom. Force driven unloading requires a force value input. If the corresponding reaction is equal to or greater than the force value, ADINA switches the loading condition to a prescribed force equal to the corresponding reaction value.

It should be noted that switching from a prescribed displacement to a prescribed force is absolutely necessary if unloading from prescribed displacements is required during the analysis.

5.6 Prescribed temperatures and temperature gradients

- If temperature-dependent material models are used in the analysis, then the nodal point temperatures at all solution steps must be defined for all nodes throughout the solution period.
- Nodal temperature gradients must be defined if shell elements or axisymmetric shell elements are present and temperature-dependent material models are used in the analysis. Nodal temperature gradients should only be assigned to shell midsurface nodes and/or axisymmetric shell nodes. The direction of the temperature gradient at a node is the same as the direction of the normal vector (or the midsurface director vector) at the node, see Section 2.7.1.
- Temperature data can be obtained from an ADINA-T analysis in which the same finite element mesh is used and thus the number of nodal points, the number of elements, etc., are the same as in the ADINA analysis. In this case ADINA-T directly creates a file containing the temperatures. This file is written as an unformatted FORTRAN file with the following line:

```
WRITE (56) TIME, (TEMP (I), I=1, NUMNP)
```

See Chapter 8 of the ADINA-T Theory and Modeling Guide for more information.

- Temperature data can also be obtained from an ADINA-T or ADINA-F analysis in which a different finite element mesh is used. In this case, ADINA-T and ADINA-F create mapping files, which are then read by the AUI in order to create temperature files for the ADINA analysis.
- Also, the nodal temperatures for all solution steps can be defined via direct input, in which case the time variations of the temperatures are specified by time functions. In the following, we describe how this is done using the AUI.

The formulas used to determine the prescribed nodal temperature at node i are

$'\theta_i = \text{TLOAD}$, if no temperature is prescribed at the node

and

$$'\theta_i = \frac{1}{N} \sum_{j=1}^N f_j(t) F_i^j \quad \begin{array}{l} \text{If at least one temperature is prescribed} \\ \text{at the node using geometry} \end{array}$$

and

$$'\theta_i = \sum_{j=1}^N f_j(t) F_i^j \quad \begin{array}{l} \text{if at least one temperature is prescribed} \\ \text{directly at the node} \end{array}$$

In these formulas, TLOAD is the overall prescribed temperature (also referred to as the default prescribed temperature or the reference prescribed temperature), N = number of temperature load applications at the node, f_j is the time function associated with load application j (not necessarily time function number j) and F_i^j is the multiplying factor associated with load application j .

- If the arrival time option is used in the nodal temperature definition, then the time function value for time $t \leq \text{ARTM}$ is zero and hence a thermal strain will result if the initial temperature is nonzero. In order to obtain a zero thermal strain prior to the arrival time, multiple temperature data sets can be assigned to the same nodal points. For example, when temperatures are prescribed directly to the nodes (not through geometry), this can be done as shown in Fig. 5.6-1.

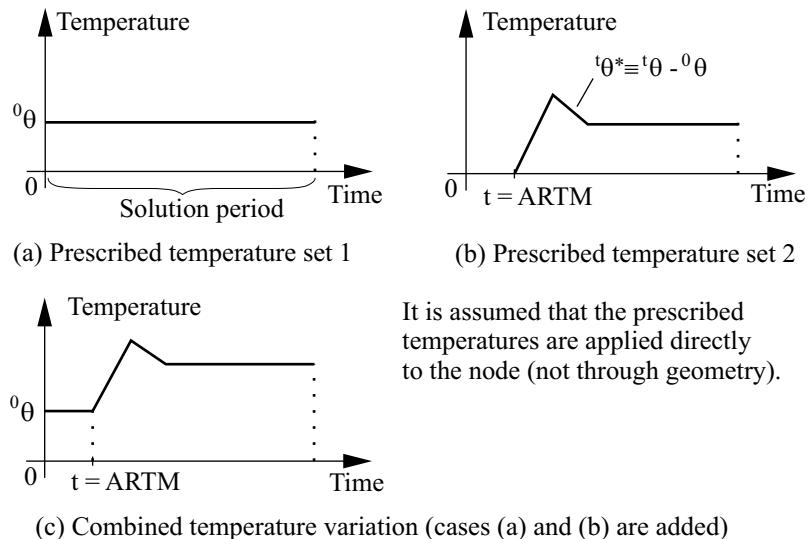


Figure 5.6-1: Use of the arrival time option for prescribed temperatures

- If temperatures are obtained both from a separate file and from direct input, the final temperatures used in the analysis are obtained by adding the temperatures from the file and the direct input.
- Temperature gradients at nodes can be read from a file and/or from direct input in a way analogous to the nodal temperature input.
- The input of prescribed temperatures and temperature gradients is completely separate from the input of initial temperatures and temperature gradients. See Section 11.6 for the input of initial temperatures and temperature gradients.

5.7 Pipe internal pressure data

- If pipe elements are used, then pipe internal pressures can be input as a load on the structure (see Section 2.8.2). The pipe internal pressure is a nodal point variable which must be defined for all nodes throughout the solution period.

- The time variation of the nodal values of the internal pressure is specified by the use of time functions.
- Pipe internal pressures can also be read from a separate user-supplied file. The format of this file is analogous to the file format used for reading nodal temperatures (see Section 5.6).
- If pipe internal pressures are specified by time functions and also read from a file, the final pipe internal pressures are obtained by adding the separate contributions.

5.8 Electromagnetic loading

- The interaction effects of short-circuit currents in cables and beams can be modeled with ADINA using the electromagnetic loading option. Cables and beams can be modeled with truss and beam elements.
- Considering two electric conductors traversed by short-circuit currents $i_1(t)$ and $i_2(t)$ and placed in a medium of magnetic permeability μ_0 , each differential element $d\mathbf{r}_2$ of one conductor exerts on a differential element $d\mathbf{r}_1$ of the other conductor a magnetic force given by Grassman's formula (see Fig. 5.8-1):

$$d\mathbf{F}_{21} = \frac{\mu_0}{4\pi} i_1(t) i_2(t) \frac{d\mathbf{r}_1 \times (d\mathbf{r}_2 \times \mathbf{r}_{21})}{\|\mathbf{r}_{21}\|^3}$$

where \mathbf{r}_{21} is the vector between the middle of the differential elements $d\mathbf{r}_1$ and $d\mathbf{r}_2$. Integration of $d\mathbf{F}_{21}$ along a finite element yields the total electromagnetic force on the element, half of which is assigned to each end node.

- When entering this loading in the AUI, it is necessary to apply the load onto each of the electric conductors. For the load on cable k , the time function multiplied by the load multiplier should equal

$$\sqrt{\frac{\mu_0}{4\pi}} i_k(t).$$

- 3-point Gauss integration is used for all numerical integrations.

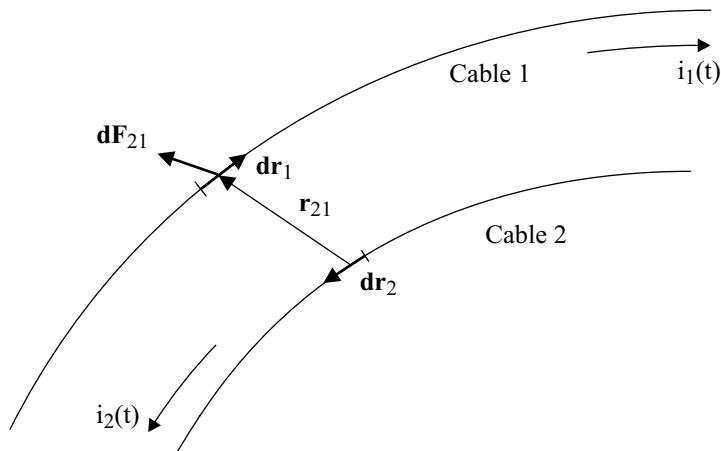


Figure 5.8-1: Conventions for electromagnetic loading

5.9 Poreflow loads

- The flux of pore fluids in porous media can be specified using poreflow loads. See Section 3.12 for more information about analysis of porous media.
- The flux can either be concentrated (that is, applied directly to nodes), or distributed (that is, applied to the boundaries of 2-D and 3-D elements).

5.10 Phiflux loads

- The flux of fluids in potential-based fluid elements can be specified using phiflux loads. See Section 2.11 for more information about the potential-based fluid elements and Section 2.11.14 for more information about loads applied to potential-based fluid elements.
- The flux can either be concentrated (that is, applied directly to nodes), or distributed (that is, applied to the boundaries of 2-D and 3-D elements).

- Positive mass flux is assumed to represent mass flowing into the fluid domain.

5.11 Contact slip loads

- The contact slip loading enables you to add a tangential or slip velocity component between contacting surfaces (see Figure 5.11-1). The contact slip is defined by an axis of rotation, an angular velocity and a time function. The total slip velocity between two contacting surfaces is the vectorial sum of the actual slip resulting from the relative displacement between the surfaces and this contact slip load. The total slip velocity is used to evaluate stick and slip and is also used to determine the amount of heat generated due to friction. In 2-D analysis, the slip velocity component can be out-of-plane (i.e. in the x-direction). Note that in static analysis the tangential velocity is equal to the tangential displacement divided by the time variable.

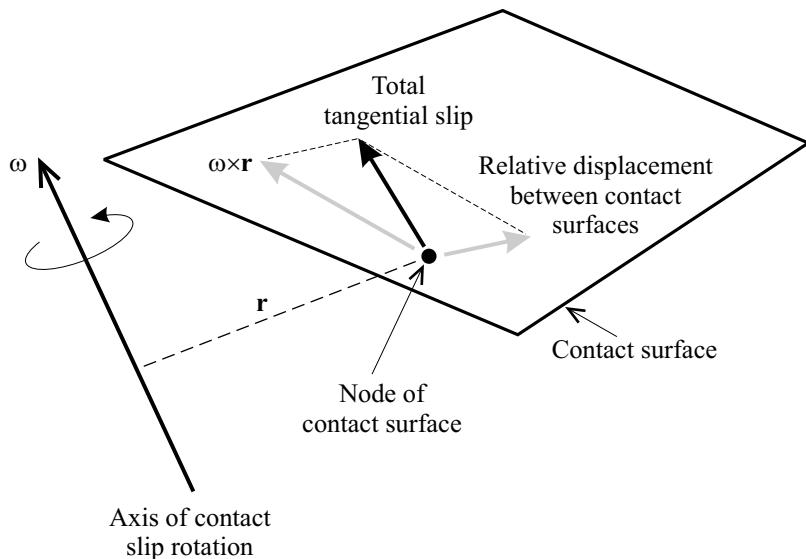


Figure 5.11-1: Contact slip loading

- Contact slip loading can be applied to 2-D and 3-D contact surfaces.

5.12 Surface tension boundary

- A constant surface tension value can be applied to the boundaries of 2-D and 3-D elements as shown in Figure 5.12-1. A similar boundary condition for fluids exists in ADINA-F.

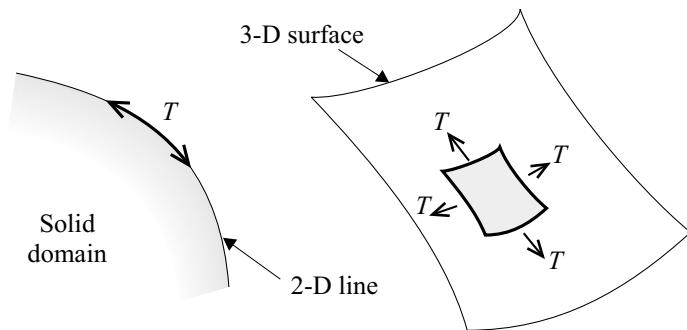


Figure 5.12-1: Surface tension applied to 2-D and 3-D boundaries

- In 2-D analysis, the surface tension boundary can be applied to geometry lines or element edges. In 3-D analysis, the surface tension boundary can be applied to geometry surfaces or element faces.
- The surface tension boundary is converted by the AUI into equivalent ADINA elements when the ADINA data file is generated. The ADINA elements have an initial stress equal to the magnitude of the surface tension, T . In 2-D analysis, isobeam elements are used. In 3-D analysis, 2-D generalized plane stress elements are used. When post-processing the results, the AUI automatically converts the ADINA elements back into a surface tension boundary.
- The surface tension option can be used with the **small displacement/small strain**, **large displacement/small strain** and **large displacement/large strain** formulations. When large displacements are selected, the surface tension is treated as a true (Cauchy) stress.

5.13 Initial load calculations

- Considering temperature, temperature gradient and pipe internal pressure loads, see Section 11.6 where the initial conditions are discussed.
- Considering all other externally applied loading, the loads corresponding to time TSTART are not used and not calculated, except when the ATS method is employed. When the ATS method is employed, the loads are calculated either directly from the time functions, or, in the case of fluid-structure interaction with ADINA-FSI, from the initial displacements, initial velocities, initial accelerations and initial strains.

5.14 User-supplied loads

- An option for user-supplied loads is available in ADINA to allow the modeling of very general types of loading. For example, loads can be defined as functions of nodal coordinates, displacements, velocities, accelerations, temperatures, times, etc.

5.14.1 General considerations

- In order to specify the loads, you must provide the desired definitions of the loads in the form of FORTRAN coding which is inserted in the specially provided subroutines in ADINA.
- The user-supplied loads applied at a node can be a function of the various nodal quantities at the same node, or they can be a function of nodal quantities at other nodes, as described in detail below.
- The presence of user-supplied loads requires in general the use of equilibrium iterations in the response solution, see Section 7.2.

5.14.2 Usage of the user-supplied loads

- User-supplied loads can be specified in static or dynamic, linear or nonlinear analysis. However, the definition of the loads must be

consistent with the type of analysis; e.g., the loads cannot be a function of the velocities in a static analysis.

- User-supplied loads cannot be specified if the LDC method is employed (see Section 7.2.1 for information about the LDC method). In addition, user-supplied loads cannot be applied to any substructures; however, these loads can be applied to the main structure to which substructures are coupled.
- During the step-by-step response calculation, the user-supplied loads are evaluated in each time step and in each iteration. These loads are added to the effective load vector normally established in the response calculation for the time steps.

Evaluation of the user-supplied loads: You must modify the coding in the user-supplied subroutines IUSER and USERSL in order to select the nodes whose nodal variables are used in the load calculation, to select the nodes to which the loads are applied, and to calculate the loads.

Sample coding is provided in file ovl170u.f. This coding corresponds to the example given in Section 5.14.3 below.

- You enter some control parameters in the AUI in the specification of the user-supplied loading (in the command-line interface, the command is APPLY USER-SPECIFIED-LOADS). If loads calculated at a given node M depend only upon variables at the node M, set parameter NODE-DEPENDENCE to 1; if loads calculated at a given node M depend upon variables at other nodes, set parameter NODE-DEPENDENCE to 2.

You also can define integer and floating point constants in the AUI for use in the user-supplied loading subroutines:

NRCONS: Number of floating point constants (maximum of 100).

NICONS: Number of integer constants (maximum of 100).

RCONS(NRCONS): Floating point constants. In subroutine USERSL, this corresponds to vector ULDATA(100).

ICONS(NICONS): Integer constants. In subroutine USERSL, this corresponds to vector KULINT(100).

NODE-DEPENDENCE=1: If NODE-DEPENDENCE=1, the loads applied at a node M can depend only on nodal variables at the same node.

Node selection using subroutine IUSER: Subroutine IUSER is called once for each nodal point each time the effective load vector is calculated. In this case MFLAG should be set to 1 only for nodes at which loads are to be applied.

Load calculation using subroutine USERSL: Subroutine USERSL is called once for each node selected by subroutine IUSER each time the effective load vector is calculated.

Note that for each node M, only the non-zero components of the new additional load vector need to be calculated. ADINA distinguishes internally independent, dependent or fixed degrees of freedom and performs the proper summation of the new additional load vector to the effective load vector.

NODE-DEPENDENCE=2: If NODE-DEPENDENCE=2, then the loads applied at a node M depend on the nodal variables at a number of other nodes N.

Node selection using subroutine IUSER: Two flags, MFLAG and NFLAG, are used in subroutine IUSER to select the appropriate nodes for the load calculation, as schematically shown.

```
DO 100 M=1,NUMNP
DO 200 N=1,NUMNP
.
.
.
CALL IUSER (M,N,NUMNP,MFLAG,NFLAG)
IF (MFLAG.EQ.0) GO TO 100
IF (NFLAG.EQ.0) GO TO 200
.
.
.
CALL USERSL(...)
```

```
.  
. .  
200 CONTINUE  
100 CONTINUE
```

Note that subroutine USERSL is only called for nodes selected by the flags set in subroutine IUSER.

Load calculation using subroutine USERSL: Each time the effective load vector is calculated, for each node M at which the user-supplied loads are applied, subroutine USERSL is called once for N=N1,N2 ..., NN where N1, N2,..., NN are arranged in ascending order and correspond to the nodes whose nodal variables are used to calculate the loads at node M. Nodes N1, N2,..., NN are specified in subroutine IUSER as described above.

Note that for the load calculations at node M, the required nodal variables must be stored in auxiliary working arrays when N.LT.NN, and the actual load computation must be performed only when N.EQ.NN. The arrays WA(6), WB(6), ... WF(6) are provided for this purpose.

Note that for each node M, only the nonzero components of the additional load vector need to be calculated. ADINA distinguishes internally independent, dependent or fixed degrees of freedom and performs proper summation of the new additional load vector to the effective load vector.

5.14.3 Example: Hydrodynamic forces

- For a cylinder moving with velocity $\dot{\mathbf{U}}_c(t)$ and acceleration $\ddot{\mathbf{U}}_c(t)$ in the direction of an oncoming fluid stream of velocity $\dot{\mathbf{U}}_f(\mathbf{x}, t)$, the total hydrodynamic force per unit length due to drag and inertial effects, excluding diffraction effects, can be approximated by Morison's general formula:

$$\begin{aligned}\mathbf{F}_c = & 0.5D\rho C_D (\dot{\mathbf{U}}_f - \dot{\mathbf{U}}_c) \|\dot{\mathbf{U}}_f - \dot{\mathbf{U}}_c\| \\ & + 0.25\pi D^2 \rho (1+C_A) \left[\frac{\partial \dot{\mathbf{U}}_f}{\partial t} + (\dot{\mathbf{U}}_f - \dot{\mathbf{U}}_c) \cdot \frac{\partial \dot{\mathbf{U}}_f}{\partial \mathbf{x}} \right] \quad (5.14-1) \\ & - 0.25\pi D^2 \rho C_A \ddot{\mathbf{U}}_c\end{aligned}$$

where ρ is the density of the fluid, D is the diameter of the cylinder and C_D and C_A are the drag and added mass coefficients of the cylinder.

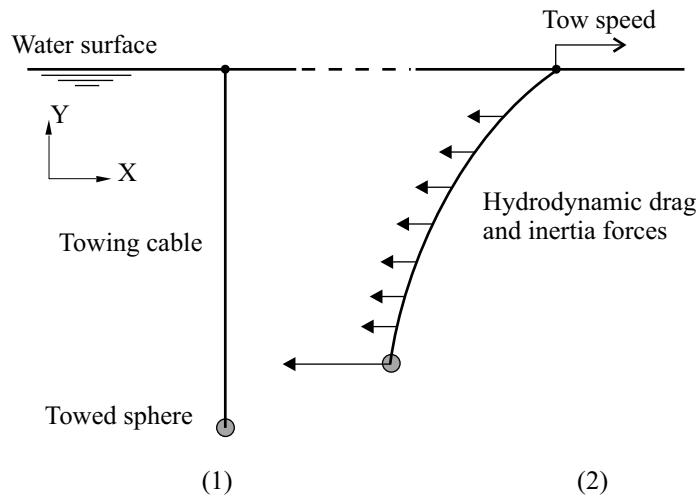
In the right-hand-side of (5.14-1), the first term represents the form drag, the second term corresponds to inertial forces due to the local and convective accelerations of the fluid about the cylinder and the third term corresponds to the inertial force due to the motion of the cylinder.

Morison's equation is commonly used for the analysis of submerged and partially submerged structures such as offshore rigs, risers, underwater pipes and cables.

More information about Morison's equation and about the drag and inertia coefficients C_D and C_A (which depend upon many parameters, including the section shape and surface roughness of the cylinder and the Reynolds number) can be found in the following reference:

ref. Sarpkaya, T. and Isaacson, M., *Mechanics of Wave Forces on Offshore Structures*, Van Nostrand Reinhold Co., New York, 1981.

- In ADINA, the user-supplied load interface has been used to implement the calculation of the Morison forces in the case of a sphere suspended at the bottom of a cable, and towed in a still fluid $\dot{\mathbf{U}}_f(\mathbf{x}, t) = \mathbf{0}$ (see Figure 5.14-1). The model must consist of one line of elements, with possibly concentrated masses and/or dampers at the nodes. The nodes must be numbered 1, 2, ..., N . At the bottom node N , a sphere generating additional drag and inertia forces can be added. The following items can be specified: the length of the cable, the fluid density, the diameter of the sphere, the cable drag and added mass coefficients and the sphere drag and added mass coefficients.



(1) Initial geometry

(2) Structure submitted to gravity and hydrodynamic forces

Figure 5.14-1: Usage of the user-supplied load subroutines
for the analysis of a sphere towed underwater

- The hydrodynamic force on a unit length of the cylinder (velocity $\dot{\mathbf{U}}_c$, acceleration $\ddot{\mathbf{U}}_c$) is calculated using

$$\mathbf{F}_c = -0.5D\rho C_D \dot{\mathbf{U}}_c \|\dot{\mathbf{U}}_c\| - 0.25\pi D^2 \rho C_A \ddot{\mathbf{U}}_c$$

where ρ is the density of the fluid, D is the diameter of the cylinder and C_D and C_A are the drag and added mass coefficients of the cylinder.

- The hydrodynamic force on the sphere (velocity $\dot{\mathbf{U}}_s$, acceleration $\ddot{\mathbf{U}}_s$) is calculated using

$$\mathbf{F}_s = -0.5 \left(\frac{D}{2}\right)^2 \rho C_D \dot{\mathbf{U}}_s \|\dot{\mathbf{U}}_s\| - \frac{4}{3}\pi \left(\frac{D}{2}\right)^3 \rho C_A \ddot{\mathbf{U}}_s$$

where ρ is the density of the fluid, D is the diameter of the sphere and C_D and C_A are the drag and added mass coefficients of the sphere.

- Either a static or a dynamic analysis can be performed with ADINA. In a static analysis, the velocity at the top of the cable must be entered.
- Verification problem B.130 in the ADINA Verification Manual demonstrates the usage of this user-supplied load.

5.15 Constraint equations

5.15.1 General considerations

- ADINA can impose two basic types of constraints between nodal degrees of freedom. The first type has the form

$$U_k = \sum_j \beta_j U_j \quad (5.15-1)$$

where U_k is a dependent (or slave) degree of freedom which is controlled by multiple independent (master) U_j degrees of freedom via factors β_j . Alternatively, a general constraint can be defined as

$$\sum_j \beta_j U_j = 0 \quad (5.15-2)$$

where none of the U_j degrees of freedom are made dependent (there are no slave degrees of freedom). In this case, the constraint is imposed using Lagrange Multipliers.

Note that each constraint of the first type reduces the number of independent degrees of freedom by one, while each general constraint of the second type increases the number of degrees of freedom by one (by adding a Lagrange Multiplier). Hence, the first type should be used whenever possible. In some cases, however, one cannot express a constraint in the form of Eq. (5.15-1) where a slave degree of freedom is only related to master degrees of

freedom. See the example below for an illustration.

$$\begin{aligned} U_{2(z)} &= U_{1(z)} + 3\theta_{1(x)} && \leftarrow \text{valid constraint} \\ U_{3(z)} &= U_{2(z)} + U_{1(z)} - 4\theta_{2(x)} && \leftarrow \text{invalid constraint} \\ &\quad \uparrow \\ \text{This DOF is already dependent} \end{aligned}$$

The mesh glueing feature (Section 5.16) automatically produces general constraints which are expressed in the form of Eq. (5.15-2).

- Displacement constraints can be applied to static and dynamic analyses.
- The basic constraints constraints of Eq. (5.15-1) are only approximately satisfied in an explicit analysis, since imposing the constraint exactly requires a non-diagonal mass matrix.
- General constraints cannot be used in explicit analysis.
- It is important to note that adding constraint equations to the finite element model results in adding external forces (and possibly moments) at the degrees of freedom specified by the constraint equations. These forces are included in the reaction calculations.

5.15.2 Rigid links

- Rigid links are special constraint equations established automatically by the program between two nodes – a master node and a slave node.

As the nodes displace due to deformation, the slave node is constrained to translate and rotate such that the distance between the master node and the slave node remains constant, and that the rotations at the slave node are the same as the corresponding rotations at the master node.

- Rigid links can be kinematically linear or nonlinear. For linear rigid links, the constraint equations are unchanged throughout the solution and hence they are applicable for small displacement analysis.

In a large displacement analysis, nonlinear rigid links should be used recognizing that the effects of large displacements are taken into consideration in updating the constraint equations for the master and slave nodes. Hence, the coefficients in Eq. (5.15-1) are no longer constant.

- Rigid links are useful in the modeling of stiffened shells, for example, in coupling iso-beam elements (see Section 2.5) to shell elements (see Section 2.7).
- The tied contact feature discussed in Section 4.1.1 internally creates rigid links between its two contact surfaces.
- When a rigid link is attached to a node of a shell element, 6 degrees of freedom must be used for the node.

The rotation around the shell node director vector may have zero stiffness, depending upon the modeling of the other end of the rigid link (see Figure 5.15-1). There is an option to assign stiffness to those shell node rotational degrees of freedom with zero stiffness that are attached to rigid links.

Shell nodes attached to rigid links must have 6 DOF.

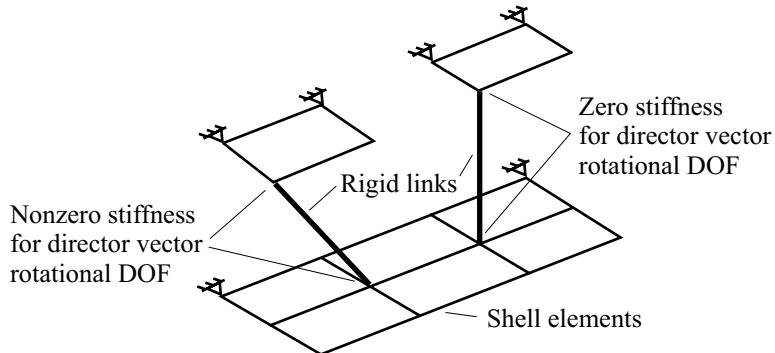


Figure 5.15-1: Rigid links and shell elements

- You can use fixities on nodes connected to rigid links, provided that the fixities not prevent the rigid link from maintaining constant distance between the nodes.

- Nonlinear rigid links cannot be used in conjunction with the Wilson theta method of time integration.

5.16 Mesh Glueing

- The glueing feature is used to attach two surfaces (or lines in 2-D) together. These two surfaces usually involve different finite element meshes (see Fig. 5.16-1). The glueing procedure should lead to smooth transition of displacements and tractions between the glued surfaces. This feature is useful for several applications:
 - ▶ When a fine mesh is desired in a certain region and coarser meshes are desired in other regions.
 - ▶ When different regions are meshed independently with unstructured free meshes.
 - ▶ When different regions are meshed with different element types (such as a tetrahedral mesh attached to a brick mesh).

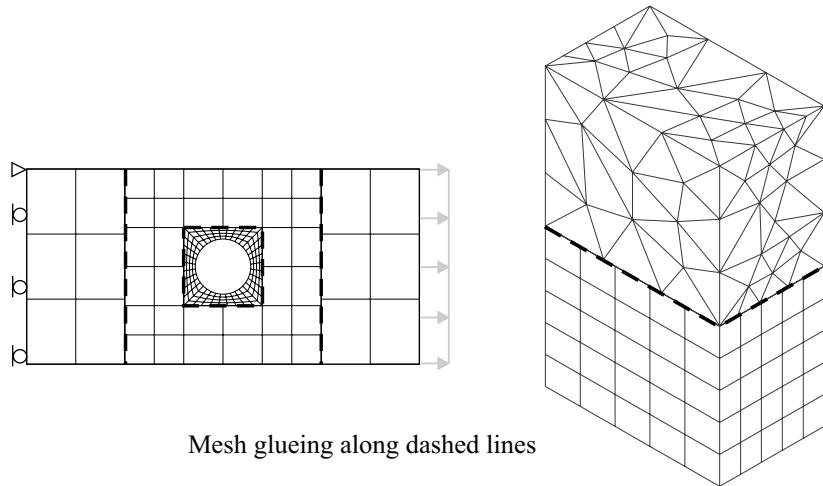


Figure 5.16-1. Examples requiring mesh glueing

- The proper glueing constraint between the two surfaces can be expressed as

$$\int_{\Gamma} \lambda \cdot (u^1 - u^2) d\Gamma = 0 \quad (5.16-1)$$

where u^1 is the displacement of the first glued surface, u^2 is the displacement of the second surface and λ is the Lagrange Multiplier field imposing the constraint.

One of the glued surfaces is designated as the master and the other as the slave. The Lagrange Multiplier field involves nodal degrees of freedom at the nodes of the slave surface, and the integration is also performed over the slave surface. Hence Eq. (5.16-1) becomes

$$\int_{\Gamma^S} \lambda^S \cdot (u^M - u^S) d\Gamma^S = 0 \quad (5.16-2)$$

The accurate integration of Eq. (5.16-2) is not trivial since the displacements u^M and u^S are generally interpolated over different domains. This integration is performed by the AUI and results in multiple general constraints (see Section 5.15.1).

- Mesh glueing is not available in explicit analysis.
- Only 2-D solid elements can be used in the gluing of 2-D lines. The glued element side can have linear or quadratic displacement interpolation. Truss, beam and pipe elements are not supported.
- Only 3-D solid elements can be used in the gluing of 3-D surfaces. The glued element faces can be triangles or quads, and they can have linear or quadratic displacement interpolation. Plate and shell elements are not supported.
- Each node on the slave surface must locate a target segment on the master surface. Therefore, if one surface is smaller than the other, as shown in Fig. 5.16-2, the smaller surface should be the slave.

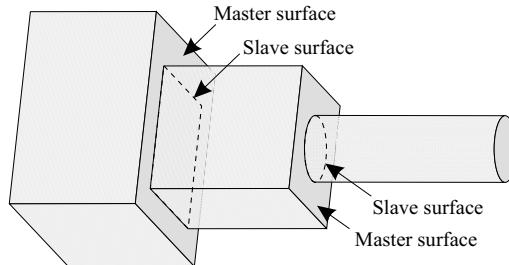


Figure 5.16-2. Selection of master and slave glue surfaces

- The two glued surfaces should ideally be smooth surfaces (no sharp corners). If corners exist it is better to create multiple glued meshes, as shown in Fig. 5.16-3.
- The Lagrange Multiplier field is modified at nodes where multiple glued meshes intersect. These nodes are called cross-over points.

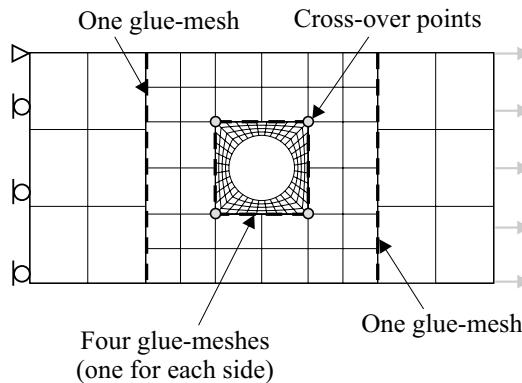


Figure 3. Glueing non-smooth surfaces

- If the two surfaces have different mesh densities, both can be used as slave (unlike contact where the finer one should be the contactor). Using the finer meshed surface as a slave will produce more equations, since the Lagrange Multipliers degrees of freedom are on the nodes of the slave surface.

6. Eigenvalue problems

*ref. KJB
Section 6.8.2
and Chapter 10*

The two types of analyses in ADINA for which the solution is obtained by solving the corresponding generalized eigenvalue problem are:

- ▶ Linearized buckling analysis
- ▶ Frequency analysis

These two types of analyses are discussed in the following sections.

6.1 Linearized buckling analysis

6.1.1 General considerations

- In a linearized buckling analysis, the eigenvalue problem to be solved is either (secant formulation)

$${}^{t_1} \mathbf{K} \phi_i = \gamma_i {}^{t_0} \mathbf{K} \phi_i; \quad \gamma_i > 0$$

or (classical formulation)

$${}^{t_1} \mathbf{K} \phi_i = \gamma_i \left({}^{t_1} \mathbf{K} - {}^{t_1} \mathbf{K}_{NL} \right) \phi_i, \quad \gamma_i > 0$$

where ${}^{t_1} \mathbf{K}$ and ${}^{t_0} \mathbf{K}$ are the stiffness matrices of the structural systems at times t_1 and t_0 , respectively, ${}^{t_1} \mathbf{K}_{NL}$ is the geometrically nonlinear part of ${}^{t_1} \mathbf{K}$, t_0 is the time at solution start, t_1 equals $t_0 + \Delta t$ where Δt is the time step increment, ϕ_i is the buckling mode shape for mode i and the critical (buckling) load factor λ_i for mode i is related to the eigenvalue γ_i using

$$\lambda_i = \frac{1}{1 - \gamma_i}$$

ref. K.J. Bathe and E.N. Dvorkin, "On the Automatic

Solution of Nonlinear Finite Element Equations,"
Computers & Structures, Vol. 17, No. 5-6, pp. 871-879,
1983.

- The classical buckling formulation can compute classical buckling loads more accurately than the secant formulation, especially in problems that include initial strains.
- The eigenvalues are extracted in numerically ascending sequence. If the following eigenvalues are obtained from the eigenvalue solution:

$$\gamma_1 \leq \gamma_2 \leq \dots \leq \gamma_M < 1 < \gamma_{M+1} \leq \dots \leq \gamma_N$$

where N is the number of buckling modes requested, then the following inequalities hold for the critical load factors:

$$1 < \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_M$$

and

$$\lambda_{M+1} \leq \lambda_{M+2} \leq \dots \leq \lambda_N < 0$$

- In the linearized buckling analysis, the critical (buckling) load vector is determined by the critical load factor using

$$\mathbf{R}_{cri} = {}^{t_0} \mathbf{R} + \lambda_i \left[{}^{t_1} \mathbf{R} - {}^{t_0} \mathbf{R} \right] \text{ for mode } i$$

where ${}^{t_1} \mathbf{R}$ and ${}^{t_0} \mathbf{R}$ are the externally applied load vectors at t_1 and t_0 , respectively.

- The λ_1 value obtained should satisfy $1 < \lambda_1 < 500$. If this condition is violated, the program issues an error message and stops the execution, and you must adjust the applied load level for the first time step to successfully complete the analysis.
- At the completion of the eigenvalue solution, an error bound for each eigenvalue and eigenvector is calculated using

$$\left\{ \begin{array}{l} \text{Error} \\ \text{bound} \\ \text{for} \\ \text{mode } i \end{array} \right\} = \left[1 - \frac{(\lambda_i^{(k)})^2}{(\mathbf{q}_i^{(k)})^T \mathbf{q}_i^{(k)}} \right]^{\frac{1}{2}}$$

where $\mathbf{q}_i^{(k)}$ is the vector of the matrix \mathbf{Q}_k corresponding to $\lambda_i^{(k)}$ (see ref. KJB).

- In order to perform a linearized buckling analysis at step n , where $n > 1$, the ADINA restart option can be employed. In the first run, the nonlinear static response is solved for $n - 1$ steps. In the restart run, a linearized buckling analysis can be specified to evaluate the buckling response linearized at step n .
- The buckling modes obtained in a linear analysis can be used in subsequent analyses as input for initial imperfections in the structural system; see Section 6.1.2.
- Note that linearized buckling analysis must be specified in ADINA as a static analysis. In addition, only the subspace iteration method can be used in the eigenproblem solution.
- In the eigenvalue extraction, convergence of the eigenvalues γ_i is measured using

$$\frac{\gamma_i^{(k+1)} - \gamma_i^{(k)}}{\gamma_i^{(k+1)}} \leq \text{SSTOL}; \quad i = 1, 2, \dots, N$$

where N is the number of eigenmodes requested, k is the iteration counter and SSTOL is a user-specified convergence tolerance.

- If termination of solution occurs without attaining convergence, you should rerun the analysis using:
 - a larger load level for the first time step; and
 - a larger number of iteration vectors.
- A Sturm sequence check is always performed after the subspace iterations have converged to verify that indeed the smallest

required eigenvalues and the corresponding mode shapes have been calculated. If this check fails, the program prints an approximation (λ_p) to the actual critical load factor (λ_1), where $\lambda_1 < \lambda_p$.

In order to perform a successful analysis, you should rerun the analysis using (a) and (b) above.

6.1.2 Introduction of geometric imperfections

- In a collapse analysis, the collapse loads can significantly depend on the geometric imperfections of the structure. A natural way to introduce appropriate geometric imperfections in a finite element analysis is to first perform a linearized buckling analysis and then superimpose on the "perfect" geometry of the finite element model a coordinate perturbation proportional to the buckling mode shapes. The collapse analysis is then effectively performed using the LDC solution scheme on the model of the geometrically imperfect structure.
- In ADINA, the mode shapes can be readily used to generate geometric imperfections.

6.2 Frequency analysis

- The natural frequencies and mode shapes of vibration of the structural system are calculated using

$$\mathbf{K}\phi_i = \omega_i^2 \mathbf{M}\phi_i$$

where \mathbf{K} is the system stiffness matrix corresponding to the time of solution start, \mathbf{M} is the system mass matrix corresponding to the time of solution start, ω_i and ϕ_i are the angular frequency and mode shape, respectively, for mode i . Note that the frequencies are extracted in the eigenvalue solution in numerically ascending sequence. The eigenvectors are M-orthonormal, that is

$$\phi_i^T \mathbf{M} \phi_i = 1$$

- If the structure is nonlinear and is preloaded, you must perform

the analysis in two runs. In the first run, you apply the preload (gravity loads, thermal loads, initial strains, etc.) in a nonlinear static analysis. This run can contain one or more solution steps.

In the second run, you restart from the last solution of the first run, then perform a frequency analysis. The frequency analysis is, as stated above, performed based on the stiffness and mass matrix corresponding to the time of solution start; in this case the time of solution start is the time corresponding to the restart analysis.

Therefore the stiffness and mass matrices include all geometric and material nonlinearities corresponding to the end of the nonlinear static analysis.

- At the completion of the eigenvalue solution, an error bound for each eigenvalue and eigenvector is calculated using

$$\left\{ \begin{array}{l} \text{Error} \\ \text{bound} \\ \text{for} \\ \text{mode } i \end{array} \right\} = \left[1 - \frac{(\lambda_i^{(k)})^2}{(\mathbf{q}_i^{(k)})^T \mathbf{q}_i^{(k)}} \right]^{\frac{1}{2}}$$

where $\mathbf{q}_i^{(k)}$ is the vector of the matrix \mathbf{Q}_k corresponding to $\lambda_i^{(k)}$ (see ref. KJB). The “physical” error bound for each eigenvalue and eigenvector is calculated using

$$\left\{ \begin{array}{l} \text{Physical} \\ \text{error} \\ \text{bound} \\ \text{for mode } i \end{array} \right\} = \frac{\|\mathbf{K}\phi_i - \omega_i^2 \mathbf{M}\phi_i\|}{\|\mathbf{K}\phi_i\|}$$

where $\|\mathbf{a}\|$ is the Euclidean norm of vector \mathbf{a} .

- If there are frequencies that are numerically zero, the eigensolution procedures can still calculate the frequencies and corresponding mode shapes. Frequencies that are numerically zero are present in the model if all rigid body modes have not been suppressed, or if there are input or modeling errors. The number of rigid body modes in the model is equal to the number of zero frequencies.

In order to obtain a frequency solution when there are zero

frequencies, you must activate the "allow rigid body modes" option, otherwise ADINA will stop. When you have activated the rigid body mode shift option, you can also enter the rigid body mode shift (which must be negative). If you do not enter a rigid body mode shift, ADINA calculates it using the formula

$$\text{RBMSH} = f \min_{i=1,\dots,N} \left[\frac{k_{ii}}{m_{ii}} \right]$$

where $f = -10^{-3}$ for the determinant search method or if contact is present, $f = -10^{-7}$ otherwise and N is the number of mass degrees of freedom in the finite element model.

- The frequencies and mode shapes of vibration can be calculated using the following methods: **determinant search, subspace iteration** or **Lanczos iteration**.

6.2.1 Use of determinant search method

*ref. KJB
Section 11.4*

- The use of the determinant search method is usually effective when
 - ▶ A narrow banded finite element system is considered.
 - ▶ The total stiffness matrix can be processed in-core.
- The determinant search method calculates the smallest frequencies and corresponding mode shapes of the system.

6.2.2 Use of subspace iteration method

*ref. KJB
Section 11.6*

- The subspace iteration method used in ADINA contains all the capabilities of the original basic subspace iteration method described in
 - ref. K.J. Bathe, "Solution Methods of Large Generalized Eigenvalue Problems in Structural Engineering," *Report UC SESM 71-20*, Univ. of California, Berkeley, 1971.

plus the acceleration techniques developed more recently:

ref. K.J. Bathe and S. Ramaswamy, "An Accelerated Subspace Iteration Method," *J. Computer Meth. in Applied Mech. and Eng.*, Vol. 23, pp. 313-331, 1980.

- When the acceleration techniques are employed, the method can be significantly more effective. These techniques consist of
 - ▶ The use of the Lanczos method to generate the starting subspace
 - ▶ Vector over-relaxation in the vector iterations
- The subspace iteration can be employed to calculate the smallest frequencies and corresponding mode shapes, or the frequencies (and mode shapes) within a certain prescribed interval.
- When a relatively large number of frequencies needs to be calculated, it can be effective to use fewer iteration vectors than the number of frequencies sought. For example, to calculate the 60 smallest frequencies, it may be effective to employ only 10 iteration vectors. In this case the total number of frequencies is calculated in groups by shifting through the band of frequencies to be solved.
- If q iteration vectors are used and there are only q mass degrees of freedom, the subspace iteration method corresponds to a static condensation (Guyan reduction) and subsequent solution of the eigenvalues.

ref. KJB
Sections 10.3.1
10.3.2 and
11.6.3

This analysis feature may be useful when you assign zero mass density to the material, and account for the mass by means of concentrated masses corresponding to a few degrees of freedom only. However, note that such mass lumping requires much experience with the problem to be solved and the finite element model used.

ref. KJB
Section 11.5

- The standard Lanczos method calculates only approximations to the eigenvalues and eigenvectors, and for this reason it is here only used to generate the starting subspace. If the Lanczos method is employed, it is usually effective to use a number of iteration vectors q considerably larger than the number of frequencies

required p , i.e., use $q = 2p$, with q still much smaller than the total number of frequencies of the finite element system.

- ADINA can perform a Sturm sequence check after the subspace iterations have converged to verify that the smallest required eigenvalues and corresponding mode shapes have been calculated. The cost of the Sturm sequence check is approximately equal to the cost of a static solution.

6.2.3 Use of Lanczos iteration method

*ref. KJB
Section 11.5*

- The Lanczos method can be used to calculate the smallest frequencies and corresponding mode shapes. The method is very effective for large problems.
- When potential-based fluid elements are present, the Lanczos method should be used. (For small problems, the determinant search method can also be used; for large problems, the determinant search method is ineffective.)
- The convergence of the iteration is assessed by the criterion

$$\frac{\gamma_i^{(k-1)} - \gamma_i^{(k)}}{\gamma_i^{(k-1)}} \leq \varepsilon; \quad i = 1, \dots, N$$

where N is the number of eigenmodes requested, k is the Lanczos iteration stage counter and the value $\varepsilon = 10^{-9}$ is used. See ref. KJB, Section 11.5 for all notation not described here.

- After convergence, the error bound $|\beta_q s_{qi}|$ and the physical error estimate $\frac{\|\mathbf{K}\phi_i - \omega_i^2 \mathbf{M}\phi_i\|}{\|\mathbf{K}\phi_i\|}$ are calculated.
- ADINA can perform a Sturm sequence check after the Lanczos iterations have converged to verify that the smallest required eigenvalues and corresponding mode shapes have been calculated.

When ADINA performs a Sturm sequence check, ADINA also

calculates the M-orthogonal checks

$$\text{DII} = \max_{1 \leq i \leq n} \left| \left(\phi_i^T \right)^{t_0} \mathbf{M}(\phi_i) - 1 \right|, \quad \text{DIJ} = \max_{\substack{1 \leq i, j \leq n \\ i \neq j}} \left| \left(\phi_i^T \right)^{t_0} \mathbf{M}(\phi_j) \right|$$

to verify that the eigenvectors are M-orthogonal.

- The sparse solver is always used as the equation solver in the Lanczos iteration method.
- For potential-based fluid elements, when a relatively large number of frequencies needs to be calculated, it can be effective, and sometimes necessary, to use the shifting procedure. The option to use the shifting procedure is controlled by the parameters NSHIFT and NSHIFT-BLOCK in the FREQUENCIES command.

6.2.4 Modal stresses

- Modal strains, stresses, forces and reactions can be requested in ADINA. To calculate these quantities, small displacements and small strains are assumed. Modal displacements and modal stresses are therefore proportional.
- A frequency analysis must be performed first to obtain the modal displacements. This analysis can follow a materially or geometrically nonlinear analysis.
- In linear analysis, the modal stresses are directly calculated from the M-orthonormal modal displacements ϕ_i , where

$$\phi_i^T \mathbf{M} \phi_i = 1.$$

- In materially or geometrically nonlinear analysis, the modal stresses for mode i ($1 \leq i \leq N$) are calculated using the approximation

$$\sigma_i^m = \frac{\left(\sigma_i \left(\mathbf{U}_i^m \right) - {}^t \boldsymbol{\sigma} \right)}{\alpha_i}$$

where $\sigma_i \left(\mathbf{U}_i^m \right)$ are the stresses computed from the perturbed

displacement vector $\mathbf{U}_i^m = {}^t\mathbf{U} + \alpha_i \boldsymbol{\phi}_i$, ${}^t\boldsymbol{\sigma}$ is the stress vector at time t , and α_i is a perturbation factor, controlled by parameter DUSIZE as follows.

If DUSIZE is not equal to 0, then $\alpha_i = \frac{\text{DUSIZE}}{\|\boldsymbol{\phi}_i\|}$; this choice

implies $\|\mathbf{U}_i^m - {}^t\mathbf{U}\| = \text{DUSIZE}$. Here the symbol $\|\cdot\|$ is the norm of the enclosed vector only including translations. DUSIZE is seen to be the size of the displacement perturbation.

If DUSIZE is equal to 0, then ADINA computes

$$\alpha_i = 0.001 \frac{\|{}^t\mathbf{U}\|}{\|\boldsymbol{\phi}_i\|} \text{ where } {}^t\mathbf{U} \text{ is the total displacement vector at time } t$$

(last step of analysis before frequency analysis). This choice implies $\|\mathbf{U}_i^m - {}^t\mathbf{U}\| = 0.001 \|{}^t\mathbf{U}\|$, which is usually quite reasonable. In the special case in which the total displacement vector is zero (for example, if the modal stress calculation is requested at the start of the analysis), then ADINA sets α_i to 0.001.

The choice $\alpha_i = 0.001$ can lead to very large or very small displacement perturbations because then $\|\mathbf{U}_i^m - {}^t\mathbf{U}\| = \frac{0.001}{\|\boldsymbol{\phi}_i\|}$ and, depending upon the units used, $\|\boldsymbol{\phi}_i\|$ can be very large or very small. So we recommend that you specify DUSIZE when the total displacement vector is zero.

7. Static and implicit dynamic analysis

7.1 Linear static analysis

*ref. KJB
Sections 8.2.1,
8.2.2 and 8.2.3*

- The finite element system equilibrium equations

$$\mathbf{KU} = \mathbf{R}$$

are solved in ADINA using a direct solution scheme or an iterative solution scheme.

- The equation solvers assume that the system stiffness matrix is symmetric.
- The equation solvers assume that the system stiffness matrix is positive definite. This requirement can be summarized as follows: The Rayleigh quotient

$$\rho(\boldsymbol{\phi}) = \frac{\boldsymbol{\phi}^T \mathbf{K} \boldsymbol{\phi}}{\boldsymbol{\phi}^T \boldsymbol{\phi}}$$

must be greater than zero for any displacement vector $\boldsymbol{\phi}$. Since $\rho(\boldsymbol{\phi})$ is equal to twice the strain energy stored in the system (for $\boldsymbol{\phi}^T \boldsymbol{\phi} = 1$), this is equivalent to the requirement that the strain energy stored in the finite element system when subjected to any displacement vector $\boldsymbol{\phi}$ must be greater than zero.

- ▶ Hence, the finite element system must be properly supported, so that the system cannot undergo any rigid body displacements or rotations.
- ▶ It also follows that no part of the total finite element system must represent a mechanism, see ref. KJB, Fig. 8.7, p. 704, for such a case.
- ▶ Nodal point degrees of freedom for which there is no stiffness must have been deleted. A degree of freedom does not

carry any stiffness if all of the elements connected to the nodal point do not carry stiffness into that degree of freedom. In this case the degree of freedom must be deleted using any one of the boundary conditions.

Note that nodal degrees of freedom which are not connected to any elements and are not used as slave nodes in constraint equations are (by default) automatically deleted by the AUI.

- ▶ The elements joining into a nodal point and contributing to the stiffness of the nodal point degrees of freedom must all be defined properly and in a physically correct manner. For example, if there are program input errors that yield a zero or negative Young's modulus or an incorrect nodal point numbering for an element, the stiffness at a system degree of freedom may be zero or negative.

7.1.1 Direct solver; sparse solver

- The main direct solution method in ADINA is a sparse matrix solver. An active column solver based on the Gauss elimination procedure (skyline solver or COLSOL) is also available, but should only be used for small problems.
- The ADINA sparse matrix solver has been proven to yield drastic reductions in solution times, being one to two orders of magnitude faster than the active column solver. In addition, disk and overall memory requirements are reduced.
- The ADINA sparse matrix solver is very reliable and robust and is generally to be used for most problems in ADINA. The ADINA sparse matrix solver is the default solver.
- The sparse solver allocates memory outside that allocated by ADINA. Therefore the total memory used is the sum of the memory reserved by ADINA plus that dynamically allocated by the sparse solver code.

- The sparse solver can be used both in-core and out-of-core. It should be noted that it is more efficient to run an out-of-core sparse solver using real (physical) memory than it is to run an in-core sparse solver using virtual memory. Therefore, if real memory is limited, we recommend the following guidelines:

- 1) The memory allocation MTOT for ADINA should be specified just high enough to allow ADINA to use IOPTIM=3 (refer to the output file for the printout of the IOPTIM value).
- 2) In the AUI, you should provide a memory limit for the sparse solver using the following formula

$$\text{MAXSOLMEM} \leq 0.85 \times (\text{real memory of your computer}) - \text{MTOT}$$

in which MAXSOLMEM, (real memory) and MTOT are measured in MW (or MB).

- When a non-positive definite stiffness matrix (i.e. one with a zero or negative diagonal element) is encountered during solution, ADINA may stop or continue, according to the following rules:

- ▶ If a diagonal element is exactly equal to 0.0, ADINA stops unless
 - The equation number corresponding to the zero diagonal element is attached to an inactive element (element birth/death option is used).
 - You have requested that ADINA continue execution (see below).
- ▶ If the value of a diagonal element is smaller than 10^{-12} but not equal to zero, or the value of a diagonal element is negative, ADINA stops unless one of the following options is used:
 - Automatic load-displacement (LDC)
 - Automatic time-stepping (ATS)
 - Potential-based fluid elements
 - Contact analysis

or unless you have requested that ADINA continue execution.

- When ADINA stops, ADINA prints informational messages for the zero or negative diagonal elements, up to 25% of the number of equations.
- When ADINA continues execution, ADINA assigns a very large number to the diagonal element, effectively attaching a very stiff spring to that degree of freedom.
- You can request that ADINA continue execution using the control parameter “Continue Even When Non-Positive Definite Stiffness Matrix Encountered” in the AUI Control–Solution Process dialog box.
- Note that the stiffness matrix can be non-positive definite due to a modeling error, for example if the model is not sufficiently restrained in static analysis. In this case the results obtained can be misleading.

7.1.2 Iterative solvers; multigrid solver

- In the analysis of large problems, the amount of storage required by a direct solution method may be too large for the available computer resources. For such problems, the use of the iterative method of solution is necessary.
- The iterative solution method used in ADINA is described in ref KJB, Section 8.3.2, and also in:

ref. L.H. Tan and K.J. Bathe, "Studies of Finite Element Procedures – the Conjugate Gradient and GMRES Methods in ADINA and ADINA-F," *J. Computers & Structures*, Vol. 40, pp. 441-449, 1991.

- The iterative solver can be used with all solution options of ADINA, in linear and nonlinear, static and dynamic analysis (except for frequency and linearized buckling analysis). The conditioning of the coefficient matrix is generally significantly better in dynamic analysis because of the effects of the mass matrix (inertia effects) in the coefficient matrix.

- The iterative solver cannot recognize that the stiffness matrix has a zero pivot. If the stiffness matrix is not positive definite, the iteration will continue without reaching convergence.
- The main practical differences between the use of the direct solver and the iterative solver are as follows:
 - ▶ The direct solver executes a predetermined number of operations after which the solution is obtained.
 - ▶ The iterative solver performs a predetermined number of operations, but the number of iterations is not known beforehand. The number of iterations depends on the condition number of the coefficient matrix: the higher the condition number, the more iterations are needed.

The number of iterations required varies from a few hundred to a few thousand. In general, the solution of bulky 3-D solids requires significantly less iteration than the solution of thin structural configurations.

- Regarding the convergence of the iterative method, assume that the system of equations to be solved is $\mathbf{Ax} = \mathbf{b}$, where $\mathbf{x}^{(k)}$ is the approximate solution at iteration k , and the residual vector is $\mathbf{r}^{(k)} = \mathbf{b} - \mathbf{Ax}^{(k)}$. The iterative method is converged when either

$$\left\| \mathbf{r}^{(k)} \right\|_2 \leq \text{EPSIA} \quad \text{and} \quad \frac{\left\| \mathbf{r}^{(k)} \right\|_2}{\left\| \mathbf{b} \right\|_2} \leq \text{EPSIB}$$

or

$$\left\| \mathbf{r}^{(k)} \right\|_2 \leq \text{EPSII}$$

Note that EPSII should be less than or equal to EPSIA. The defaults are EPSIA=1E-6, EPSIB=1E-4, EPSII=1E-8. However, for nonlinear analysis with equilibrium iterations, looser tolerances can be used, e.g. EPSIA=1E-4, EPSIB=1E-3, EPSII=1E-5.

- Difficulties might be encountered when the matrix \mathbf{A} is ill-conditioned. For example, static shell problems are difficult to solve because shell structures are much stiffer in membrane action than in bending action. For such cases, a shift factor SHIFT can be effective, with SHIFT > 1.0, e.g. SHIFT=1.02. SHIFT is used to scale the off-diagonal entries of \mathbf{A} by 1/SIFT in order to make the preconditioner more diagonally dominant.
- ADINA also includes a multigrid solver. The multigrid solver is intended for use with very large systems of equations. The multigrid solver is efficiently used when the model consists of 3-D tetrahedral solid elements that were generated using free-form meshing techniques.

7.2 Nonlinear static analysis

*ref. KJB
Section 8.4*

- In nonlinear static analysis the equilibrium equations to be solved are:

$${}^{t+\Delta t} \mathbf{R} - {}^{t+\Delta t} \mathbf{F} = \mathbf{0}$$

where ${}^{t+\Delta t} \mathbf{R}$ is the vector of externally applied nodal loads at time (load) step $t+\Delta t$, and ${}^{t+\Delta t} \mathbf{F}$ is the force vector equivalent (in the virtual work sense) to the element stresses at time $t+\Delta t$.

- The nonlinearity may come from the material properties, the kinematic assumptions, the use of contact surfaces, or the use of special features such as the element birth/death option.
- The solution to the static equilibrium equations can be obtained in ADINA using
 - ▶ Load/displacement incrementation without iteration
 - ▶ Modified Newton iterations, with or without line searches
 - ▶ The BFGS (Broyden-Fletcher-Goldfarb-Shanno) method

- ▶ Full Newton iterations, with or without line searches
- ▶ Automatic step incrementation (automatic-time-stepping and load-displacement-control methods)

These methods are described in detail in the following sections and also in Sections 6.1 and 8.4 of ref. KJB.

- The same equation solvers are used in nonlinear analysis as in linear analysis. However, the automatic time stepping algorithms do not require the stiffness matrix to be positive definite, thus allowing for the solving of bifurcation problems.
- A special case is reached when the element birth or death option is used for some elements. In this case the equation solver allows that during the solution a degree of freedom may not carry any stiffness at a particular time, because the adjoining element(s) may not have been born yet (or may have died already). Note that the program does not check if any load is applied in that degree of freedom, and that the appropriate time functions should be used to generate/delete that load according to the element birth or death.
- The stiffness stabilization feature can be used to treat some nonlinear static problems involving a non-positive definite stiffness matrix. See section 11.14 for details.

7.2.1 Solution of incremental nonlinear static equations

Load/displacement incrementation without iteration: The following algorithm is used if equilibrium iteration is not performed:

$${}^{\tau}\mathbf{K}\mathbf{U} = {}^{t+\Delta t}\mathbf{R} - {}^t\mathbf{F}$$

$${}^{t+\Delta t}\mathbf{U} = {}^t\mathbf{U} + \mathbf{U}$$

where ${}^{\tau}\mathbf{K}$ is the tangent stiffness matrix at time τ , where $\tau \leq t$; ${}^{t+\Delta t}\mathbf{R}$ is the externally applied load vector at time $t+\Delta t$; ${}^t\mathbf{F}$ is the consistent nodal force vector corresponding to the element stresses due to the displacement vector ${}^t\mathbf{U}$, and \mathbf{U} is the incremental displacement vector.

Note that ${}^{\tau}\mathbf{K}$ is the stiffness matrix corresponding to the last stiffness reformation. Stiffness reformations are performed only at the solution steps that you specify.

Since equilibrium iteration is not performed, the solution in general does not satisfy accurately nodal point equilibrium. Hence, in practice this method should only be used with a sufficiently small time step increment so that divergence from the equilibrium solution is small. Whether nodal point equilibrium is sufficiently satisfied can be approximately checked by performing equilibrium iterations at the final solution step and noting how many iterations are required.

Modified Newton iteration: In modified Newton iteration the following algorithms are used:

- ▶ Without line search

$$\begin{aligned} {}^{\tau}\mathbf{K} \Delta\mathbf{U}^{(i)} &= {}^{t+\Delta t}\mathbf{R} - {}^{t+\Delta t}\mathbf{F}^{(i-1)} \\ {}^{t+\Delta t}\mathbf{U}^{(i)} &= {}^{t+\Delta t}\mathbf{U}^{(i-1)} + \Delta\mathbf{U}^{(i)} \end{aligned} \quad (7.2-1a,b)$$

- ▶ With line search

$$\begin{aligned} {}^{\tau}\mathbf{K} \Delta\mathbf{U}^{(i)} &= {}^{t+\Delta t}\mathbf{R} - {}^{t+\Delta t}\mathbf{F}^{(i-1)} \\ {}^{t+\Delta t}\mathbf{U}^{(i)} &= {}^{t+\Delta t}\mathbf{U}^{(i-1)} + \beta^{(i)} \Delta\mathbf{U}^{(i)} \end{aligned} \quad (7.2-1c,d)$$

where ${}^{\tau}\mathbf{K}$ is the tangent stiffness matrix at time τ , where $\tau \leq t$; ${}^{t+\Delta t}\mathbf{R}$ is the externally applied load vector at time $t + \Delta t$; ${}^{t+\Delta t}\mathbf{F}^{(i-1)}$ is the consistent nodal force vector corresponding to the element stresses due to the displacement vector ${}^{t+\Delta t}\mathbf{U}^{(i-1)}$; $\Delta\mathbf{U}^{(i)}$ is the incremental displacement vector in iteration (i) and $\beta^{(i)}$ is an acceleration factor obtained from line search.

Note that ${}^{\tau}\mathbf{K}$ is the stiffness matrix corresponding to the last stiffness reformation. Stiffness reformations are performed only at the solution steps that you specify.

Modified Newton iterations are performed only at the solution steps that you specify.

*ref. KJB
Section 8.4.2*

BFGS method: In the BFGS matrix update method, the following algorithm is used:

$$\begin{aligned} {}^{t+\Delta t} \mathbf{K}^{*(i-1)} \Delta \mathbf{U}^{(i)} &= {}^{t+\Delta t} \mathbf{R} - {}^{t+\Delta t} \mathbf{F}^{(i-1)} \\ {}^{t+\Delta t} \mathbf{U}^{(i)} &= {}^{t+\Delta t} \mathbf{U}^{(i-1)} + \beta^{(i)} \Delta \mathbf{U}^{(i)} \end{aligned}$$

where ${}^{t+\Delta t} \mathbf{K}^{*(i-1)}$ is an updated stiffness matrix (based on the iteration history). The definitions of the other variables are as for Eq. (7.2-1).

Note that ${}^{t+\Delta t} \mathbf{K}^{*(i-1)}$ is not explicitly formed, but instead the inverse of the stiffness matrix is updated using vector products to provide a secant approximation to the stiffness matrix in successive iterations.

BFGS matrix updates are performed only at the solution steps that you specify.

Full Newton iterations: In the full Newton iteration method, the algorithms employed without and with line searches are:

- ▶ Without line search

$$\begin{aligned} {}^{t+\Delta t} \mathbf{K}^{(i-1)} \Delta \mathbf{U}^{(i)} &= {}^{t+\Delta t} \mathbf{R} - {}^{t+\Delta t} \mathbf{F}^{(i-1)} \\ {}^{t+\Delta t} \mathbf{U}^{(i)} &= {}^{t+\Delta t} \mathbf{U}^{(i-1)} + \Delta \mathbf{U}^{(i)} \end{aligned}$$

- ▶ With line search

$$\begin{aligned} {}^{t+\Delta t} \mathbf{K}^{(i-1)} \Delta \mathbf{U}^{(i)} &= {}^{t+\Delta t} \mathbf{R} - {}^{t+\Delta t} \mathbf{F}^{(i-1)} \\ {}^{t+\Delta t} \mathbf{U}^{(i)} &= {}^{t+\Delta t} \mathbf{U}^{(i-1)} + \beta^{(i)} \Delta \mathbf{U}^{(i)} \end{aligned}$$

where ${}^{t+\Delta t} \mathbf{K}^{(i-1)}$ is the tangent stiffness matrix based on the solution calculated at the end of iteration ($i - 1$) at time $t + \Delta t$.

The definitions of the other variables are as for Eq. (7.2-1). Note that when the full Newton iteration method is employed a

new stiffness matrix is always formed at the beginning of each new load step and in each iteration.

7.2.2 Line search

When the line search feature is activated, the incremental displacements obtained from the solver are modified as follows

$${}^{t+\Delta t}\mathbf{U}^{(i)} = {}^{t+\Delta t}\mathbf{U}^{(i-1)} + \beta^{(i)}\Delta\mathbf{U}^{(i)}$$

where β is a scaling factor obtained from a line search in the direction $\Delta\mathbf{U}^{(i)}$ in order to reduce out-of-balance residuals, according to the following criterion

$$\frac{\Delta\mathbf{U}^{(i)T} \left[{}^{t+\Delta t}\mathbf{R} - {}^{t+\Delta t}\mathbf{F}^{(i)} \right]}{\Delta\mathbf{U}^{(i)T} \left[{}^{t+\Delta t}\mathbf{R} - {}^{t+\Delta t}\mathbf{F}^{(i-1)} \right]} \leq \text{STOL} \quad (7.2.2)$$

where STOL is a user-input line search convergence tolerance, and ${}^{t+\Delta t}\mathbf{F}^{(i)}$ is calculated using the total displacement vector ${}^{t+\Delta t}\mathbf{U}^{(i)}$.

The magnitude of β is also governed by the following bounds

$$\text{LSLOWER} \leq \beta \leq \text{LSUPPER} \quad (7.2.3)$$

where LSUPPER and LSLOWER are user-input parameters.

The incremental displacements are not modified (i.e. $\beta = 1$) if no suitable line search parameter satisfying equations 7.2.2 and 7.2.3 is found within a reasonable number of line search iterations, or if the unbalanced energy falls below a certain user-specified energy threshold ENLSTH.

Line search is off by default except for the BFGS method. It is useful for problems involving plasticity, as well as large displacement problems involving beams and shells. It is also helpful in many contact problems. In the case of contact problems, it is sometimes better to set LSUPPER to 1.0 so that the line search only scales down displacements.

The effect of line search is more prominent when the current displacements are far from the converged solution. This usually happens in the first few iterations of a time step, or when a major change occurs in the model, due for example, to contact initiation/separation, or the onset of plasticity.

Note that line search increases the computational time for each iteration. Most of the extra time goes towards the evaluation of $\mathbf{F}^{(i)}$ in equation 7.x.1. However, for the types of problems mentioned above the reduction in the number of iterations and the ability to use bigger time steps leads to an overall reduction in solution time.

Automatic step incrementation by the Automatic-Time-Stepping (ATS) method

- The automatic-time-stepping (ATS) method is used to obtain a converged solution when the modified Newton (with equilibrium iterations), BFGS or full Newton iteration techniques are used and the user predefined time or load steps are possibly too large.

The iterations are performed as described above and if no convergence is reached, ADINA automatically subdivides the total load step increment so as to reach convergence.

- You can choose the maximum number of subdivisions allowed for each time (or load) step increment (the default is 10). The solution output is only furnished at the discrete user-specified time steps, except when the solution is abandoned because the maximum number of subdivisions is too small. In this case, the solution output is also given for the last subdivision time at which equilibrium has still been established.
- You can choose the division factor that ADINA uses to subdivide the time step (the default is 2).
- Note that the loads are defined as if automatic step incrementation was not used, and the only effect of the ATS method is to continue the iteration solution by the load scaling described above when no convergence is reached.
- When contact surfaces are present in the analysis, ADINA uses

by default the complete time step increment to start the next step calculation once the current solution has been obtained. However, in this case, if a time function point lies between the current time and the time of the next step (using the full time step increment), then the time function point is used as the time for the next step. Hence, no point of the time functions is ever skipped.

When no contact surfaces are present, ADINA uses by default the last subdivided time increment to continue the analysis.

- After the solution is obtained, ADINA chooses the next time step according to one of the following options:
 - ▶ ADINA chooses using the criteria given above (the default).
 - ▶ ADINA uses the last subdivided time step.
 - ▶ ADINA uses the time step prior to subdivision.
 - ▶ ADINA uses the original (user-supplied) time step.
- For dynamic nonlinear analysis, the ATS method can also be used to obtain a more accurate solution. The accuracy checking is performed after iteration convergence in each step, using the accuracy criterion

$$\left| \frac{t+\frac{\Delta t}{2}}{2} U_{im} - \frac{t+\frac{\Delta t}{2}}{2} U_{ih} \right|_{\max} \leq \text{DISTOL}$$

in which i represents each translational degree of freedom,

$$\begin{aligned} \frac{t+\frac{\Delta t}{2}}{2} U_{im} &= \frac{\left({}^tU_i + {}^{t+\Delta t}U_i \right)}{2} \\ \frac{t+\frac{\Delta t}{2}}{2} U_{ih} &= {}^tU_i + \frac{\Delta t}{2} {}^t\dot{U}_i + \frac{\Delta t^2}{16} \left({}^t\ddot{U}_i + {}^{t+\Delta t}\dot{U}_i \right) \end{aligned}$$

and $\frac{t+\frac{\Delta t}{2}}{2} U_{ih}$ is the displacement evaluated at time $t + \frac{\Delta t}{2}$.

- Note that all nonlinear analyses, except those only possible with the load-displacement-control (LDC) method, can be carried out with the ATS method.
- The ATS method includes a low-speed dynamics option (static analysis only). Solving as a low-speed dynamics problem can help to overcome convergence difficulties in collapse, post-collapse and contact problems.

In essence, for low-speed dynamics, ADINA solves

$$\mathbf{M}^{t+\Delta t} \ddot{\mathbf{U}}^{(i)} + \mathbf{C}^{t+\Delta t} \dot{\mathbf{U}}^{(i)} + {}^{t+\Delta t} \mathbf{K}^{(i-1)} \Delta \mathbf{U}^{(i)} = {}^{t+\Delta t} \mathbf{R} - {}^{t+\Delta t} \mathbf{F}^{(i-1)}$$

(compare Equation (7.4-2) below). \mathbf{M} is evaluated as usual (e.g. from the density). \mathbf{C} is evaluated using

$$\mathbf{C} = \beta \mathbf{K}$$

where β is a user-specified parameter (default 10^{-4}) and \mathbf{K} is the (initial) total stiffness matrix (corresponding to zero initial displacements).

When the low-speed dynamics option is used, the time step size will influence the results. It is recommended that the time step size be at least $10^5 \beta$. Or, it is recommended that the loads be held constant for a period of time of at least $10^4 \beta$ so that the dynamic effects die out.

- The ATS method cannot be used in linearized buckling analysis, mode superposition analysis, response spectrum analysis or dynamic analysis with the Wilson theta method.

Automatic step incrementation by the Load-Displacement-Control (LDC) method

- The load-displacement-control (LDC) method can be used to solve for the nonlinear equilibrium path of a model until its collapse. If desired, the post-collapse response of the model can also be calculated. The main feature of the method is that the level of the externally applied loads is adjusted automatically by the

program.

The LDC method can only be used in nonlinear static analysis in which there are no temperature, strain-rate, pipe internal pressure or creep effects.

The LDC method used in ADINA is described in the following reference:

ref. Bathe, K.J. and Dvorkin, E.N., "On the Automatic Solution of Nonlinear Finite Element Equations," *J. Computers and Structures*, Vol. 17, No. 5-6, pp. 871-879, 1983.

- The equations employed in the equilibrium iterations are

$$\begin{aligned} {}^{t+\Delta t} \mathbf{K}^{(i-1)} \Delta \mathbf{U}^{(i)} &= \left({}^{t+\Delta t} \lambda^{(i-1)} + \Delta \lambda^{(i)} \right) \mathbf{R} + \mathbf{R}_p - {}^{t+\Delta t} \mathbf{F}^{(i-1)} \\ {}^{t+\Delta t} \mathbf{U}^{(i)} &= {}^{t+\Delta t} \mathbf{U}^{(i-1)} + \Delta \mathbf{U}^{(i)} \\ f(\Delta \lambda^{(i)}, \Delta \mathbf{U}^{(i)}) &= 0 \end{aligned} \quad (7.2-4)$$

where

${}^{t+\Delta t} \mathbf{K}^{(i-1)}$ = tangent stiffness matrix at time $t+\Delta t$, end of iteration $(i-1)$

\mathbf{R} = constant reference load vector

\mathbf{R}_p = load vector from previous solution run

${}^{t+\Delta t} \lambda^{(i-1)}$ = load scaling factor (used on \mathbf{R}) at the end of iteration $(i-1)$ at time $t+\Delta t$

$\Delta \lambda^{(i)}$ = increment in the load scaling factor in iteration (i)

The quantities ${}^{t+\Delta t} \mathbf{F}^{(i-1)}$ and $\Delta \mathbf{U}^{(i)}$ are as defined for Eq. (7.2-1).

Note that in Eq. (7.2-4), the equation $f=0$ is used to constrain the length of the load step. The constant spherical arc length constraint method is usually used and the constant increment of external work method is used if the arc length method has difficulties to converge.

The reference load vector \mathbf{R} is evaluated from all the mechanical loads (except user-supplied and prescribed displacement loadings).

- The load vector from the previous solution run \mathbf{R}_p is held constant. This feature is useful, for example, to model varying loads in the presence of a constant gravity load. In the first run, enter only the gravity loading. Then, in the restart run, add the varying loads using the LDC method.

The load vector from the previous solution run \mathbf{R}_p is available when you choose, in the previous solution run, to transfer the load vector to a restart run. Otherwise \mathbf{R}_p is set to zero.

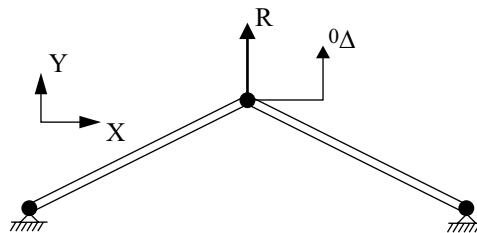
- To start the LDC method, the load multiplier for the first step (${}^{\Delta t} \lambda$), (used to obtain the corresponding load vector ${}^{\Delta t} \lambda \mathbf{R}$) is calculated using a user-specified prescribed displacement. This prescribed displacement (see Fig. 7.2-1) acts in the specified direction at the specified node. The direction of the displacement is given by its sign.

Note that the input for the initial prescribed displacement (in particular whether it is positive or negative) is critical in establishing successive equilibrium positions using the LDC method.

As an example, two entirely different solution paths will be obtained for the same model shown in Fig. 7.2-1 if initial displacements of different signs are prescribed for the first solution step.

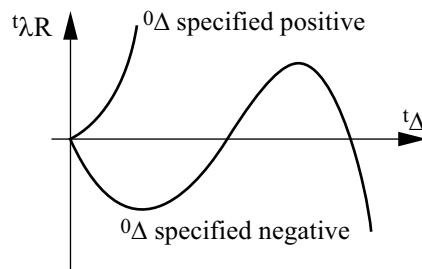
- After the first step, the program automatically traces the nonlinear response by scaling the external load vector \mathbf{R} proportionally, subject to the constraint of Eq. (7.2-4), so that at any discrete time t in iteration (i) , the external load vector is ${}^t \lambda^{(i)} \mathbf{R}$.

The scaling of the reference load vector using ${}^t \lambda$ is analogous to the scaling of the applied loads \mathbf{R} using a user predefined time function when the LDC method is not used (see Chapter 5). In the case of the LDC method, the scaling function is determined internally by the program instead of being user-specified.



Reference load = R , actual load at time $t = {}^t\lambda R$
 Prescribed displacement for first step = ${}^0\Delta$, displacement at time $t = {}^t\Delta$

a) Model considered



b) Equilibrium paths

Figure 7.2-1: Example of the dependence of solution path on the displacement prescribed in the first step for the LDC method

- Note that the converged displacement must satisfy the following relation:

$$\|\mathbf{U}\|_2 \leq 100\alpha \|\Delta \mathbf{U}\|_2$$

where $\mathbf{U} = {}^{t+\Delta t}\mathbf{U} - {}^t\mathbf{U}$ is the incremental displacement vector for the current step, α is an input parameter and $\Delta \mathbf{U}$ is the displacement vector obtained in the first step. If the above inequality is not satisfied, an internal restart of the iteration for the current step is performed by the program.

- The LDC solution terminates normally when any one of the following conditions is satisfied:
 - ▶ The maximum displacement that you specify is reached.
 - ▶ A critical point on the equilibrium path has been passed and you request solution termination.
 - ▶ The number of converged solution steps is reached.
 - ▶ The incremental solution has been attempted a number of times as specified by input (default is ten) from a calculated equilibrium configuration using different strategies but each time the solution has failed to converge within the number of allowed iterations.

7.2.3 Convergence criteria for equilibrium iterations

- The following convergence criteria can be specified in ADINA: **energy only**, **energy and force/moment**, **energy and translation/rotation**, **force/moment only**, **translation/rotation only**.
- If contact surfaces are defined in an analysis, then the contact force convergence criterion is always used in addition to the above criteria (see Chapter 4).

LDC method not used

- If the LDC method is not used, the convergence in equilibrium iterations is reached when the following inequalities are satisfied:

Energy convergence criterion: For all degrees of freedom

$$\frac{\Delta \mathbf{U}^{(i)T} \left[{}_{t+\Delta t}^t \mathbf{R} - {}_{t+\Delta t}^t \mathbf{F}^{(i-1)} \right]}{\Delta \mathbf{U}^{(1)T} \left[{}_{t+\Delta t}^t \mathbf{R} - {}_{t+\Delta t}^t \mathbf{F} \right]} \leq \text{ETOL} \quad (7.2-5)$$

where ETOL is a user-specified energy convergence tolerance.

Force and moment convergence criteria: For the translational degrees of freedom

$$\frac{\left\| {}^{t+\Delta t} \mathbf{R} - {}^{t+\Delta t} \mathbf{F}^{(i-1)} \right\|_2}{\text{RNORM}} \leq \text{RTOL}$$

For the rotational degrees of freedom

$$\frac{\left\| {}^{t+\Delta t} \mathbf{R} - {}^{t+\Delta t} \mathbf{F}^{(i-1)} \right\|_2}{\text{RMNORM}} \leq \text{RTOL}$$

where RTOL, RNORM and RMNORM are user-specified parameters.

Translation/rotation convergence criteria: For the translational degrees of freedom

$$\frac{\left\| \Delta \mathbf{U}^{(i)} \right\|_2}{\text{DNORM}} \leq \text{DTOL}$$

For the rotational degrees of freedom

$$\frac{\left\| \Delta \mathbf{U}^{(i)} \right\|_2}{\text{DMNORM}} \leq \text{DTOL}$$

where DTOL, DNORM and DMNORM are user-specified parameters.

Note that in each of these convergence criteria the residual norm is measured against a user-specified maximum residual value; for example, the force criterion could be interpreted as

$$(\text{norm of out-of-balance loads}) \leq \text{RTOL} \times \text{RNORM}$$

where $\text{RTOL} \times \text{RNORM}$ is equal to the user-specified maximum allowed out-of-balance load.

Note also that these convergence criteria are used in each

subdivision of time or load step when the ATS method of automatic step incrementation is used.

- If contact surface groups are present in the analysis the following additional criterion is always used in measuring convergence

$$\frac{\left\| \mathbf{R}_c^{(i-1)} - \mathbf{R}_c^{(i-2)} \right\|_2}{\max \left(\left\| \mathbf{R}_c^{(i-2)} \right\|_2, RCONSM \right)} \leq RCTOL$$

where $\mathbf{R}_c^{(i-1)}$ is the contact force vector at the end of iteration $(i-1)$, RCONSM is a reference contact force level used to prevent possible division by zero and RCTOL is a user-specified contact force convergence tolerance.

Nonconvergence: If the specified convergence criteria are not satisfied within the allowed number of iterations, but convergence might be reached with more iterations, the following message is issued by ADINA:

ITERATION LIMIT REACHED WITH NO CONVERGENCE
STOP OF SOLUTION

In this case, the analysis can be rerun with a higher limit for the number of allowed iterations or a larger number of maximum allowable subdivisions when the ATS method is used. Alternatively, a smaller time/load step or a more powerful iteration scheme can be used, see the discussions given in Section 7.2.3.

When the energy convergence criterion is employed, divergence in equilibrium iterations is detected if the ratio in Eq. (7.2-3) becomes unacceptably large. The program execution is then terminated with the following message:

OUT-OF-BALANCE LOADS LARGER THAN
INCREMENTAL LOADS AFTER ITERATION = (iteration
number)

In this case, a smaller time/load step or a more powerful iteration scheme should be used.

LDC method used

- Convergence in the equilibrium iterations is reached when the following inequalities are satisfied:

Energy convergence criterion: For all degrees of freedom

$$\frac{\Delta \mathbf{U}^{(i)T} \left[{}^{t+\Delta t} \boldsymbol{\lambda}^{(i-1)} \mathbf{R} - {}^{t+\Delta t} \mathbf{F}^{(i-1)} \right]}{\Delta \mathbf{U}^{(1)T} [\Delta \boldsymbol{\lambda}^{(1)} \mathbf{R}]} \leq \text{ETOL}$$

where ETOL is a user-specified energy convergence tolerance.

Force and moment convergence criteria: For the translational degrees of freedom

$$\frac{\left\| {}^{t+\Delta t} \boldsymbol{\lambda}^{(i-1)} \mathbf{R} - {}^{t+\Delta t} \mathbf{F}^{(i-1)} \right\|_2}{\text{RNORM}} \leq \text{RTOL}$$

For the rotational degrees of freedom

$$\frac{\left\| {}^{t+\Delta t} \boldsymbol{\lambda}^{(i-1)} \mathbf{R} - {}^{t+\Delta t} \mathbf{F}^{(i-1)} \right\|_2}{\text{RMNORM}} \leq \text{RTOL}$$

where RTOL, RNORM and RMNORM are user-specified parameters.

The translation/rotation convergence criteria, and the contact convergence criterion, are the same as when the LDC method is not used, see above.

Nonconvergence: If convergence has not been reached from an established equilibrium configuration with ten restart attempts, the program saves the required restart information and the following message is printed prior to termination of program execution.

TEN ATTEMPTS HAVE BEEN MADE TO ESTABLISH THE
NEXT EQUILIBRIUM CONFIGURATION.

NO CONVERGENCE IS REACHED.

STOP OF SOLUTION.

You can continue the solution by performing a restart run. Note that in this case the LDC method must be used in the restart run. A different value for the initial displacement can be prescribed at a different nodal point in the first step of the restart run. The prescribed initial displacement then corresponds to a displacement increment from the last converged equilibrium position, that is, at the time of solution start for the restart analysis.

7.2.4 Selection of incremental solution method

In practice, the question is frequently which incremental solution method to use for a given analysis. The following recommendations can be given.

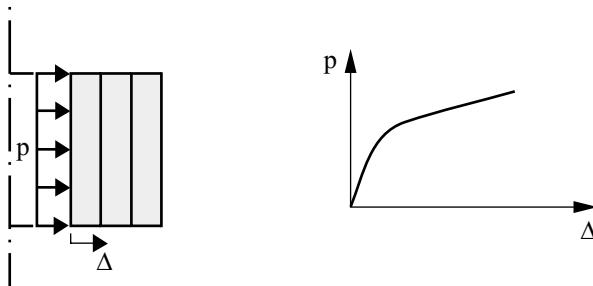
- Every nonlinear analysis should be preceded by a linear analysis, if only to check that the model has been set up correctly. The linear analysis results will tell whether proper boundary conditions are imposed, all degrees of freedom without stiffness have been deleted, and so on (see Section 7.1), and the finite element mesh is adequate.

In practice, it may be best to set up the complete nonlinear model, but only perform a one load step analysis without equilibrium iterations. The displacement results obtained from this solution are linear analysis results and can frequently be used to establish a reasonable load incrementation for the nonlinear solution.

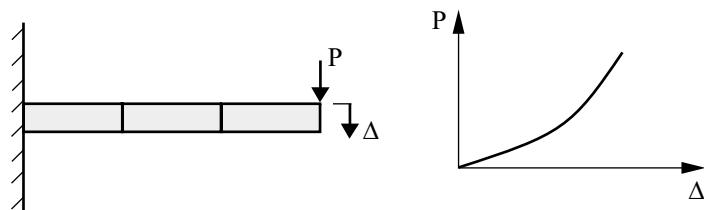
- If the use of a sufficient number of load steps and equilibrium iterations with tight convergence tolerances at each load step is considered to yield an accurate solution of the model, then the basic aim is to obtain a response prediction close to this accurate one at as small a solution cost as possible.
- If you are not at all familiar with the nonlinear response of the structural model considered, it may be advisable to first perform a nonlinear analysis without equilibrium iterations, but using a sufficiently large number of load steps. The calculated response is

then not necessarily very accurate, but at least you now have a good idea of the nonlinear response of the model. The next objective must be to ensure that the analysis results are close enough to the accurate solution of the model. This can be achieved by performing the analysis again with more load increments and by the use of equilibrium iterations.

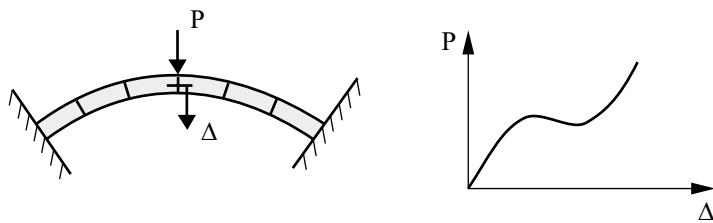
- If the cost of solution without equilibrium iterations seems reasonable and reasonable nonlinearities per step were observed, it is usually best to perform the next analysis with equilibrium iterations using the default iteration technique, the full Newton method, and default tolerances of the program.
- The convergence of the equilibrium iterations is affected by the kind of nonlinear response of the model. If the structural model softens under increasing loads, the iteration will generally converge more easily than when the model stiffens. A structural model can, for example, soften due to increasing spread of plasticity whereas the model can stiffen or soften due to geometric nonlinear effects (see Fig. 7.2-2).
- The use of the full Newton iteration method and a reasonable load incrementation is frequently sufficient to ensure that an accurate solution of the response of the model will be obtained. However, considerations for the use of the other solution techniques available in ADINA are given below.
 - ▶ The modified Newton iteration is less expensive (per iteration) than the BFGS method, which is, in turn, less expensive per iteration than the full Newton iteration. However, the more expensive method (per iteration) may require less iterations to achieve convergence. Line searches help in achieving convergence but also add to the cost per iteration.



- (a) Softening problem. Materially-nonlinear-only analysis,
elasto-plastic analysis of a cylinder



- (b) Stiffening problem. Large displacement nonlinear elastic
analysis of a cantilever



- (c) Softening/stiffening problem. Large displacement analysis of a
thin arch

Figure 7.2-2: Different types of nonlinear analyses

- ▶ Of the methods available with line searches, modified Newton iteration should be used for mild nonlinearities; the BFGS method is effective with somewhat larger nonlinearities, and full Newton iteration is most powerful. Hence, when convergence difficulties are encountered full Newton iteration with line searches should be used.
- ▶ The ATS method can be used with the modified Newton iteration, the BFGS method and the full Newton method. If a reasonable number of time (or load) steps is used for the nonlinear response, then the use of the ATS method will result in almost the same "iteration path" as when not using the method. Namely, no step subdividing will be performed if convergence is always directly reached at the user-specified load levels.

Hence, in general, if you want to use the most convenient method, it is recommended to use a reasonable number of load steps with the default iteration scheme (full Newton method), with default tolerances and the ATS method. However, a less expensive solution can frequently be reached by use of the other available iteration methods.
- ▶ The LDC method is very useful if collapse of the structure occurs during the (static) solution. However, the use of this algorithm can be very costly. The LDC method is recommended if you do not want to specify the load increments for the solution period and computational costs are not of primary concern. The LDC method allows also the calculation of the post-collapse response. Note, however, that the displacement solution at a specific load level cannot be obtained using the LDC method because the load increments are calculated by the program.
- Note that usually it is quite adequate to employ the energy convergence tolerance only.

The need to use the force/moment criteria and/or translation/rotation criteria arises when the energy convergence is not tight (small) enough. In addition, there exist special loading conditions under which the denominator of the inequality (7.2-4) in Section 7.2.3 becomes small and hence the inequality is difficult to satisfy.

7.2.5 Example

		OUT-OF-		NORM OF					
		BALANCE	ENERGY	FORCE	MOMENT	NORM OF	INCREMENTAL		
		(EQ MAX)	(EQ MAX)	(EQ MAX)	(EQ MAX)	DISP.	ROTN.	CFORCE	...
		VALUE	VALUE	VALUE	VALUE	CFNORM	...		
ITE= 0	0.35E+03	0.75E+04	0.26E-12	0.61E-01	0.11E-03	0.00E+00	...		
		(-40)	(-29)	(-54)	(-45)	0.00E+00	...		
		-0.45E+04	0.19E-12	-0.24E-01	-0.49E-04				
ITE= 1	0.28E-03	0.58E+01	0.10E+01	0.13E-02	0.87E-04	0.00E+00	...		
		(-23)	(-22)	(-1)	(-45)	0.00E+00	...		
		-0.34E+01	0.50E+00	0.64E-03	-0.39E-04				
ITE= 2	0.60E-04	0.79E-01	0.84E+00	0.10E-02	0.70E-04	0.00E+00	...		
		(-15)	(-22)	(-1)	(-45)	0.00E+00	...		
		0.34E-01	0.40E+00	0.52E-03	-0.31E-04				
ITE= 3	0.27E-05	0.19E+00	0.18E+00	0.22E-03	0.15E-04	0.00E+00	...		
		(-47)	(-22)	(-1)	(-45)	0.00E+00	...		
		-0.12E+00	0.85E-01	0.11E-03	-0.66E-05				
ITE= 4	0.26E-06	0.47E+00	0.33E-01	0.47E-04	0.32E-05	0.00E+00	...		
		(-47)	(-22)	(-1)	(-45)	0.00E+00	...		
		0.30E+00	0.16E-01	0.23E-04	-0.14E-05				
ITE= 5	0.77E-07	0.67E-02	0.31E-01	0.37E-04	0.25E-05	0.00E+00	...		
		(-47)	(-22)	(-1)	(-45)	0.00E+00	...		
		-0.41E-02	0.15E-01	0.19E-04	-0.11E-05				
...	CONVERGENCE RATIOS	CONVERGENCE RATIOS		OUT-OF-BALANCE LOAD					
...	FOR OUT-OF-BALANCE	FOR INCREMENTAL		VECTOR CALCULATION					
...	ENERGY	FORCE	DISP.	CFORCE	BETA	RATIO			
...		MOMENT	ROTN.				(ITERNS)		
...	COMPARE WITH	COMPARE WITH							
...	ETOL	RTOL	DTOL	RCTOL					
...			(NOT USED)	(NOT USED)					
...	0.10E+01	0.75E+03	0.00E+00	0.00E+00					
...		0.26E-13	0.00E+00						
...	0.78E-06	0.58E+00	0.00E+00	0.00E+00	0.10E+01	0.71E-03			
...		0.10E+00	0.00E+00			(1)			
...	0.17E-06	0.79E-02	0.00E+00	0.00E+00	0.10E+01	0.27E+00			
...		0.84E-01	0.00E+00			(1)			
...	0.76E-08	0.19E-01	0.00E+00	0.00E+00	0.40E+01	0.21E+00			
...		0.18E-01	0.00E+00			(3)			
...	0.72E-09	0.47E-01	0.00E+00	0.00E+00	0.40E+01	0.19E+00			
...		0.33E-02	0.00E+00			(3)			
...	0.22E-09	0.67E-03	0.00E+00	0.00E+00	0.10E+01	0.37E+00			
...		0.31E-02	0.00E+00			(1)			

Figure 7.2-3: Example of iteration history printout

We now present a worked example that illustrates the concepts previously discussed. Figure 7.2-3 shows the iteration history for a load step. The modified Newton method with line searches is used (STOL=0.5), with energy and force convergence criteria (ETOL=1.E -8, RTOL=1.E-2, RNORM=1.E+1, RMNORM=1.E+1).

Row ITE=0: This row shows the result of the initial iteration. For ITE=0, ADINA performs the following steps:

Computes ${}^{t+\Delta t} \mathbf{U}^{(0)} = {}^t \mathbf{U}$.

Computes ${}^{t+\Delta t} \mathbf{F}^{(0)}$ and ${}^{t+\Delta t} \mathbf{K}^{(0)}$ using ${}^{t+\Delta t} \mathbf{U}^{(0)}$.

Computes the out-of-balance force vector ${}^{t+\Delta t} \mathbf{R} - {}^{t+\Delta t} \mathbf{F}^{(0)}$. Only considering translational degrees of freedom, the norm of the out-of-balance force vector is $\| {}^{t+\Delta t} \mathbf{R} - {}^{t+\Delta t} \mathbf{F}^{(0)} \|_2 = 0.75E + 04$

and the largest magnitude in the out-of-balance force vector is $-0.45E + 04$ at equation number 40. Only considering rotational degrees of freedom, the norm of the out-of-balance force vector is $\| {}^{t+\Delta t} \mathbf{R} - {}^{t+\Delta t} \mathbf{F}^{(0)} \|_2 = 0.26E - 12$ and the largest magnitude in the out-of-balance force vector is $0.19E - 12$ at equation number 29.

Computes $\Delta \mathbf{U}^{(1)}$ using ${}^{t+\Delta t} \mathbf{K}^{(0)} \Delta \mathbf{U}^{(1)} = {}^{t+\Delta t} \mathbf{R} - {}^{t+\Delta t} \mathbf{F}^{(0)}$. Only considering translational degrees of freedom, the norm of the incremental displacement vector is $\| \Delta \mathbf{U}^{(1)} \|_2 = 0.61E - 01$ and the largest magnitude in the incremental displacement vector is $-0.24E - 01$ at equation number 54. Only considering rotational degrees of freedom, the norm of the incremental displacement vector is $\| \Delta \mathbf{U}^{(1)} \|_2 = 0.11E - 03$ and the largest magnitude in the incremental displacement vector is $-0.49E - 04$ at equation number 45.

Computes the “out-of-balance energy”

$$\Delta \mathbf{U}^{(1)T} \left({}^{t+\Delta t} \mathbf{R} - {}^{t+\Delta t} \mathbf{F}^{(0)} \right) = 0.35E + 03.$$

Computes the energy convergence criterion

$$\frac{\Delta \mathbf{U}^{(1)T} \left[{}^{t+\Delta t} \mathbf{R} - {}^{t+\Delta t} \mathbf{F}^{(0)} \right]}{\Delta \mathbf{U}^{(1)T} \left[{}^{t+\Delta t} \mathbf{R} - {}^t \mathbf{F} \right]} = 0.10E + 01$$

Computes the force and moment convergence criteria

$$\frac{\left\| {}^{t+\Delta t} \mathbf{R} - {}^{t+\Delta t} \mathbf{F}^{(0)} \right\|_2}{\text{RNORM}} = 0.75E + 03 \text{ and}$$

$$\frac{\left\| {}^{t+\Delta t} \mathbf{R} - {}^{t+\Delta t} \mathbf{F}^{(0)} \right\|_2}{\text{RMNORM}} = 0.26E - 13.$$

Since the energy convergence criterion is greater than ETOL=1.E-8 and the force convergence criterion is greater than RTOL=1.E-2, convergence is not satisfied.

Row ITE=1: This row shows the results of the first equilibrium iteration. For row ITE=1, ADINA performs the following steps:

Computes ${}^{t+\Delta t} \mathbf{U}^{(1)} = {}^{t+\Delta t} \mathbf{U}^{(0)} + \Delta \mathbf{U}^{(1)}$, ${}^{t+\Delta t} \mathbf{F}^{(1)}$ and the line

$$\text{search ratio } \frac{\Delta \mathbf{U}^{(1)T} \left[{}^{t+\Delta t} \mathbf{R} - {}^{t+\Delta t} \mathbf{F}^{(1)} \right]}{\Delta \mathbf{U}^{(1)T} \left[{}^{t+\Delta t} \mathbf{R} - {}^{t+\Delta t} \mathbf{F}^{(0)} \right]} = 0.71E - 03. \text{ This ratio}$$

is less than STOL=0.5, so ADINA sets the line search factor $\beta^{(1)} = 1$.

Computes ${}^{t+\Delta t} \mathbf{U}^{(1)} = {}^{t+\Delta t} \mathbf{U}^{(0)} + \beta^{(1)} \Delta \mathbf{U}^{(1)}$.

Computes ${}^{t+\Delta t} \mathbf{F}^{(1)}$ using ${}^{t+\Delta t} \mathbf{U}^{(1)}$. Since the modified Newton method is used, ADINA does not compute ${}^{t+\Delta t} \mathbf{K}^{(1)}$.

Computes the out-of-balance force vector ${}^{t+\Delta t} \mathbf{R} - {}^{t+\Delta t} \mathbf{F}^{(1)}$. Only considering translational degrees of freedom, the norm of the out-of-balance force vector is $\left\| {}^{t+\Delta t} \mathbf{R} - {}^{t+\Delta t} \mathbf{F}^{(1)} \right\|_2 = 0.58E + 01$ and the largest magnitude in the out-of-balance force vector is –

0.34E+01 at equation number 23. Only considering rotational degrees of freedom, the norm of the out of balance force vector is $\left\| {}^{t+\Delta t} \mathbf{R} - {}^{t+\Delta t} \mathbf{F}^{(1)} \right\|_2 = 0.10E + 01$ and the largest magnitude in the out-of-balance force vector is 0.50E-00 at equation number 22.

Computes $\Delta \mathbf{U}^{(2)}$ using ${}^{t+\Delta t} \mathbf{K}^{(0)} \Delta \mathbf{U}^{(2)} = {}^{t+\Delta t} \mathbf{R} - {}^{t+\Delta t} \mathbf{F}^{(1)}$. Only considering translational degrees of freedom, the norm of the incremental displacement vector is $\left\| \Delta \mathbf{U}^{(2)} \right\|_2 = 0.13E - 02$ and the largest magnitude in the incremental displacement vector is 0.64E-03 at equation number 1. Only considering rotational degrees of freedom, the norm of the incremental displacement vector is $\left\| \Delta \mathbf{U}^{(2)} \right\|_2 = 0.87E - 04$ and the largest magnitude in the incremental displacement vector is -0.39E-04 at equation number 45.

Computes the “out-of-balance energy”

$$\Delta \mathbf{U}^{(2)T} \left({}^{t+\Delta t} \mathbf{R} - {}^{t+\Delta t} \mathbf{F}^{(1)} \right) = 0.28E - 03.$$

Computes the energy convergence criterion

$$\frac{\Delta \mathbf{U}^{(2)T} \left[{}^{t+\Delta t} \mathbf{R} - {}^{t+\Delta t} \mathbf{F}^{(1)} \right]}{\Delta \mathbf{U}^{(1)T} \left[{}^{t+\Delta t} \mathbf{R} - {}^t \mathbf{F} \right]} = 0.78E - 06.$$

Computes the force and moment convergence criteria

$$\frac{\left\| {}^{t+\Delta t} \mathbf{R} - {}^{t+\Delta t} \mathbf{F}^{(1)} \right\|_2}{\text{RNORM}} = 0.58E + 00 \text{ and}$$
$$\frac{\left\| {}^{t+\Delta t} \mathbf{R} - {}^{t+\Delta t} \mathbf{F}^{(1)} \right\|_2}{\text{RMNORM}} = 0.10E + 00.$$

Since the energy convergence criterion is greater than ETOL=1.E-8, the force convergence criterion is greater than RTOL=1.E-2 and the moment convergence criterion is greater than RTOL=1.E-2, convergence is not satisfied.

Row ITE=2: This row shows the results from the second equilibrium iteration. This row is interpreted exactly as is row ITE=1. Again, convergence is not satisfied.

Row ITE=3: This row shows the results from the third equilibrium iteration. This row is interpreted exactly as is row ITE=2, with one exception. In row ITE=3, the line search proceeds as follows:

Computes $t+\Delta t \mathbf{U}^{(3)} = t+\Delta t \mathbf{U}^{(2)} + \Delta \mathbf{U}^{(3)}, t+\Delta t \mathbf{F}^{(3)}$ and the line

search ratio $\frac{\Delta \mathbf{U}^{(3)T} [t+\Delta t \mathbf{R} - t+\Delta t \mathbf{F}^{(3)}]}{\Delta \mathbf{U}^{(3)T} [t+\Delta t \mathbf{R} - t+\Delta t \mathbf{F}^{(2)}]}$. This ratio is greater

than STOL=0.5, so ADINA computes the line search factor $\beta^{(3)} = 0.40E + 01$ in three line search iterations. At the end of the third line search iteration, $t+\Delta t \mathbf{U}^{(3)} = t+\Delta t \mathbf{U}^{(2)} + 4\Delta \mathbf{U}^{(3)}$, $t+\Delta t \mathbf{F}^{(3)}$ is computed from $t+\Delta t \mathbf{U}^{(3)}$ and the line search ratio

$\frac{\Delta \mathbf{U}^{(3)T} [t+\Delta t \mathbf{R} - t+\Delta t \mathbf{F}^{(3)}]}{\Delta \mathbf{U}^{(3)T} [t+\Delta t \mathbf{R} - t+\Delta t \mathbf{F}^{(2)}]} = 0.21E + 00$, which is less than

STOL.

Again, convergence is not satisfied.

Rows ITE=4, 5: These rows show the results from the fourth and fifth equilibrium iterations. At the end of the fifth equilibrium iteration, the energy convergence criterion is

$\frac{\Delta \mathbf{U}^{(6)T} [t+\Delta t \mathbf{R} - t+\Delta t \mathbf{F}^{(5)}]}{\Delta \mathbf{U}^{(1)T} [t+\Delta t \mathbf{R} - t \mathbf{F}]} = 0.22E-09 < ETOL$, and the force and

moment convergence criteria are

$\frac{\|t+\Delta t \mathbf{R} - t+\Delta t \mathbf{F}^{(5)}\|_2}{RNORM} = 0.67E-03 < RTOL$ and

$\frac{\|t+\Delta t \mathbf{R} - t+\Delta t \mathbf{F}^{(5)}\|_2}{RMNORM} = 0.31E - 02 < RTOL$.

Convergence is satisfied. Five equilibrium iterations were used to

obtain the converged solution.

Comments: If full Newton iterations are used, the iteration history printout is interpreted in exactly the same way. The only difference is that $^{t+\Delta t} \mathbf{K}^{(i)}$ is computed for each equilibrium iteration, and the displacement increment is computed using
 $^{t+\Delta t} \mathbf{K}^{(i-1)} \Delta \mathbf{U}^{(i)} = ^{t+\Delta t} \mathbf{R} - ^{t+\Delta t} \mathbf{F}^{(i-1)}$.

7.3 Linear dynamic analysis

- The following procedures are available in ADINA for solution of the finite element equations in a linear dynamic analysis:

Step-by-step direct integration

- ▶ Explicit time integration using the central difference method (described in Chapter 8)
- ▶ Implicit time integration using the Newmark method or the Wilson theta method

Time history mode superposition analysis

- ▶ Implicit time integration using the Newmark method (trapezoidal rule)

In addition, the eigenproblem for the vibration of the structural system can be solved for the natural frequencies and mode shapes, see Chapter 6.

- The notation given below is used in the following sections in the descriptions of the equilibrium equations:

- \mathbf{M} = constant mass matrix
 \mathbf{C} = constant damping matrix
 \mathbf{K} = constant stiffness matrix
 $^t \mathbf{R}, ^{t+\Delta t} \mathbf{R}$ = external load vector applied at time $t, t+\Delta t$
 $' \mathbf{F}$ = nodal point force vector equivalent to the element stresses at time t
-

- ${}^t\dot{\mathbf{U}}, {}^{t+\Delta t}\dot{\mathbf{U}}$ = vectors of nodal point velocities at time $t, t+\Delta t$
- ${}^t\ddot{\mathbf{U}}, {}^{t+\Delta t}\ddot{\mathbf{U}}$ = vectors of nodal point accelerations at time $t, t+\Delta t$
- \mathbf{U} = vector of nodal point displacement increments from time t to time $t+\Delta t$, i.e., $\mathbf{U} = {}^{t+\Delta t}\mathbf{U} - {}^t\mathbf{U}$.

7.3.1 Step-by-step implicit time integration

- The governing equilibrium equation at time $t+\Delta t$ is given by

$$\mathbf{M} {}^{t+\Delta t}\ddot{\mathbf{U}} + \mathbf{C} {}^{t+\Delta t}\dot{\mathbf{U}} + \mathbf{K} {}^{t+\Delta t}\mathbf{U} = {}^{t+\Delta t}\mathbf{R}$$

The procedures used in the time integration of the governing equations for both the Newmark method and the Wilson theta method are summarized in ref. KJB., Sections 9.2.3 and 9.2.4.

ref. KJB
Sections 9.2.4
and 9.4.4

- Note that the trapezoidal rule (also called the constant-average-acceleration method of Newmark) obtained by using $\delta = 0.5, \alpha = 0.25$ in the Newmark method is recommended for linear dynamic analysis.
- The trapezoidal rule has the following characteristics:
 - ▶ It is an implicit integration method, meaning that equilibrium of the system is considered at time $t+\Delta t$ to obtain the solution at time $t+\Delta t$.
 - ▶ Whether the mass and damping matrices are diagonal or banded (lumped or consistent discretization), the solution always requires that a coefficient matrix be assembled and factorized.
 - ▶ The trapezoidal rule is unconditionally stable. Hence, the time step size Δt is selected based on accuracy considerations only, see ref. KJB, Section 9.4.4.
 - The use of the trapezoidal rule can be effective in wave propagation problems, but only if the finite element system has a narrow bandwidth. In this case the use of higher-order elements with consistent mass idealization can be a good choice.

- The trapezoidal rule is usually more effective for structural vibration problems. In these analyses, the use of higher-order elements, just as in static analysis, and the use of a consistent mass discretization can be effective.
- The step-by-step direct integration solution of a vibration problem is frequently more effective than a mode superposition analysis when the response need only be calculated over a few time steps, say less than 100 steps.
- The time step increment (Δt) recommended for dynamic analysis with the trapezoidal rule is given by $\omega_{co}\Delta t \leq 0.20$ where ω_{co} is the highest frequency of interest in the dynamic response.
- In FSI analysis with ADINA-FSI, the time integration is controlled by the ADINA-F model's α parameter, and the input for the ADINA model's Newmark constants is ignored.

7.3.2 Time history by mode superposition

ref. KJB
Section 9.3

- In mode superposition analysis, the governing finite element equations are solved using the transformation

$$\mathbf{U} = \sum_{i=r}^s \phi_i x_i$$

where the $\phi_i, i = r, \dots, s$ are the mode shapes calculated in a frequency solution, and the x_i are the corresponding unknown generalized displacements. The x_i are calculated by solving the decoupled modal equations

$$\ddot{x}_i + 2\xi_i\omega_i\dot{x}_i + \omega_i^2 x_i = r_i$$

where ξ_i is the critical damping ratio corresponding to the frequency ω_i , and $r_i = \phi_i^T \mathbf{R}$. The Newmark method (trapezoidal rule) is used for the time integration.

- In ADINA, either the mode shapes corresponding to the

smallest frequencies or the frequencies in a band can be used (see Chapter 6).

- Using ADINA, either no damping is specified, or modal damping is specified, in which case the values of $\xi_i, i = r, \dots, s$ can be all different.
- Mode superposition is effective when the time integration has to be carried out over many time steps (e.g., earthquake loading), and the cost of calculating the required frequencies and mode shapes is reasonable.

7.3.3 Lumped and consistent damping; Rayleigh damping

*ref. KJB
Section 9.3.3*

- For a lumped damping matrix, contributions come from concentrated nodal dampers and/or lumped general element dampers and/or lumped spring element dampers. Direct time integration must be employed.
- For a consistent damping matrix, contributions come from Rayleigh damping and/or consistent general element damping and/or consistent spring element damping and/or concentrated nodal dampers. Direct time integration must be employed. Explicit time integration cannot be used. Substructuring cannot be used.
- If Rayleigh damping is specified, the contributions of the following matrix ($\mathbf{C}_{Rayleigh}$) are added to the total system damping matrix \mathbf{C} described in Section 7.3:

$$\mathbf{C}_{Rayleigh} = \alpha\mathbf{M} + \beta\mathbf{K}$$

where \mathbf{M} is the total system mass matrix which can be lumped or consistent, and \mathbf{K} is the (initial) total system stiffness matrix (corresponding to zero initial displacements).

- Different Rayleigh damping factors can be input for different element groups.

- Note that $\mathbf{C}_{Rayleigh}$ is in general a consistent damping matrix ($\mathbf{C}_{Rayleigh}$ is diagonal if β is zero and a lumped mass matrix is used).
- Since $\mathbf{C}_{Rayleigh}$ is constant throughout the solution, it is formed only once in ADINA – before the step-by-step solution of the equilibrium equations.
- See Ref. KJB, Section 9.3.3, for information about selecting the Rayleigh damping constants α, β . In the modal basis, the damping ratio can be written as

$$\xi_i = \frac{\alpha}{2\omega_i} + \frac{\beta\omega_i}{2}$$

where ξ_i is the damping ratio for mode ω_i . It is seen that α tends to damp lower modes and β tends to damp higher modes.

If α is not used, it is seen that a value of $\beta = \frac{T_p}{\pi}$ will overdamp

all motions with periods smaller than T_p . Hence motions with

periods smaller than T_p can be suppressed by choosing $\beta = \frac{T_p}{\pi}$.

This may be of interest when using damping to suppress numerical oscillations.

The above comments apply only when the stiffness matrix does not change significantly during the analysis, however.

- If the potential-based fluid elements are used in the model, the matrices \mathbf{M} and \mathbf{K} also include the fluid elements contributions (see Section 2.11.2). However, the fluid element contributions to \mathbf{M} and \mathbf{K} are not included in $\mathbf{C}_{Rayleigh}$.

7.4 Nonlinear dynamic analysis

- In nonlinear dynamic analysis the solution of the finite element

equations is usually obtained by direct integration procedures. The following major techniques can be employed with ADINA:

Explicit integration (described in Chapter 8)

Implicit integration (using the Newmark method, Wilson theta method or the ADINA composite method)

Some considerations on the choice of the time integration method are given in Section 7.4.3.

- The ADINA composite method is a new composite time integration scheme for nonlinear analysis. In this scheme, the displacements, velocities, accelerations are solved for at a time $t + \gamma\Delta t$, where $\gamma \in (0,1)$ (the default value is 0.5), using the standard Newmark method. Then the ADINA composite scheme is used to solve for the displacements, velocities, accelerations at time $t + \Delta t$.

For large deformation problems, the Newmark method can become unstable. The ADINA composite scheme remains stable for these problems, so the ADINA composite scheme is preferred for these problems. However, for a given time step size, the scheme is about twice as expensive computationally as the Newmark method.

- For certain types of problems it can be effective to use mode superposition for the solution of the nonlinear response. In this case, the finite element equilibrium equations are established in the selected mode shapes and solved by using the Newmark method (see Section 7.4.2).
- It can also be effective to use substructuring for the solution of the equations (see Section 10.1).
- The use of Rayleigh damping is the same as described in Section 7.3.3. Note that the Rayleigh damping matrix remains constant throughout the time integration.

7.4.1 Step-by-step implicit time integration

- In nonlinear analysis the incremental finite element equilibrium equations used are, in implicit time integration (without

equilibrium iterations),

$$\mathbf{M}^{t+\Delta t} \ddot{\mathbf{U}} + \mathbf{C}^{t+\Delta t} \dot{\mathbf{U}} + {}^t \mathbf{K} \mathbf{U} = {}^{t+\Delta t} \mathbf{R} - {}^t \mathbf{F} \quad (7.4-1)$$

- If equilibrium iterations are performed, the governing equations are

$$\mathbf{M}^{t+\Delta t} \ddot{\mathbf{U}}^{(i)} + \mathbf{C}^{t+\Delta t} \dot{\mathbf{U}}^{(i)} + {}^{t+\Delta t} \mathbf{K} \Delta \mathbf{U}^{(i)} = {}^{t+\Delta t} \mathbf{R} - {}^{t+\Delta t} \mathbf{F}^{(i-1)} \quad (7.4-2)$$

where ${}^{t+\Delta t} \ddot{\mathbf{U}}^{(i)}$, ${}^{t+\Delta t} \dot{\mathbf{U}}^{(i)}$, ${}^{t+\Delta t} \mathbf{U}^{(i-1)} + \Delta \mathbf{U}^{(i)}$ are the approximations to the accelerations, velocities, and displacements obtained in iteration (i) respectively.

The vector of nodal point forces ${}^{t+\Delta t} \mathbf{F}^{(i-1)}$ is equivalent to the element stresses in the configuration corresponding to the displacements ${}^{t+\Delta t} \mathbf{U}^{(i-1)}$.

- Note that the trapezoidal rule obtained by using $\delta = 0.5$ and $\alpha = 0.25$ in the Newmark method is recommended for the analysis.
- The dynamic equilibrium equations are solved by use of the iterative procedures also used in static analysis, including the ATS method. However, the LDC method cannot be used (see Section 7.2.1).
- The energy and force/moment convergence criteria used in nonlinear dynamic analysis are:

Energy convergence criterion: For all degrees of freedom

$$\frac{\Delta \mathbf{U}^{(i)T} \left[{}^{t+\Delta t} \mathbf{R} - \mathbf{M}^{t+\Delta t} \ddot{\mathbf{U}}^{(i-1)} - \mathbf{C}^{t+\Delta t} \dot{\mathbf{U}}^{(i-1)} - {}^{t+\Delta t} \mathbf{F}^{(i-1)} \right]}{\Delta \mathbf{U}^{(1)T} \left[{}^{t+\Delta t} \mathbf{R} - \mathbf{M}^{t+\Delta t} \ddot{\mathbf{U}}^{(0)} - \mathbf{C}^{t+\Delta t} \dot{\mathbf{U}}^{(0)} - {}^t \mathbf{F} \right]} \leq \text{ETOL}$$

Force and moment convergence criteria: For the translational degrees of freedom

$$\frac{\left\| \mathbf{R}^{t+\Delta t} - \mathbf{M}^{t+\Delta t} \ddot{\mathbf{U}}^{(i-1)} - \mathbf{C}^{t+\Delta t} \dot{\mathbf{U}}^{(i-1)} - \mathbf{F}^{(i-1)} \right\|_2}{\text{RNORM}} \leq \text{RTOL}$$

For the rotational degrees of freedom

$$\frac{\left\| \mathbf{R}^{t+\Delta t} - \mathbf{M}^{t+\Delta t} \ddot{\mathbf{U}}^{(i-1)} - \mathbf{C}^{t+\Delta t} \dot{\mathbf{U}}^{(i-1)} - \mathbf{F}^{(i-1)} \right\|_2}{\text{RMNORM}} \leq \text{RTOL}$$

The other convergence criteria and the notation and considerations for the use of the convergence criteria are the same as in nonlinear static analysis; see Section 7.2.2.

7.4.2 Mode superposition

*ref. KJB
Section 9.5.3*

- A nonlinear dynamic problem is solved effectively by the mode superposition procedure if the response can be adequately represented by the superposition of the response in a few selected mode shapes.
- The equations solved in ADINA to obtain the solution of the nonlinear dynamic equations by mode superposition are presented on pp. 828-829 of ref. KJB. In ADINA, throughout the time integration, the eigenmodes corresponding to the time at solution start are employed, and the modal equations are integrated using the Newmark method.
- If desired, a restart analysis can be performed continuing the mode superposition analysis. In this case, ADINA will recalculate the eigenmodes for the configuration at which the restart analysis begins, and use these eigenmodes for the mode superposition analysis.
- Some applications using the mode superposition method are given in the following reference:

ref. K.J. Bathe and S. Gracewski, "On Nonlinear Dynamic Analysis Using Substructuring and Mode Superposition," *J. Computers and Structures*, Vol. 13, pp. 699-707, 1981.

7.5 Choosing between implicit and explicit formulations

*ref. KJB
Section 9.2*

- The main criterion governing the selection of the implicit or explicit formulations is the time scale of the solution.
- The implicit method can use much larger time steps since it is unconditionally stable. However, it involves the assembly and solution of a system of equations, and it is iterative. Therefore, the computational time per load step is relatively high. The explicit method uses much smaller time steps since it is conditionally stable, meaning that the time step for the solution has to be less than a certain critical time step, which depends on the smallest element size and the material properties. However, it involves no matrix solution and is non-iterative. Therefore, the computational time per load step is relatively low.
- For both linear and nonlinear static problems, the implicit method is the only option.
- For slow-speed dynamic problems, the time solution time spans a period of time considerably longer than the time it takes the wave to propagate through an element. The solution in this case is dominated by the lower frequencies of the structure. This class of problems covers most structural dynamics problems, certain metal forming problems, crush analysis, earthquake response and biomedical problems. When the explicit method is used for such problems the resulting number of time steps will be excessive, unless mass-scaling is applied, or the loads are artificially applied over a shorter time frame. No such modifications are needed in the implicit method. Hence, the implicit method is the optimal choice.
- For high-speed dynamic problems, the solution time is comparable to the time required for the wave to propagate through the structure. This class of problems covers most wave propagation problems, explosives problems, and high-speed impact problems. For these problems, the number of steps required with the explicit method is not excessive. If the implicit method uses a similar time step it will be much slower and if it uses a much larger time step it will introduce other solution errors since it will not be capturing the

pertinent features of the solution (but it will remain stable). Hence, the explicit method is the optimal choice.

- A large number of dynamics problems cannot be fully classified as either slow-speed or high-speed dynamic. This includes many crash problems, drop tests and metal forming problems. For these problems both solution methods are comparable. However, whenever possible (when the time step is relatively large and there are no convergence difficulties) we recommend the use of the implicit solution method.
- Note that the explicit solution provided in ADINA does not use reduced integration with hour-glassing. This technique reduces the computational time per load step. However, it can have detrimental effect on the accuracy and reliability of the solution.
- Since the explicit time step size depends on the length of the smallest element, one excessively small element will reduce the stable time step for the whole model. Mass-scaling can be applied to these small elements to increase their stable time step. The implicit method is not sensitive to such small elements.
- Since the explicit time step size depends on the material properties, a nearly incompressible material will also significantly reduce the stable time step. The compressibility of the material can be increased in explicit analysis to achieve a more acceptable solution time. The implicit method is not as sensitive to highly incompressible materials (provided that a mixed formulation is used).
- Higher order elements such as the 10-node tetrahedral, 20 and 27 node brick elements are only available in implicit analysis. They are not used in explicit analysis because no suitable mass-lumping technique is available for these elements.
- Model nonlinearity is another criterion influencing the choice between implicit and explicit solutions. As the level of nonlinearity increases, the implicit method requires more time steps, and each time step may require more iterations to converge. In some cases, no convergence is reached. The explicit method however, is less sensitive to the level of nonlinearity.

Note that when the implicit method fails it is usually due to non-convergence within a time step, while when the explicit method fails it is usually due to a diverging solution.

- The memory requirements is another factor. For the same mesh, the explicit method requires less memory since it does not store a stiffness matrix and does not require a solver. This can be significant for very large problems.
- Since ADINA handles both implicit and explicit analysis with very similar inputs, the user can in many cases restart from one analysis type to the other. This capability can be used, for example, to perform implicit springback analysis following an explicit metal forming simulation, or to perform an explicit analysis following the implicit application of a gravity load.

It can also be used to overcome certain convergence difficulties in implicit analyses. A restart from the last converged implicit solution to explicit can be performed, then, once that stage is passed, another restart from explicit to implicit can be performed to proceed with the rest of the solution.

8. Explicit dynamic analysis

This chapter presents the formulations and algorithms used to solve explicit dynamic problems, including time step calculation. A summary of the elements and material properties available for explicit analysis are listed in Table 8-1. More details are provided in the appropriate sections in Chapters 2 and 3.

Table 8-1: Elements and materials available in explicit time integration

Element type	Nodes	Elastic	Nonlin.-elast.	Elastic-orth.	Drucker-Prag.	Ilyushin	Plastic-bilin.	Plastic-mult.	Plastic-orth.	Gurson-plast.	Thermo-iso.	Thermo-orth.	Thermo-plast.	Ogden	Mooney-Riv.	Fluid	Gasket**
Truss	2	✓	✓				✓	✓			✓		✓				
Beam	2	✓					✓										
Isobeam	2	✓					✓	✓			✓		✓				
Plate	3	✓		✓	✓												
Shell (MITC)	3/4	✓		✓			✓	✓	✓		✓	✓	✓				
Shell (MITC-9)	9	✓		✓			✓	✓	✓		✓	✓	✓				
Solid (2D, displ.)	3/4/9	✓		✓	✓		✓	✓	✓		✓	✓	✓	✓	✓	✓	✓
Solid (2D, u/p)	3/4	✓					✓	✓					✓	✓	✓		
Solid (2D, u/p)	9	✓					✓	✓					✓	✓	✓		
Solid (2D, inc.mod)	3*/4	✓		✓			✓	✓	✓		✓	✓	✓				
Solid (3D, displ.)	4/5/6/8 27	✓		✓	✓		✓	✓	✓		✓	✓	✓	✓	✓	✓	✓
Solid (3D, u/p)	4/6/8	✓					✓	✓					✓	✓	✓		
Solid (3D, u/p)	5/27	✓					✓	✓					✓	✓	✓		
Solid (3D, inc.mod)	4*/8	✓		✓			✓	✓	✓		✓	✓	✓				
Disp.fluid 2-D	4																✓
Disp.fluid 3-D	8																

In addition, linear and nonlinear spring elements are available.

*) 3-node 2-D and 4-node 3-D incompatible mode elements are automatically converted by ADINA to isoparametric elements

**) No critical time step checking for gasket materials

Table 8-2 lists more element types not available in explicit analysis.

Table 8-2: Elements types not available in explicit analysis
potential-based fluid elements
porous media
pipe elements
general elements
user-supplied elements
shell elements not classified as new MITC elements (e.g., layered shells, shells with composite failure criteria, MITC8, isoparametric variable node shells, etc.)

Table 8-3 lists the ADINA options not available in explicit analysis.

Table 8-3: Elements types not available in explicit analysis
cyclic symmetry
substructuring
mapping
zooming
Thermo-mechanical coupling (TMC)
Fluid-structure interaction (FSI)
consistent mass matrix
consistent Rayleigh damping
temperatures provided via tape
load penetration (for ruptured elements)

8.1 Formulation

The central difference method (CDM) is used for time integration in explicit analysis (see ref. KJB, Section 9.2.1). In this case, it is assumed that

$${}^t\ddot{\mathbf{U}} = \frac{1}{\Delta t^2} \left({}^{t-\Delta t}\mathbf{U} - 2{}^t\mathbf{U} + {}^{t+\Delta t}\mathbf{U} \right) \quad (8.1-1)$$

and the velocity is calculated using

$${}^t\dot{\mathbf{U}} = \frac{1}{2\Delta t} \left(-{}^{t-\Delta t}\mathbf{U} + {}^{t+\Delta t}\mathbf{U} \right) \quad (8.1-2)$$

The governing equilibrium equation at time t is given by

$$\mathbf{M} {}^t\ddot{\mathbf{U}} + \mathbf{C} {}^t\dot{\mathbf{U}} = {}^t\mathbf{R} - {}^t\mathbf{F} \quad (8.1-3)$$

Substituting the relations for ${}^t\ddot{\mathbf{U}}$ and ${}^t\dot{\mathbf{U}}$ in Eq. (8.1-1) and (8.1-2), respectively, into Eq. (8.1-3), we obtain

$$\left(\frac{1}{\Delta t^2} \mathbf{M} + \frac{1}{2\Delta t} \mathbf{C} \right) {}^{t+\Delta t}\mathbf{U} = {}^t\mathbf{R} - {}^t\mathbf{F} + \frac{2}{\Delta t^2} \mathbf{M} {}^t\mathbf{U} - \left(\frac{1}{\Delta t^2} \mathbf{M} + \frac{1}{2\Delta t} \mathbf{C} \right) {}^{t-\Delta t}\mathbf{U} \quad (8.1-4)$$

from which we can solve for ${}^{t+\Delta t}\mathbf{U}$.

- ref. KJB Sections 9.2.1, 9.4 and 9.5.1*
- The central difference method has the following characteristics:
 - ▶ It is an explicit integration method, meaning that equilibrium of the finite element system is considered at time t to obtain the solution at time $t+\Delta t$.
 - ▶ When the mass and damping matrices are diagonal, no coefficient matrix needs to be factorized, see ref. KJB, p. 772. The use of the central difference method is only effective when this condition is satisfied. Therefore, only lumped mass can be used. Also damping can only be mass-proportional.

- ▶ No degree of freedom should have zero mass. This will lead to a singularity in the calculation of displacements according to Eq. 8.1-4, and will also result in a zero stable time step.
- ▶ The central difference method is conditionally stable. The time step size Δt is governed by the following criterion

$$\Delta t \leq \Delta t_{CR} = \frac{T_{Nmin}}{\pi}$$

where Δt_{CR} is the critical time step size, and T_{Nmin} is the smallest period in the finite element mesh.

- The central difference method is most effective when low-order elements are employed. Hence quadratic 3-D solid and shell elements are not allowed.
- The time step can be specified by the user, or calculated automatically. When the user specifies the time, the program does not perform any stability checking. It is the user's responsibility, in this case, to ensure that an appropriate stable time step is used.
- When automatic time step calculation is selected, the TIMESTEP entry is only used to determine the number of nominal time steps and the frequency of output of results. The stable time step is used instead of the value in TIMESTEP (unless the value in TIMESTEP is smaller).

For example, if the user requests 12 steps of size 1.0 with porthole output every 4 steps, there will be 12 nominal steps each of size 1.0. If the stable time step is smaller than 1.0 it will be used instead and results will be saved as soon as the solution time exceeds 4.0, 8.0 and exactly at 12.0 since it is the last step of the analysis.

8.1.1 Mass matrix

- The construction of the lumped mass matrix depends on the type of element used. Details are provided in the appropriate section in Chapter 2.

For elements with translational degrees of freedom only, the total mass of the element is divided equally among its nodes. For elements with rotational masses (beam and shell elements), the lumping procedure is element dependent.

Note that the lumping of rotational degrees of freedom is slightly different in implicit and explicit analysis. The rotational masses in explicit analysis are sometimes scaled up so that they do not affect the element's critical time step.

8.1.2 Damping

- Damping can be added directly to the model through Rayleigh damping. Additional indirect damping results from plasticity, friction and rate dependent penalty contact.
- Only mass-proportional Rayleigh damping is available in explicit analysis. Hence, the damping matrix \mathbf{C} in Eq. 8.1-3 is set to:

$$\mathbf{C}_{Rayleigh} = \alpha \mathbf{M}$$

where \mathbf{M} is the total lumped mass matrix.

See Ref. KJB, Section 9.3.3, for information about selecting the Rayleigh damping constant α .

8.2 Stability

*ref. KJB
Section 9.4.2*

- The stable time step for a single degree of freedom with central difference time integration is

$$\Delta t_{CR} = \frac{T_N}{\pi} = \frac{2}{\omega_N}$$

The stable time step for a finite element assembly is

$$\Delta t \leq \Delta t_{CR} = \frac{T_{N\min}}{\pi} = \frac{2}{\omega_{N\max}} \leq \frac{2}{\omega_{E\max}}$$

where $\omega_{N\max}$ is the highest natural frequency of the system, which is bound by the highest natural frequency of all individual elements in a model $\omega_{E\max}$ (see Ref. KJB, Example 9.13, p. 815).

- When automatic time step is selected, the time step size is determined according to the following relationship

$$\Delta t = K \times \Delta t_{E\min} = K \times \frac{2}{\omega_{E\max}} \quad (8.2-1)$$

where K is a factor that scales the time step.

- For most element types the critical time step can be expressed in terms of a characteristic length and a material wave speed

$$\Delta t_E = \frac{L}{c} \quad (8.2-2)$$

where the definition of the length L and the wave speed c depend on the element and material type. For all elastic-plastic materials the elastic wave is used. This condition is used in the program instead of actually evaluating the natural frequency in Eq. (8.2-1).

- Note that the critical time step calculated for all elements is only an estimate. For some elements and material combinations it is exact, and for others it is slightly conservative. However, it may not be small enough for excessively distorted elements (3-D solid and shells), and it will therefore need scaling using K factor in Eq (8.2-1).
- The time step also changes with deformation, due to the change in the geometry of the elements and the change in the wave speed through the element (resulting from a change in the material properties).

Truss and cable elements

The critical time step for a 2-node truss element is

$$\Delta t_E = \frac{L}{c}$$

where L is the length of the element, and c is the wave speed through the element

$$c = \sqrt{\frac{E}{\rho}}$$

Beam and iso-beam elements

The critical time step for the beam element is

$$\Delta t_E = \frac{L}{c} / \sqrt{1 + \frac{12I}{AL^2}}$$

where L is the length of the element, A is the element area, I is largest moment of area, and c is the wave speed through the element

$$c = \sqrt{\frac{E}{\rho}}$$

Shell elements

The critical time step for shell elements is

$$\Delta t_E = \frac{L}{c}$$

where L is a characteristic length of the element based on its area and the length of its sides, and c is the planar wave speed through the shell, which for linear isotropic elastic materials is

$$c = \sqrt{\frac{E}{\rho(1-\nu^2)}}$$

The critical time step estimated here is only approximate, and may be too large for excessively distorted shell elements.

3-D solid elements

The critical time step for the 3-D solid elements is

$$\Delta t_E = \frac{L}{c}$$

where L is a characteristic length of the element, based on its volume and the area of its sides, and c is the wave speed through the element. For linear isotropic elastic materials c is given as

$$c = \sqrt{\frac{E(1-\nu)}{\rho(1+\nu)(1-2\nu)}}$$

The critical time step estimated here is only approximate, and may be too large for excessively distorted 3-D solid elements.

Spring elements

The critical time step for a spring element is

$$\Delta t_E = \frac{2}{\omega_N} = 2 \sqrt{\frac{M_1 M_2}{K(M_1 + M_2)}}$$

where M_1 and M_2 are the masses of the two spring nodes and K is its stiffness. Massless springs are not taken into account in the calculation of the stable time step.

Rigid links

These elements are perfectly rigid and therefore do not affect the stability of explicit analysis.

8.3 Time step management

- The stable time step size has a major influence on the total simulation time. Since this time step is determined based on the

highest eigenvalue of the smallest element, a single small or excessively distorted element could considerably increase the solution time, even if this element is not relevant to the full model.

Note that the element having the smallest critical time step size is always provided in the output file.

- Ideally, all elements should have similar critical time steps. If the material properties are uniform throughout the model this means that elements should approximately have the same lengths (see Eq. 8.2-2).
- The evaluation of the critical time step for each element takes some computational time. Therefore, it does not need to be performed at every time step. The parameter NCRSTEP in the ANALYSIS DYNAMIC command determines how frequently the critical time step is reevaluated.
- The time step size for explicit analysis can be unduly small for a realistic solution time. Three features are provided to deal with this problem.
 - A global mass scaling variable can be applied to all elements in the model (the MASS-SCALE parameter in the ANALYSIS DYNAMIC command). This scale factor is applied to the densities of all elements, except scalar elements where it is applied directly to their mass.
 - Mass scaling could also be applied to elements whose automatically calculated initial time step is below a certain value (DTMIN1 parameter in ANALYSIS DYNAMIC). A mass scale factor is then applied to these elements to make their time steps reach DTMIN1. The mass scaling ratio is then held constant for the duration of the analysis.
 - Elements with automatically calculated time step smaller than a specified value (DTMIN2 parameter in ANALYSIS DYNAMIC) can be completely removed from the model. This parameter is useful for extremely small or distorted elements that do not affect the rest of the model.

- The three parameters explained above (NCRSTEP, DTMIN1 and DTMIN2) should all be used with great care to ensure that the accuracy of the analysis is not significantly compromised.

9. Frequency domain analysis

- The ADINA system includes several capabilities for the characterization of a structural response using frequency domain analysis.
 - ▶ The dynamic response of a structure submitted to a given excitation described by a response spectrum (ground or support motions typical of earthquakes or shocks) can be investigated (see Section 9.1).
 - ▶ A Fourier analysis of the time history of a point in the structure can be performed (see Section 9.2).
 - ▶ Structural response to harmonic (periodic) vibrations or to random vibrations, whether from a base motion or from applied forces, can be analyzed (see Sections 9.3 and 9.4).
 - ▶ The response of an SDOF system connected to the finite element model can also be investigated (see Section 9.5).

Much of the frequency domain analysis is performed by the AUI during post-processing.

9.1 Response spectrum analysis

- Consider the response of a finite element system to an earthquake, with a ground acceleration a_k , into direction k . We assume that all supports of the system are accelerated simultaneously with a_k . The governing equilibrium equations are (see ref. KJB)

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{C}\dot{\mathbf{u}} + \mathbf{K}\mathbf{u} = -\mathbf{M}\mathbf{d}_k a_k(t) \quad (9.1-1)$$

where \mathbf{d}_k is a direction vector with "1" in the portions of the translational degrees of freedom acting into the direction k and "0" in the other portions, and \mathbf{u} is the vector of relative displacements of the finite element system to the ground motion.

If we perform the mode superposition analysis, the governing decoupled equations are

$$\ddot{x}_i + 2\xi_i\omega_i \dot{x}_i + \omega_i^2 x_i = -\Gamma_i^k a_k(t) \quad (9.1-2)$$

where Γ_i^k are the modal participation factors for ground motion loading (see equation (9.1-28) below). The response solution is

$$x_i^k(t) = -\Gamma_i^k \left\{ \frac{1}{\omega_i \sqrt{1-\xi_i^2}} \int_0^t e^{-\xi_i \omega_i (t-\tau)} \sin \left[\omega_i \sqrt{1-\xi_i^2} (t-\tau) \right] a_k(\tau) d\tau \right\} \quad (9.1-3)$$

or

$$x_i^k(t) = -\Gamma_i^k D_i^k \quad (9.1-4)$$

If we note that only the lowest modes (up to a cut-off frequency ω_u) contribute to the response, we obtain for the complete response of the system,

$$\mathbf{u} = - \sum_{\omega_i \leq \omega_u} \Gamma_i^k D_i^k \phi_i + \mathbf{u}_r \quad (9.1-5)$$

$$\boldsymbol{\sigma} = - \sum_{\omega_i \leq \omega_u} \Gamma_i^k D_i^k \boldsymbol{\sigma}_i + \boldsymbol{\sigma}_r \quad (9.1-6)$$

where \mathbf{u}_r and $\boldsymbol{\sigma}_r$ are the residual responses due to summing the modal contributions only up to frequency ω_u .

- The modal participation factors are calculated as follows by ADINA:

$$\Gamma_i^k = \boldsymbol{\phi}_i^T \mathbf{M} \mathbf{d}_k \quad (9.1-7)$$

where $\boldsymbol{\phi}_i$ is the mode shape of mode i and \mathbf{M} is the system mass matrix.

- The modal participation factors have the following properties:

$$1) \quad \sum_{\text{all modes}} \Gamma_i^k \phi_i = \mathbf{d}_k \quad (9.1-8)$$

This formula is used in deriving the residual (static correction) formulas given below (see equation (9.1-29)), and also is used in deriving the floor response spectrum formulas in Section 9.5.

$$2) \quad \sum_{\text{all modes}} (\Gamma_i^k)^2 = (\mathbf{d}_k)^T \mathbf{M} \mathbf{d}_k \quad (9.1-9)$$

This formula states that the sum of the squares of the modal participation factors for direction k equals the structural mass for direction k , excluding mass associated with fixed degrees of freedom of the structure. Hence the “modal mass” for mode i , direction k , is $(\Gamma_i^k)^2$.

- In the response spectrum analysis, the actual dynamic response is estimated in the following manner.

Response not including residual terms: Let $S_a(\omega_i, \xi_i)$ be the design response spectrum acceleration, then the maximum accelerations for mode i are

$$(\ddot{\mathbf{u}}_i)_{\max} = \Gamma_i^k S_a(\omega_i, \xi_i) \phi_i \quad (9.1-10)$$

The maximum relative displacements for mode i are

$$(\mathbf{u}_i)_{\max} = \frac{1}{\omega_i^2} \Gamma_i^k S_a(\omega_i, \xi_i) \phi_i \quad (9.1-11)$$

The maximum stresses for mode i are

$$(\boldsymbol{\sigma}_i)_{\max} = \frac{1}{\omega_i^2} \Gamma_i^k S_a(\omega_i, \xi_i) \boldsymbol{\sigma}_i \quad (9.1-12)$$

The maximum element forces and section results for mode i are

$$(\mathbf{F}_i)_{\max} = \frac{1}{\omega_i^2} \Gamma_i^k S_a(\omega_i, \xi_i) \mathbf{F}_i \quad (9.1-13)$$

The maximum reactions for mode i are

$$(\mathbf{R}_i)_{\max} = \frac{1}{\omega_i^2} \Gamma_i^k S_a(\omega_i, \xi_i) \mathbf{R}_i \quad (9.1-14)$$

Let M_i represent any maximum nodal response component for mode i . Then to obtain the modal combination response for the selected component considering all modes with circular frequencies less than or equal to ω_u , the following modal combinations are available:

SRSS method (Square Root of the Sum of the Squares):

$$M_{SUM} = \left(\sum_{\omega_i \leq \omega_u} (M_i)^2 \right)^{\frac{1}{2}} \quad (9.1-15)$$

Absolute sum method:

$$M_{SUM} = \sum_{\omega_i \leq \omega_u} |M_i| \quad (9.1-16)$$

Ten percent method:

$$M_{SUM} = \left(\sum_{\omega_i \leq \omega_u} (M_i)^2 + 2 \sum_{\substack{\omega_i \leq \omega_u \\ \omega_j \leq \omega_u}} |M_i M_j| \right)^{\frac{1}{2}} \quad (9.1-17)$$

The second summation is done on all i and j modes having frequencies which are closely spaced, i.e., for which, with $i < j \leq u$,

$$\frac{\omega_j - \omega_i}{\omega_i} \leq 0.1$$

Double sum method, DSC method:

$$M_{SUM} = \left(\sum_{\omega_r \leq \omega_u} \sum_{\omega_s \leq \omega_u} |M_r M_s| \varepsilon_{rs} \right)^{\frac{1}{2}} \quad (9.1-18)$$

where

$$\varepsilon_{rs} = \left[1 + \left(\frac{\omega'_r - \omega'_s}{\xi'_r \omega_r + \xi'_s \omega_s} \right)^2 \right]^{-1}$$

and

$$\begin{aligned} \omega'_r &= \omega_r \left(1 - \xi_r^2 \right)^{\frac{1}{2}} \\ \xi'_r &= x_r + \frac{2}{t_d \omega_r} \end{aligned}$$

and t_d is the duration of the earthquake. In the double sum method, the absolute value signs in (9.1-18) are included; in the DSC method, the absolute value signs in (9.1-18) are omitted.

Algebraic sum method:

$$M_{SUM} = \sum_{\omega_i \leq \omega_u} M_i \quad (9.1-19)$$

CQC (complete quadratic combination) method:

$$M_{SUM} = \left(\sum_{\omega_r \leq \omega_u} \sum_{\omega_s \leq \omega_u} M_r M_s \rho_{rs} \right)^{\frac{1}{2}} \quad (9.1-20)$$

where

$$\rho_{rs} = \frac{8\sqrt{\xi_r \xi_s} (\xi_r + a \xi_s) a^{\frac{3}{2}}}{(1-a^2)^2 + 4\xi_r \xi_s a (1+a^2) + 4(\xi_r^2 + \xi_s^2) a^2}$$

and $a = \frac{\omega_s}{\omega_r}$. You have the option to include only those terms

for which $\rho_{rs} \geq \text{CUTOFF}$. CUTOFF = 0.0 implies that all terms are included, CUTOFF = 1.0 implies that only the diagonal terms are included.

- Note: The results obtained with all methods except the algebraic sum method and DSC method depend upon the coordinate system used in modeling the problem. This is because formulas (9.1-15) to (9.1-18) and (9.1-20) are applied component by component and because the sign of each component is lost (either the component is squared or the absolute value of the component is taken).

For example, to evaluate the SRSS displacement at a node, (9.1-15) is applied to the maximum x displacement, then to the maximum y displacement and finally to the maximum z displacement. Figure 9.1-1 shows the results for a two-dimensional problem. As shown, the expected result occurs only if one coordinate system direction is aligned with the response direction.

Although the figure illustrates only displacements, the same phenomenon occurs for all results; accelerations, velocities, displacements, reactions, stresses and element forces.

This observation also applies to accelerations, velocities and displacements when skew systems are used at nodes.

Effect of residual (static correction) terms: For the response spectrum method the residual (static correction) terms are calculated using the zero period accelerations S_{ZPA} . Then the following residual terms can be calculated:

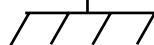
Residual accelerations:

$$\ddot{\mathbf{u}}_r = \left(\mathbf{d}_k - \sum_{\omega_i \leq \omega_u} \Gamma_i^k \boldsymbol{\phi}_i \right) S_{ZPA} \quad (9.1-21)$$

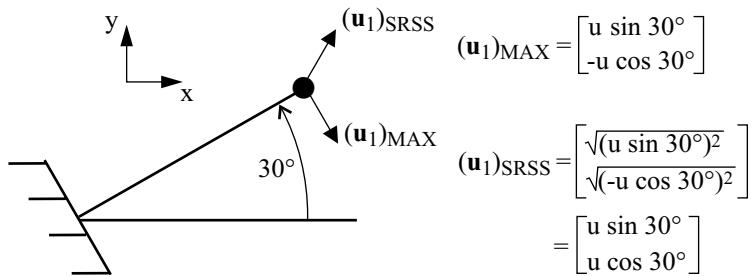
$$(\mathbf{u}_1)_{MAX} = (\mathbf{u}_1)_{SRSS}$$

$$(\mathbf{u}_1)_{MAX} = \begin{bmatrix} \mathbf{u} \\ 0 \end{bmatrix}$$

$$(\mathbf{u}_1)_{SRSS} = \begin{bmatrix} \sqrt{\mathbf{u}^2} \\ \sqrt{0^2} \end{bmatrix} = \begin{bmatrix} \mathbf{u} \\ 0 \end{bmatrix}$$



a) Model coordinate system aligned with the response direction



b) Model coordinate system not aligned with the response direction

Figure 9.1.1: Effects of choice of coordinate system on model combination responses

Residual displacements:

$$\mathbf{u}_r = \mathbf{K}^{-1} \mathbf{M} \left(\mathbf{d}_k - \sum_{\omega_i \leq \omega_u} \Gamma_i^k \boldsymbol{\phi}_i \right) S_{ZPA} \quad (9.1-22)$$

Residual stresses, residual element forces and residual reactions are denoted as $\boldsymbol{\sigma}_r$, \mathbf{F}_r , \mathbf{R}_r respectively. These residual

quantities will occur when the displacements of the model are equal to the residual displacements \mathbf{u}_r .

- Note that when the residual quantities are graphically displayed separately, then S_{ZPA} is set to unity. When the residual quantities are to be included in a modal combination, however, the expressions shown in Eqs. (9.1-21) and (9.1-22) are, of course, used.
- Let M_r denote a residual response quantity, then the modal combination response quantity, including the residual term, can be calculated as follows:

SRSS addition of residual:

$$M = \left[(M_{SUM})^2 + (M_r)^2 \right]^{\frac{1}{2}} \quad (9.1-23)$$

Absolute addition method including sign: When the SRSS, absolute sum, ten percent, double sum or CQC method is used:

$$M = |(M_{SUM})| + |M_r|$$

When the algebraic sum method is used:

$$M = M_{SUM} + \text{SIGN}(M_r, M_{SUM}) \quad (9.1-24)$$

where

$$\text{SIGN}(M_r, M_{SUM}) = \begin{cases} |M_r| & \text{if } M_{SUM} \geq 0 \\ -|M_r| & \text{if } M_{SUM} < 0 \end{cases}$$

Effect of excitations in multiple directions: In Eqs. (9.1-1) to (9.1-24) we considered the solution of the response due to earthquake excitation in one direction only. To consider excitation in more than one direction, the following spatial combination methods are available:

SRSS addition of spatial components:

$$M_{SPATIAL} = \left[\sum_k (\alpha_k M_k)^2 \right]^{\frac{1}{2}} \quad (9.1-25)$$

Absolute addition of spatial components:

$$M_{SPATIAL} = \sum_k |\alpha_k M_k| \quad (9.1-26)$$

where α_k is a factor for scaling of the component result M_k .

Effect of static load case: Let M_{ST} be a response quantity due to a static load case which has been calculated by ADINA. This static response quantity would represent the condition of the structure at the time when the earthquake occurs and can include, for example, the static response due to gravity load. The final spatial combination can then be obtained as

$$M = M_{ST} + \text{SIGN}(M_{SPATIAL}, M_{ST}) \quad (9.1-27)$$

where

$$\text{SIGN}(M_{SPATIAL}, M_{ST}) = \begin{cases} M_{SPATIAL} & \text{if } M_{ST} \geq 0 \\ -M_{SPATIAL} & \text{if } M_{ST} < 0 \end{cases} \quad (9.1-28)$$

- Initial calculations for response spectrum analysis must be performed in the ADINA analysis solution; the actual response spectrum analysis being performed during post-processing with the AUI.

Note that any number of steps of linear static solution response calculations can be performed prior to the response spectrum initial calculations in ADINA.

- The initial calculations performed in ADINA are given in the following. Note that these results are stored in the porthole file.
 - The modal participation factors Γ_i^k for each direction k of ground translational excitation and for each eigenmode, using

equation (9.1-7). Note that the number of eigenmodes used in the response spectrum calculation must be less than or equal to the number of eigenmodes available; otherwise, the program issues an error message and stops the execution.

- ▶ The modal stress vector \mathbf{s}_i for each specified eigenmode for all elements in which stress output is requested. See Section 6.2.3 for more information.

These stresses do not have a "physical meaning". They are used for calculating the spectral response of the structure.

If the nodal force output option is used in an element, the modal forces corresponding to a displacement vector equal to ϕ_i are instead calculated, printed and stored.

- In addition, initial calculations can be performed for static corrections. If requested, ADINA calculates, prints and stores on the porthole file the following data for each direction k of excitation.

- ▶ Residual accelerations $\ddot{\mathbf{u}}_R^k$, where

$$\ddot{\mathbf{u}}_R^k = \mathbf{d}_k - \sum_{i=1}^N \boldsymbol{\Gamma}_i^k \boldsymbol{\phi}_i \quad (9.1-29)$$

- ▶ Residual displacements \mathbf{u}_R^k for the total structural system, where

$$\mathbf{u}_R^k = {}^{t_0}\mathbf{K}^{-1}\mathbf{M}\ddot{\mathbf{u}}_R^k \quad (9.1-30)$$

The definitions for the variables on the right side of the equation are given above and ${}^{t_0}\mathbf{K}$ is the stiffness matrix at the start time of the solution.

- ▶ Residual forces \mathbf{F}_R^k for the total structural system, where

$$\mathbf{F}_R^k = \mathbf{M}\ddot{\mathbf{u}}_R^k \quad (9.1-31)$$

- ▶ Residual stresses \mathbf{s}_R^k for all elements which specify stress output, where

$$\mathbf{s}_R^k = {}^{t_0}\mathbf{C} {}^{t_0}\mathbf{B} \mathbf{u}_R^k \quad (9.1-32)$$

and ${}^{t_0}\mathbf{C}$, ${}^{t_0}\mathbf{B}$ are the element stress-strain matrix and strain-displacement matrix, respectively at the start time of the solution.

If the nodal force output option is used in an element, the residual forces corresponding to a displacement vector equal to \mathbf{u}_R^k are instead calculated, printed and stored.

- ▶ Residual reactions, consistent with the residual displacements.
- During post-processing the AUI performs the response spectrum analysis. You specify the following information during post-processing, to obtain the response for a single ground motion direction:
 - ▶ Loading response spectra (in terms of displacements, velocities or accelerations) for various damping ratios
 - ▶ Ground motion direction
 - ▶ Damping ratios of the modes
 - ▶ Type of modal combination
 - ▶ Type of combination of residual terms

In the AUI, all of the information needed for the calculation for one ground motion direction is entered into a response of type response-spectrum.

To perform spatial combination calculations for more than one ground motion direction, define a response of type response-combination and include the response spectrum responses for the individual ground motion directions in the definition of the response-combination response.

To include the effects of a static load case, define a response of type load-step for the static load case, then define a response of type response-combination and include the response spectrum responses and the static load case response in the definition of the response-combination response.

See Section 12.1.4 for information about responses in the AUI.

- Additional information about response spectrum analysis can be found in the following references:

ref. R.W. Clough and J. Penzien, *Dynamics of Structures*, McGraw Hill, 1974.

ref. U.S. Nuclear Regulatory Guide 1.92, *Combining Modal Responses and Spatial Components in Seismic Response Analysis*, Revision 1, February 1976.

9.2 Fourier analysis

- Given a time history $f(t)$, defined from t_0 to t_1 , the Fourier series corresponding to $f(t)$ is

$$f(t) = a_0 + \sum_{n=1}^{\infty} (a_n \cos n\omega_l t + b_n \sin n\omega_l t)$$

where

$$\omega_l = \frac{2\pi}{t_p}, \quad t_p = t_1 - t_0$$

$$a_0 = \frac{1}{t_p} \int_{t_0}^{t_1} f(t) dt$$

$$a_n = \frac{2}{t_p} \int_{t_0}^{t_1} f(t) \cos n\omega_l t dt, \quad b_n = \frac{2}{t_p} \int_{t_0}^{t_1} f(t) \sin n\omega_l t dt$$

Since

$$a_n \cos n\omega_l t + b_n \sin n\omega_l t = c_n \cos(n\omega_l t - \phi_n)$$

where

$$c_n = \sqrt{a_n^2 + b_n^2}, \quad \phi_n = \tan^{-1} \frac{b_n}{a_n}$$

$f(t)$ may be written as

$$f(t) = c_0 + \sum_{n=1}^{\infty} c_n \cos(n\omega_l t - \phi_n)$$

- Conceptually, the function $f(t)$ is considered to contain the frequencies $0, \omega_l, 2\omega_l, \dots$. The constants c_n and ϕ_n are the amplitude and phase angle of that portion of $f(t)$ which oscillates at frequency $n\omega_l$.
- A simple relationship (Parseval's identity) exists between the Fourier coefficients and the mean square amplitude of $f(t)$:

$$\frac{1}{t_p - t_0} \int_{t_0}^{t_p} f(t)^2 dt = c_0^2 + \frac{1}{2} \sum_{n=1}^{\infty} c_n^2$$

- The power spectral density of $f(t)$ for frequency $f_n = \frac{n\omega_l}{2\pi}$ may be defined as

$$S_f(f_0) = \frac{c_0^2}{\Delta f}$$

$$S_f(f_n) = \frac{c_n^2}{2\Delta f}, \quad n = 1, 2, \dots$$

where $\Delta f = \frac{\omega_1}{2\pi}$ is the interval between two successive frequencies

(in cycles/unit time). The mean square value of $f(t)$ may be expressed in terms of the power spectral density using

$$\frac{1}{t_p} \int_{t_0}^{t_1} f^2 dt = \sum_{n=1}^{\infty} S_f(f_n) \Delta f$$

It is seen that the mean square value of $f(t)$ is related to the area under the power spectral density curve.

- The AUI permits $f(t)$ to be defined as the value of any variable over any time interval contained within the solution time. The time step need not be constant in the finite element analysis. The AUI permits an additional time interval t_{gap} during which $f(t)$ is zero to be added to the variable response. These concepts are shown in Fig. 9.2-1.

Time history of variable: Time history used in Fourier analysis:

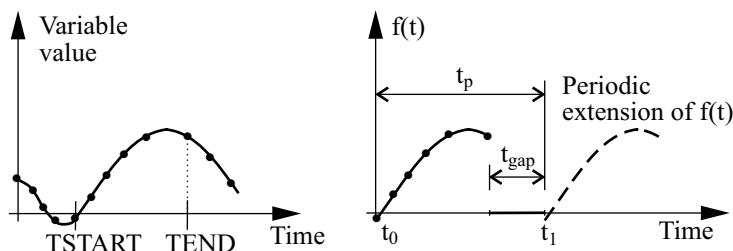


Figure 9.2-1: Fourier analysis

- The Fourier series is truncated by choosing a highest frequency f_{max} . This frequency is chosen based on the time step to avoid aliasing errors:

$$f_{max} \leq \frac{0.5}{\Delta t|_{max}}$$

Hence, $N = \text{int}(t_p f_{max})$ gives the upper limit of the Fourier series summation

$$f(t) = c_0 + \sum_{n=1}^N c_n \cos(n\omega_l t - \phi_n)$$

- The Fourier coefficient integrals are evaluated using the trapezoidal rule for numerical integration. These integrals are typically integrated with less accuracy for higher frequencies. Because a fast Fourier transform algorithm is not used, the computer time required to form the Fourier coefficients is proportional to N^2 .

9.3 Harmonic vibration analysis

- It is frequently of interest to determine the steady-state structural response to given harmonic and random loads. This structural response is conveniently expressed as the superposition of modal responses. This section summarizes the theory of harmonic vibration analysis and discusses those features of the AUI that perform harmonic vibration analysis.
- Consider the finite element system

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{C}\dot{\mathbf{u}} + \mathbf{K}\mathbf{u} = \sum_{k=1}^N \mathbf{F}_k(t) \quad (9.3-1)$$

where N load cases are being considered. If we perform a mode superposition analysis, the governing decoupled equations are

$$\ddot{x}_i + 2\xi_i\omega_i\dot{x}_i + \omega_i^2 x_i = \sum_{k=1}^N \Gamma_i^k b_k(t) \quad (9.3-2)$$

where Γ_i^k is the modal participation factor for load case k and mode i .

These equations apply for the following cases:

Case 1: Structure is grounded and forces $\mathbf{F}_k(t)$ are applied.

Here k is the load case number:

$$\mathbf{F}_k(t) = \mathbf{F}_{k0} b_k(t) \quad (9.3-3)$$

where $b_k(t)$ is a dimensionless multiplier for load case k .

Case 2: A single ground motion is applied in direction k :

$$\mathbf{F}_k(t) = -\mathbf{M}\mathbf{d}_k b_k(t) \quad (9.3-4)$$

where $b_k(t)$ is the ground acceleration in direction k .

Case 2 is seen to be a special case of case 1, except that in case 1 the displacements are interpreted as absolute displacements and in case 2 the displacements are interpreted as relative displacements.

Each load case is seen to have a spatial variation and a time variation. The time variation appears within the function $b_k(t)$. In harmonic analysis, we consider functions $b_k(t)$ of the form

$$b_k(t) = b_{k0}(t) \sin(\omega_k t - \alpha_k) \quad (9.3-5)$$

where b_{k0} is a function of the forcing frequency ω_k .

- We also note here that the transient structure response is ignored. In most cases, only the mean-square value of a response quantity can be predicted. However, in harmonic vibration analysis with all loads acting with the same loading frequency, the maximum amplitude and phase of a response quantity can be predicted as well.

Harmonic vibration analysis with all loads acting with the same loading frequency

- Consider the time variation

$$b_k(t) = b_{k0}(t) \sin(\omega_k t - \alpha_k) \quad (9.3-6)$$

for load case k , in which each b_{k0} is a function of the forcing frequency, and we write ω instead of ω_k because the forcing frequency ω is assumed to be the same for each load case k . In this case, the steady-state modal response is

$$x_i(t) = \sum_{k=1}^N (A_{ki} \sin(\omega t - \alpha_k) - B_{ki} \cos(\omega t - \alpha_k)) \quad (9.3-7)$$

where

$$A_{ki} = \frac{b_{k0} \Gamma_i^k}{\omega_i^2} \frac{1 - \left(\frac{\omega}{\omega_i} \right)^2}{\left(1 - \left(\frac{\omega}{\omega_i} \right)^2 \right)^2 + \left(2 \frac{\omega}{\omega_i} \xi_i \right)^2} \quad (9.3-8)$$

$$B_{ki} = \frac{b_{k0} \Gamma_i^k}{\omega_i^2} \frac{2 \frac{\omega}{\omega_i} \xi_i}{\left(1 - \left(\frac{\omega}{\omega_i} \right)^2 \right)^2 + \left(2 \frac{\omega}{\omega_i} \xi_i \right)^2} \quad (9.3-9)$$

and the steady-state solution is then

$$\mathbf{u}(t) = \sum_{i=1}^n \boldsymbol{\phi}^{(i)} \sum_{k=1}^N x_{ki}(t) \quad (9.3-10)$$

where n is the number of eigenvectors considered in the mode superposition. Equation (9.3-10) is of course only an approximation to the steady-state solution response when n is less than the total number of degrees of freedom in the model. However the approximation may be adequate when enough eigenvectors are considered and when harmonic loads of low enough frequency are considered.

If we consider a single degree of freedom L , then

$$\begin{aligned} u_L(t) &= \sum_{i=1}^n \phi_L^{(i)} \sum_{k=1}^N x_{ki}(t) \\ &= \left(\sum_{i=1}^n \phi_L^{(i)} \sum_{k=1}^N C_{ki} \right) \sin \omega t - \left(\sum_{i=1}^n \phi_L^{(i)} \sum_{k=1}^N D_{ki} \right) \cos \omega t \end{aligned} \quad (9.3-11)$$

where

$$\begin{aligned} C_{ki} &= A_{ki} \cos \alpha_k - B_{ki} \sin \alpha_k \\ D_{ki} &= A_{ki} \sin \alpha_k + B_{ki} \cos \alpha_k \end{aligned}$$

which can be rewritten as

$$u_L(t) = u_{L0} \sin(\omega t - \theta_L) \quad (9.3-12)$$

where

$$u_{L0} = \sqrt{\left(\sum_{i=1}^n \phi_L^{(i)} \sum_{k=1}^N C_{ki} \right)^2 + \left(\sum_{i=1}^n \phi_L^{(i)} \sum_{k=1}^N D_{ki} \right)^2} \quad (9.3-13)$$

$$\theta_L = \tan^{-1} \left(\frac{\sum_{i=1}^n \phi_L^{(i)} \sum_{k=1}^N D_{ki}}{\sum_{i=1}^n \phi_L^{(i)} \sum_{k=1}^N C_{ki}} \right) \quad (9.3-14)$$

u_{L0} and θ_L are seen to be the maximum amplitude and phase angle, respectively.

- The quasi-static amplitude of the displacements for forcing frequency ω is determined by assuming that the load $b_k(t)$ is applied with a very low forcing frequency. One way to determine the quasi-static amplitude for forcing frequency ω is to set ω to zero within equations (9.3-8) and (9.3-9) (but using b_{k0} corresponding to forcing frequency ω) and then evaluate equation (9.3-13) as described above. This method is approximate because the static structural response may not be well modeled in a modal

superposition analysis; in other words, the contribution of the remaining modes to the static response may not be negligible.

The ratio of the dynamic maximum amplitude to the quasi-static maximum amplitude is sometimes referred to as the dynamic magnification factor or the dynamic load factor.

- Equations (9.3-11) to (9.3-14) also apply to any quantity that is proportional to the displacements, such as element forces, stresses and reaction forces.
- Considering the velocity and acceleration, we see from equation (9.3-12) that

$$\dot{u}_L(t) = \omega u_{L0} \sin\left(\omega t - \left(\theta_L - \frac{\pi}{2}\right)\right) \quad (9.3-15)$$

and

$$\ddot{u}_L(t) = \omega^2 u_{L0} \sin\left(\omega t - (\theta_L - \pi)\right) \quad (9.3-16)$$

from which the maximum amplitude and phase are easily seen.

- For more insight into the above equations, we consider only one load case ($N=1$) and draw (9.3-6) and (9.3-12) in the complex plane (Fig. 9.3-1). Notice that the actual response is the imaginary component of the vectors.

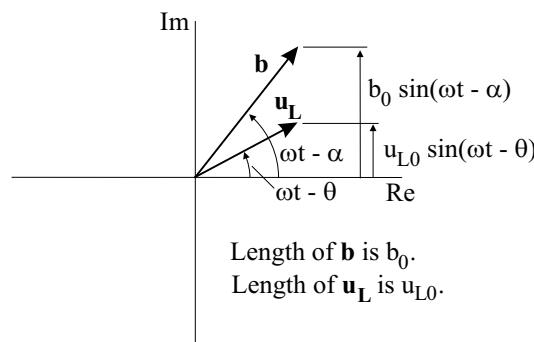


Figure 9.3-1: Force and displacement representations in the complex plane

To obtain the “in-phase” or “real” part of the response, we set

$\omega t - \alpha_k$ to 90° . This orients the force vector to be entirely imaginary (Fig 9.3-2). The in-phase part of the response is then obtained from (9.3-12).

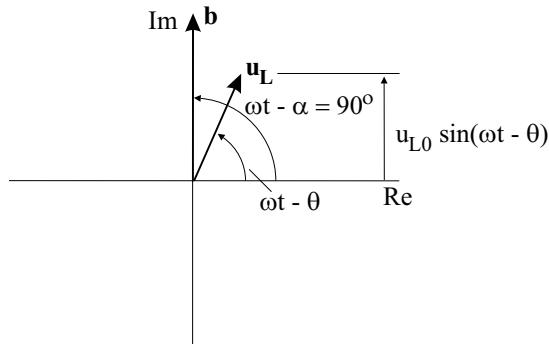


Figure 9.3-2: Obtaining the in-phase part of the displacements

To obtain the “out-of-phase” or “imaginary” part of the response, we set $\omega t - \alpha_k$ to 0° . This orients the force vector to be entirely real (Fig 9.3-3). The in-phase part of the response is then obtained from (9.3-12).

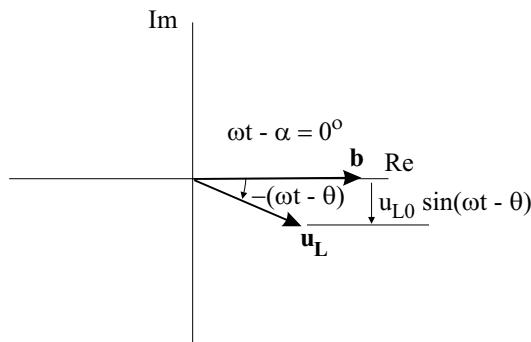


Figure 9.3-3: Obtaining the out-of-phase part of the displacements

The reason why we use 90° to obtain the real part of the response is because equations (9.3-6) and (9.3-12) are written in terms of the sin function, not the cos function.

- Additional information about harmonic vibration analysis can be found in the following references:

ref. R.W. Clough and J. Penzien, *Dynamics of Structures*, McGraw-Hill, 1974.

ref. W.C. Hurty and M.F. Rubinstein, *Dynamics of Structures*, Prentice Hall, 1964.

- Concerning the usage of the harmonic vibration analysis feature, ADINA computes the natural frequencies, mode shapes (including modal stresses and forces) and modal participation factors. The type of modal participation factor can either be applied load or ground motion. When the type of modal participation factor is applied load, ADINA computes the load vector for each time step in the analysis as if the analysis was a linear static analysis. However ADINA does not solve for the displacements corresponding to the load vector, rather ADINA computes the modal participation factor.
- During post-processing the AUI performs the harmonic vibration analysis. You specify the following information during post-processing:

- ▶ Sweep spectrum giving the dependence of b_{k0} on the forcing frequency
- ▶ Load step number (for applied loads) or ground motion direction (for ground motion loading)
- ▶ Damping ratios of the modes
- ▶ Type of result (maximum amplitude, phase angle, RMS amplitude or signed amplitude for a given value of ωt)

Then you can obtain either the entire structural response for a single forcing frequency or the response at a point for a range of forcing frequencies.

- In order to obtain the in-phase or out-of-phase part of the response, select signed amplitude for the type of result and set ωt to 90° (to obtain the in-phase part of the response) or to 0° (to obtain the out-of-phase part of the response). (We assume here that

the loading phase angle α_k is equal to 0.)

- The harmonic vibration analysis information is entered into a response of type harmonic. See Section 12.1.4 for information about responses in the AUI.

9.4 Random vibration analysis

- In the ADINA system, the only type of result that can be predicted in random vibrations is the mean-square response, defined as

$$\overline{p^2} = \lim_{T \rightarrow \infty} \frac{1}{2T} \int_{-T}^T p(t)^2 dt \quad (9.4-1)$$

for the function $p(t)$. This response can also be written as an integral in the frequency domain as

$$\overline{p^2} = \lim_{T \rightarrow \infty} \frac{1}{2\pi} \int_0^T \frac{|\tilde{p}(\omega)|^2}{T} d\omega \quad (9.4-2)$$

in which

$$\tilde{p}(\omega) = \int_{-\infty}^{\infty} p(t) \exp(-i\omega t) dt \quad (9.4-3)$$

is the Fourier transform of $p(t)$ and

$$|\tilde{p}(\omega)|^2 = \tilde{p}(\omega) \tilde{p}^*(\omega) \quad (9.4-4)$$

where $\tilde{p}^*(\omega)$ is the complex conjugate value of $\tilde{p}(\omega)$. The mean-square response in the frequency domain can also be written in terms of the power spectral density function

$$\hat{y}(\omega) = \frac{|\tilde{y}(\omega)|^2}{T} \quad (9.4-5)$$

as follows:

$$\bar{y}(t)^2 = \frac{1}{2\pi} \int_0^\infty \hat{y}(\omega) d\omega = \int_0^\infty \hat{y}(f) df \quad (9.4-6)$$

where f is the frequency in cycles/(unit time). Using Fourier transform theory, the Fourier transform of the modal response can be written in terms of the Fourier transform of the loads:

$$\tilde{x}_i(\omega) = \sum_{k=1}^N \Gamma_i^k \tilde{H}_i(\omega) \tilde{b}_k(\omega) \quad (9.4-7)$$

where

$$\tilde{H}_i(\omega) = \frac{1}{(\omega_i^2 - \omega^2) + i(2\xi_i \omega_i \omega)} \quad (9.4-8)$$

Therefore, for component u_L ,

$$\tilde{u}_L(\omega) = \sum_{i=1}^n \phi_L^{(i)} \sum_{k=1}^N \Gamma_i^k \tilde{H}_i(\omega) \tilde{b}_k(\omega) \quad (9.4-9)$$

and therefore

$$\begin{aligned} |\tilde{u}_L(\omega)|^2 &= \\ &\sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^N \sum_{l=1}^N \phi_L^{(i)} \phi_L^{(j)} \Gamma_i^k \Gamma_j^l \tilde{H}_i(\omega) \tilde{H}_j^*(\omega) \tilde{b}_k(\omega) \tilde{b}_l^*(\omega) \end{aligned} \quad (9.4-10)$$

To simplify equation (9.4-10), the approximations

$$\tilde{H}_i(\omega)\tilde{H}_j^*(\omega) = 0 \quad \text{for } \frac{\min(\omega_i, \omega_j)}{\max(\omega_i, \omega_j)} < \text{CUTOFF} \quad (9.4-11)$$

and

$$\tilde{b}_k(\omega)\tilde{b}_l^*(\omega) \approx |\tilde{b}_k(\omega)|^2 \delta_{kl} \quad (9.4-12)$$

are made. Regarding eq. (9.4-11), when CUTOFF = 0, no approximation is implied and when CUTOFF = 1, eq. (9.4-11) reduces to

$$\tilde{H}_i(\omega)\tilde{H}_j^*(\omega) = |\tilde{H}_i(\omega)|^2 \delta_{ij}$$

which is a good approximation if the structure is lightly damped and when the mean-square response is to be calculated (as in equation (9.4-15) below). Equation (9.4-12) is equivalent to neglecting the cross-spectral densities of the loading combinations.

Using eqs (9.4-11) and (9.4-12), equation (9.4-10) reduces to

$$|\tilde{u}_L(\omega)|^2 = \sum_{k=1}^N \sum_{i=1}^n \sum_{j=1}^n (\phi_L^{(i)})(\phi_L^{(j)}) (\Gamma_i^k)(\Gamma_j^k) |\tilde{H}_i(\omega)| |\tilde{H}_j^*(\omega)| |\tilde{b}_k(\omega)|^2 \quad (9.4-13)$$

Using the definition of the power-spectral-density, we write

$$\hat{u}_L(\omega) = \sum_{k=1}^N \sum_{i=1}^n \sum_{j=1}^n (\phi_L^{(i)})(\phi_L^{(j)}) (\Gamma_i^k)(\Gamma_j^k) |\tilde{H}_i(\omega)| |\tilde{H}_j^*(\omega)| \hat{b}_k(\omega) \quad (9.4-14)$$

and therefore

$$\bar{u}_L^2(\omega) = \sum_{k=1}^N \sum_{i=1}^n \sum_{j=1}^n (\phi_L^{(i)})(\phi_L^{(j)}) (\Gamma_i^k)(\Gamma_j^k) \int_0^\infty |\tilde{H}_i(f)| |\tilde{H}_j^*(f)| \hat{b}_k(f) df \quad (9.4-15)$$

In (9.4-14 and 9.4-15), the input power-spectral-density $\hat{b}_k(f)$ satisfies

$$\bar{b}_k(t)^2 = \frac{1}{2\pi} \int_0^\infty \hat{b}_k(\omega) d\omega = \int_0^\infty \hat{b}_k(f) df \quad (9.4-16)$$

where $\bar{b}_k(t)^2$ is the mean-square value of the applied load (compare (9.4-6) and (9.4-16)). Different authors use different conventions, for example

$$\bar{b}_k(t)^2 = \int_0^\infty \hat{b}_k(\omega) d\omega,$$

$$\bar{b}_k(t)^2 = \int_{-\infty}^\infty \hat{b}_k(f) df,$$

$$\bar{b}_k(t)^2 = \int_{-\infty}^\infty \hat{b}_k(\omega) d\omega$$

are all possible definitions of the input power-spectral-density. However, ADINA accepts only the convention given in (9.4-16).

Equations (9.4-14) and (9.4-15) are used when the response quantity is proportional to the displacements. When the response quantity is proportional to the velocity or the acceleration, then the transfer function given in equation (9.4-8) has to be modified:

$$\text{Velocity : } \tilde{H}_i(\omega) = \frac{i\omega}{(\omega_i^2 - \omega^2) + i(2\xi_i\omega_i\omega)}$$

$$\text{Acceleration : } \tilde{H}_i(\omega) = \frac{-\omega^2}{(\omega_i^2 - \omega^2) + i(2\xi_i\omega_i\omega)}$$

Interpretation of mean-square responses: The mean-square value of a response quantity can be used to predict the probability that the actual response will exceed any given value. This is done by assuming that the response has a Gaussian distribution and has zero mean value. Then it may be shown that the standard deviation is well approximated by the root-mean-square (RMS) value of the response quantity:

$$\sigma = \sqrt{u_L^2} \quad (9.4-17)$$

Hence we can write

$$P_r(|u_L| > \lambda\sigma) = erfc\left(\frac{\lambda}{\sqrt{2}}\right) \quad (9.4-18)$$

where the value of the erfc function can be found in standard tables:

$$P_r(|u_L| > \sigma) = 0.3173$$

$$P_r(|u_L| > 2\sigma) = 0.0455$$

$$P_r(|u_L| > 3\sigma) = 0.0027$$

For further details, see reference:

ref. W.C. Hurty and M.F. Rubinstein, *Dynamics of Structures*, Prentice-Hall, 1964.

- Concerning the usage of the random vibration analysis feature, ADINA computes the same quantities as in harmonic vibration analysis (see above).
- During post-processing the AUI performs the random vibration analysis. You specify the following information during post-processing:
 - ▶ Power-spectral-density $\hat{b}_k(f)$. Note that $\hat{b}_k(f)$ must satisfy (9.4-16). If the input power-spectral-density is obtained using another convention, you must convert it so that (9.4-16) is satisfied.
 - ▶ Load step number (for applied loads) or ground motion direction (for ground motion loading)
 - ▶ Damping ratios of the modes

Then you can obtain either the RMS response or the power-spectral-density of the response at a point.

- The random vibration analysis information is entered into a response of type random. See Section 12.1.4 for information about responses in the AUI.

9.5 SDOF system response

- It is frequently of interest to determine the maximum response of single degree of freedom (SDOF) systems to SDOF support displacement excitation as a function of the SDOF system natural frequency.
- The support is a node in the finite element mesh (called the connection point) and the SDOF systems are considered to be attached to the support. The SDOF systems are assumed to be much lighter than the structure, so that the SDOF system response does not affect the structural response.
- This type of analysis is called floor response spectrum analysis. The maximum SDOF system response can be computed when the structure time history is given, or when the response spectrum corresponding to structural ground motion is given.
- As an example of this type of analysis, consider a machine within a building that is much lighter than the building. Because the machine is much lighter than the building, it is assumed that the machine response does not affect the building response, and therefore the machine response calculations can be done as a postprocessing step once the building response (without the machine) is known.

The response of the building to an earthquake can be computed either using a time history analysis (when the earthquake ground motion is given as a function of time) or using a response spectrum analysis (when the earthquake ground motion is given as a response spectrum). The response of the machine can also be computed in each case (Figs. 9.5-1 and 9.5-2) by idealizing the machine as a SDOF system attached to a point in the building.

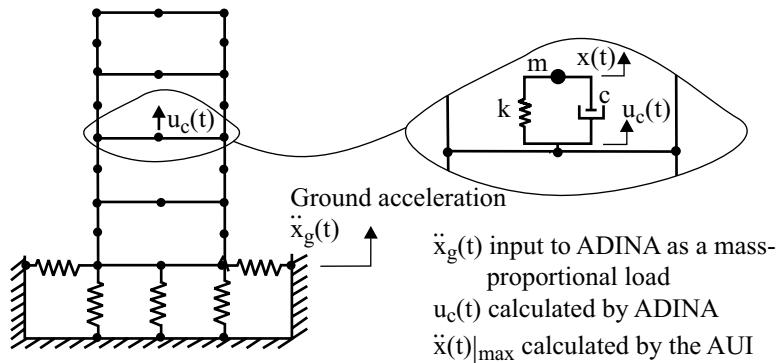


Figure 9.5-1: Floor response spectrum analysis from time history data

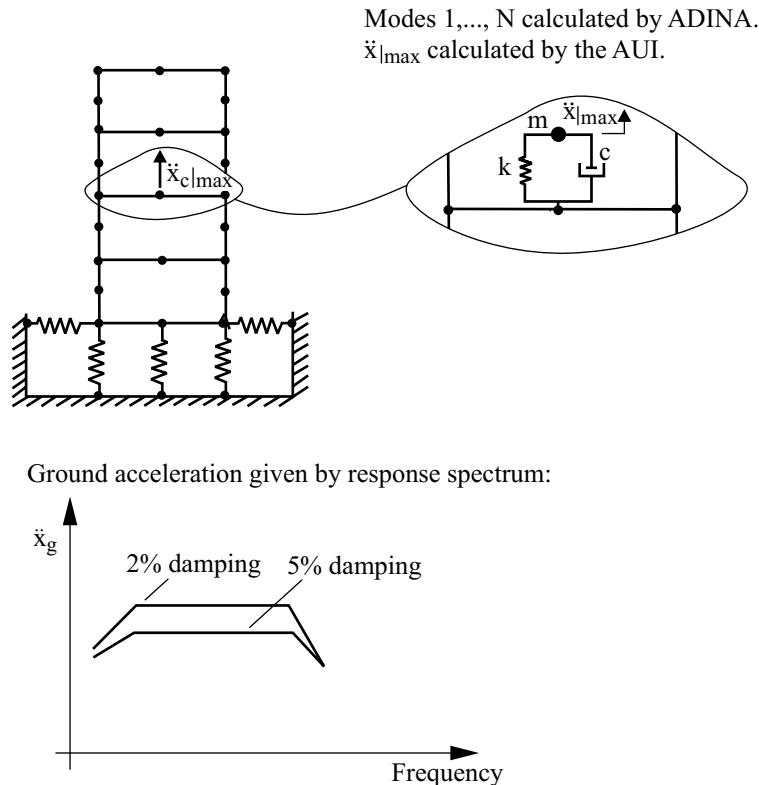


Figure 9.5-2: Floor response spectrum analysis from response spectrum data

- Considering many possible machines, each with a different natural frequency and damping ratio, it is useful to create floor response spectrum curves (Fig. 9.5-3). Each floor response spectrum curve shows, for a given machine damping ratio, the maximum machine response as a function of the machine natural frequency. One possible use of the floor response spectrum curves is in the selection of a machine mount that will minimize the machine's motion in the event of an earthquake, for example, by selecting the mount stiffness so that the machine has the natural frequency with the least response.

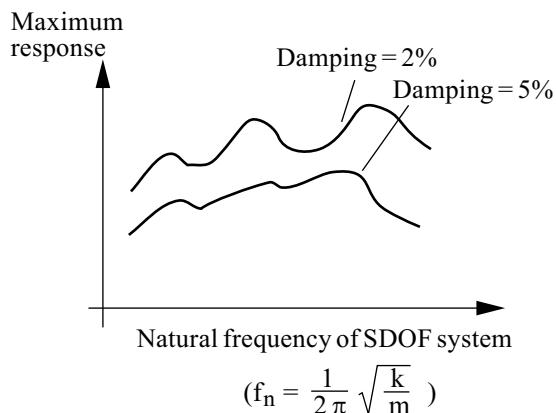


Figure 9.5-3: Floor response spectrum curves

Structural loading given by one or more time histories

- The governing equation of motion for the SDOF system is

$$m\ddot{u} + c\dot{u} + ku = ku_c + c\dot{u}_c - m\ddot{x}_g \quad (9.5-1)$$

where u is the relative displacement of the SDOF system to ground motions applied as mass-proportional loads, u_c is the relative displacement of the connection point to ground motions applied as mass-proportional loads and \ddot{x}_g are ground accelerations applied as mass-proportional loads. Equation (9.5-1) is written in terms of relative displacements because the nodal displacements, velocities and accelerations output by ADINA are relative to the ground

motion when ground motions are applied as mass-proportional loads.

Equation (9.5-1) is numerically integrated using the Newmark method with $\delta = 1/2$, $\alpha = 1/4$ from rest initial conditions and the maximum response is recorded. The maximum response can be the maximum values of $|u|, |\dot{u}|, |\ddot{u}|, |u - u_c|, |\dot{u} - \dot{u}_c|, |\ddot{u} - \ddot{u}_c|, |x|, |\dot{x}|$, or $|\ddot{x}|$. The connection point displacement, velocity and acceleration are taken directly from the ADINA output, so no numerical integration is required to evaluate these quantities. When the absolute displacement or velocity of the SDOF system are required, the AUI numerically integrates the ground accelerations to obtain ground velocities and displacements using the trapezoidal rule.

The damping ratio used for the SDOF systems can be constant or frequency-dependent.

- The AUI can combine the maximum responses from several time histories using the floor response spectrum from time history command. In this case, the AUI computes the maximum response separately for each time history and combines the results using either the SRSS method or the absolute sum method (see Section 9.1).

This feature is most useful when you have several separate time history analyses of a structure as separate ADINA runs, each time history analysis giving the structural response to loading acting in a different direction, and you want to compute the maximum response corresponding to the loads acting simultaneously. First, load the ADINA porthole files into a single database, using the "append" feature to load all but the first porthole file. Then compute the response corresponding to combined loads, in which you define response-ranges for each time history, then specify each response-range in a separate data input line. (Note: appending the porthole files shifts the time intervals associated with each time history so that the time intervals do not overlap.)

- The spacing of SDOF system frequencies can be linear, logarithmic or the recommended USNRC or ASME frequency spacings, see references:
 - ref. *1992 ASME Boiler and Pressure Vessel Code, Rules for Construction of Nuclear Power Plant Components*, Division 1, Appendix N.
 - ref. U.S. Nuclear Regulatory Guide 1.122, *Development of Floor Design Response Spectra for Seismic Design of Floor-Supported Equipment or Components*, rev. 1, Feb. 1978.

In addition, the SDOF system response is evaluated for each frequency listed in a frequency table. This feature is useful for evaluating the response at structural natural frequencies.

- After the AUI constructs a response curve, it can broaden peaks and smooth the curve. There are two peak broadening algorithms that you can use: peak broadening=yes and peak broadening=all. These are shown in Figure 9.5-4.

During smoothing, response values are raised (never lowered) from their unsmoothed values until the curve segments are as close to straight as possible. By default, it is set to 5 % of the maximum response value in the curve.

Peak broadening can be used without smoothing, or smoothing can be used without peak broadening, but these features are most effective when used together.

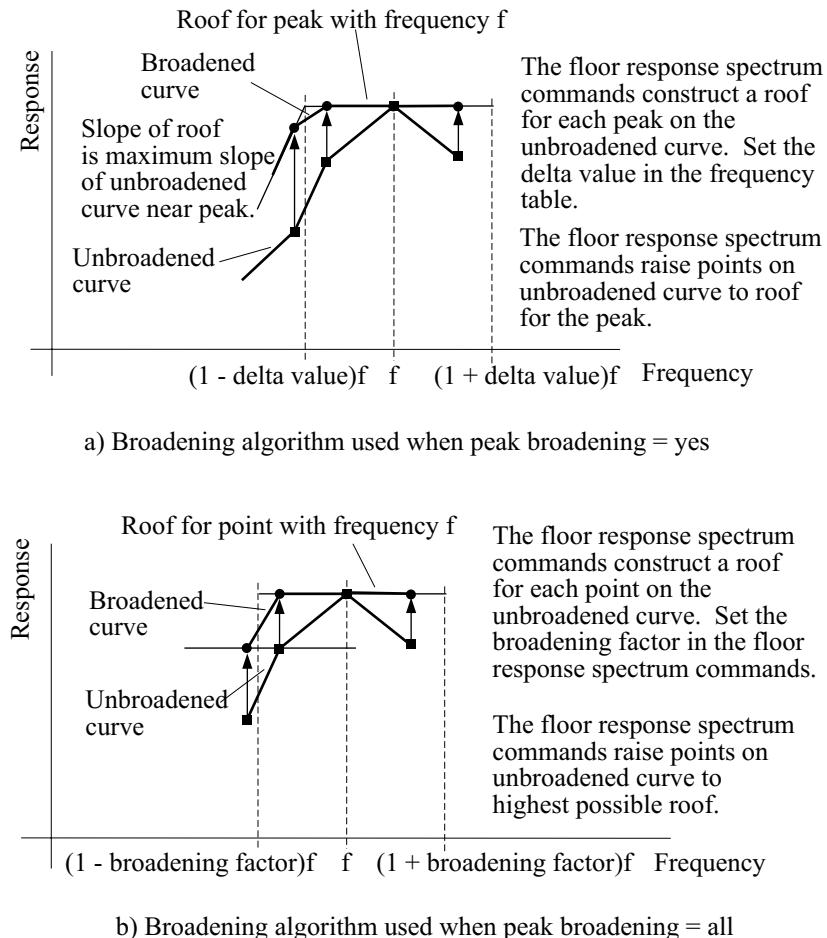


Figure 9.5-4: Floor response spectrum broadening algorithms

Structural loading specified by response spectra

- Frequently the structural loading is one or more ground motions specified as response spectra. In this case it is not possible to compute the structural response using a time history analysis. Instead the structural response is estimated using the response spectrum method (see Section 9.1).

The technique, which is presented in the following reference,

- ref. K.A. Peters, D. Schmitz and U. Wagner, "Determination of Floor Response Spectra on the Basis of the Response Spectrum Method," *Nuclear Engineering and Design*, Vol. 44, pp. 255-262, 1977.

can be summarized as follows:

Start with the eigenvalues, eigenvectors, modal participation factors and modal damping ratios of the structure. These are termed

$$\omega_i, \phi_i, \Gamma_i^k, \xi_i$$

for modes $i = 1, \dots, N$ and ground motion direction k . The eigenvectors are of length n , where n is the number of degrees of freedom of the structure. For example, the eigenvalues, eigenvectors and modal participation ratios are calculated by ADINA in response spectrum analysis and the modal damping ratios are specified in the AUI (ADINA-PLOT).

The addition of a SDOF system to the structure contributes an extra eigenvalue, eigenvector, modal participation factor and modal damping factor to the mathematical model. These are termed

$$\omega_i^*, \phi_i^*, (\Gamma_i^k)^*, \xi_i^*$$

where the * is used to denote the structure plus SDOF system mathematical model. The addition of the SDOF system also modifies the structure eigenvalues, eigenvectors, modal participation factors and modal damping factors for modes $i = 1, \dots, N$:

$$\begin{aligned} \omega_i &\rightarrow \omega_i^*, \quad \phi_i \rightarrow \phi_i^*, \\ \Gamma_i^k &\rightarrow (\Gamma_i^k)^*, \quad \xi_i \rightarrow \xi_i^* \end{aligned}$$

The eigenvectors are now of length $n + 1$, with the additional entry in each eigenvector denoted ϕ_{i0}^* .

Consider a SDOF system that vibrates in direction ℓ . In order to estimate the maximum absolute acceleration of the SDOF system

due to ground motion in direction k , we use the response spectrum formula

$$\left(\ddot{x}_0^k\right)_{\max} = \text{modal-combination of } \left(\ddot{x}_{0i}^k\right)_{\max}, \quad i = 0, \dots, N \quad (9.5-2)$$

where

$$\left(\ddot{x}_{0i}^k\right)_{\max} = \left(\Gamma_i^k\right)^* \phi_{i0}^* S_a(\omega_i^*, \xi_i^*) \quad (9.5-3)$$

is the maximum acceleration due to mode i and $S_a(\omega_i^*, \xi_i^*)$ is the response spectrum acceleration corresponding to natural frequency ω_i^* and damping ratio ξ_i^* . Any of the modal combination methods listed in Section 9.1 can be used for the modal combination formula. For example, if the SRSS method of modal combination is used,

$$\left(\ddot{x}_0^k\right)_{\max} = \left(\sum_{i=1}^N \left[\left(\Gamma_i^k\right)^* \phi_{i0}^* S_a(\omega_i^*, \xi_i^*) \right]^2 \right)^{\frac{1}{2}} \quad (9.5-4)$$

We see from eq. (9.5-3) that we need $\left(\Gamma_i^k\right)^*$, ϕ_{i0}^* , $i = 1, \dots, N$ and $\left(\Gamma_0^k\right)^* \phi_{00}^*$ to apply the response spectrum method to the structure plus SDOF system mathematical model. From the last reference above, if the SDOF system is very light compared with the structure,

$$\phi_i^* \approx \begin{bmatrix} \Phi \\ \overline{\phi_{i0}^*} \end{bmatrix} \quad (9.5-5a)$$

$$\omega_i^* \approx \omega_i, \quad \left(\Gamma_i^k\right)^* \approx \Gamma_i^k, \quad \xi_i^* \approx \xi_i, \quad i = 1, \dots, N \quad (9.5-5b,c,d)$$

Also,

$$\phi_{i0}^* \approx \frac{1}{1 - \left(\frac{\omega_i}{\omega_0}\right)^2} \phi_{ia} \quad (9.5-6)$$

where a denotes the degree of freedom corresponding to the node to which the SDOF system is connected and direction ℓ . (We note that the fraction in this equation is similar to the transfer function for a SDOF system with natural frequency ω_0 and zero damping subjected to harmonic support motion). Finally,

$$\left(\Gamma_0^k\right)^* \phi_{00}^* = \delta_{kl} - \sum_{i=1}^n \left(\Gamma_i^k\right)^* \phi_{i0}^* \quad (9.5-7)$$

where $\delta_{k\ell}=1$ for $k=\ell$ and $\delta_{k\ell}=0$ otherwise. Equation (9.5-7) follows from equation (9.1-8).

Since we normally have fewer than n eigenvectors, we make the approximation

$$\left(\Gamma_0^k\right)^* \phi_{00}^* \approx \delta_{kl} - \sum_{i=1}^N \left(\Gamma_i^k\right)^* \phi_{i0}^* \quad (9.5-8)$$

by neglecting

$$\sum_{i=N+1}^n \left(\Gamma_i^k\right)^* \phi_{i0}^* = \sum_{i=N+1}^n \Gamma_i^k \frac{1}{1 - \left(\frac{\omega_i}{\omega_0}\right)^2} \phi_{ia} \quad (9.5-9)$$

From the right-hand-side of equation (9.5-9), we see that the neglected term is small provided that the natural frequency of the SDOF system is much smaller than the natural frequencies of the neglected modes.

Introducing equations (9.5-5), (9.5-6) and (9.5-7) into equation (9.5-3) gives the response spectrum approximation to the maximum absolute acceleration of the SDOF system. We term this

approximation the undamped transfer function, no resonance correction method because equation (9.5-6) corresponds to an undamped transfer function and there is no check for the SDOF system natural frequency being close to a structural natural frequency.

When the SDOF system natural frequency is close to a structural natural frequency, the denominator of equation (9.5-6) tends to zero. In this case, we use the following equations

$$\begin{aligned} \left(\ddot{x}_0^k \right)_{\max} &= \text{modal-combination of } \left(\ddot{x}_{0i}^k \right)_{\max} \text{ and } \left(\ddot{x}_{0j}^k \right)_{\max}, \\ i &= 1, \dots, N, \quad i \neq j \end{aligned} \tag{9.5-10}$$

where

$$\left(\ddot{x}_{0j}^k \right)_{\max} = \frac{\left(\Gamma_j^k \right)^* \phi_{ja}^* S_a(\omega_j^*, \xi_j^*)}{2\sqrt{\xi_0(\xi_0 + \xi_j)}} \tag{9.5-11}$$

is an estimate of the maximum acceleration of the SDOF system due to the motion of that mode j with natural frequency closest to the natural frequency of the SDOF system. It is recommended that the results using both equations (9.5-2) and (9.5-10) be calculated and the smaller result taken; we term this method the undamped transfer function, resonance correction method.

We have found that equation (9.5-10) can be smaller than equation (9.5-2) even for SDOF system natural frequencies not close to any structural natural frequency, for example when the structural motion of mode j does not affect the SDOF system motion. When this happens, the response predictions are not accurate. Therefore we introduce a third method, the damped transfer function method, which is identical to the undamped transfer function, no resonance correction method, except that equation (9.5-6) is replaced by

$$\phi_{i0}^* \approx \pm \sqrt{\frac{1 + \left(2\xi_{eff} \frac{\omega_i}{\omega_0}\right)^2}{\left(1 - \left(\frac{\omega_i}{\omega_0}\right)^2\right)^2 + \left(2\xi_{eff} \frac{\omega_i}{\omega_0}\right)^2}} \phi_{ia} \quad (9.5-12)$$

where + is used if $\omega_i \leq \omega_0$ and - is used otherwise. The term ξ_{eff} in this equation is evaluated using one of the following choices:

$$\xi_{eff} = \xi_0 \quad (9.5-12a)$$

$$\xi_{eff} = \sqrt{\xi_0 (\xi_0 + \xi_j)} \quad (9.5-12b)$$

$$\xi_{eff} = \sqrt{\xi_0^2 \pm (\xi_0 - \xi_j)^2}, + \text{if } \xi_0 \leq \xi_j, - \text{if } \xi_0 > \xi_j \quad (9.5-12c)$$

Equation (9.5-12) corresponds to the transfer function for a damped SDOF system subjected to support motion. The damped transfer function method avoids numerical difficulties when the SDOF system natural frequency equals a structural natural frequency and gives results close to the undamped transfer function, no resonance correction method otherwise, provided that the SDOF damping ratio is small (which is usually the case).

- The maximum relative velocity and maximum relative displacement corresponding to ground motions in direction k can be estimated using similar techniques by substituting the spectrum velocity S_v or spectrum displacement S_d for the spectrum acceleration S_a in the above formulas. Note that, although the calculated acceleration is an absolute acceleration, the velocities and displacements are relative to the ground motion.
- Once the responses are known for a single load case, the response for a combination of load cases can be computed as follows:

$$(\ddot{x}_0)_{\max} = \text{spatial-combination of } (\ddot{x}_0^k)_{\max}, \quad k=1, \dots, N \quad (9.5-13)$$

For example, when the SRSS method is used to combine three load cases, each load case corresponding to ground motions in one of the three coordinate directions,

$$(\ddot{x}_0)_{\max} = \left[\sum_{k=1}^3 (\ddot{x}_0^k)^2 \right]^{\frac{1}{2}} \quad (9.5-14)$$

- The options available for selecting SDOF system frequencies, damping values, broadening peaks and smoothing curves are exactly the same as when calculating floor response spectrum curves using time history data, see above.

10. Fracture mechanics

10.1 Overview

- Linear and nonlinear fracture mechanics analysis can be performed with the ADINA system.
- The ADINA fracture mechanics capabilities include the computation of conservation criteria (J-integral, energy release rate) in 2-D and 3-D finite element models, and the analysis of actual crack propagation in 2-D finite element models.
- The computation of conservation criteria, based on the domain theory, is applicable to small displacement/small strain, large displacement/small strain, and large displacement/large strain analysis. Additional information about the theory used in ADINA can be found in the following references:
 - ref. J.K. Knowles and E. Stenberg, "On a Class of Conservation Laws in Linearized and Finite Elastostatics," *Archiv. Rat. Mech. Anal.*, Vol. 44, pp. 187-211, 1972.
 - ref. S.N. Atluri, "Path-independent integrals in finite elasticity and inelasticity, with body forces, inertia, and arbitrary crack-face conditions," *Engng. Fracture Mech.*, Vol. 16, pp. 341-364, 1982.
 - ref. Rice J.R., "A Path Independent Integral and the Approximate Analysis of Strain Concentration by Notches and Cracks," *J. Appl. Mech.*, Vol. 35, pp. 379-386, 1968.
 - ref. T.L. Anderson, *Fracture Mechanics, Fundamentals and Applications*, CRC Press, 1991.
- Based upon the above theoretical results, two different numerical methods are available for the computation of the conservation criteria: the **line contour method** and the **virtual**

crack extension method. More information about these methods can be found in the following references:

- ref. Hellen T.K., "On the Method of Virtual Crack Extensions," *Int. J. Num. Meth. Engng.*, Vol. 9, pp. 187-207, 1975.
 - ref. Parks D.M., "The Virtual Crack Extension Method for Nonlinear Material Behavior," *Comp. Meth. Appl. Mech. Engng.*, Vol. 12, pp. 353-364, 1977.
 - ref. H.G. deLorenzi, 'On the energy release rate and the J-integral for 3-D crack configurations', *Int. J. Fract.*, Vol. 19, pp. 183-193, 1982.
 - ref. Guillermin, O., "Some Developments in Computational Methods for Nonlinear Fracture Mechanics," *Proceedings IKOSS FEM '91*, Baden-Baden, Germany, 1991.
- An interface for user-supplied fracture mechanics solutions also exists in ADINA (see Section 10.6).
 - The material models that can be used in fracture mechanics are: **isotropic-elastic**, **orthotropic-elastic**, **plastic-bilinear**, **plastic-multilinear**, **Drucker-Prager**, **plastic-orthotropic**, **thermo-isotropic**, **thermo-orthotropic**, **thermo-plastic**, **creep**, **plastic-creep**, **multilinear-plastic-creep**, **creep-variable**, **plastic-creep-variable**, **multilinear-plastic-creep-variable**, **concrete**, **curve-description**, **user-supplied**.

Any of the allowed kinematic formulations for the material models employed can be used in fracture mechanics.

- For the analysis of crack propagation, the "node shift/release" technique is used in ADINA. With this technique, which includes local remeshing of the crack front during crack propagation, a smooth and continuous advance of the crack can be modeled.
- Currently, fracture mechanics analysis can be performed with a model including one crack only.

- The crack line or surface can be located on the boundary of the finite element model (symmetric specimen and loading) or inside the finite element model (asymmetric specimen and/or loading). However, for crack propagation analysis, the specimen and loading must be symmetric.
- In 2-D analysis, the crack line must be parallel to the Y axis, and in 3-D analysis, the crack propagation surface must be parallel to the X-Y plane.

10.2 The line contour integration method

- The contour integration method is used in two-dimensional analysis to calculate a contour independent parameter (J-integral) characterizing the severity of the displacement, stress and strain fields at the tip of a crack. The expression of this J-integral is given by (see Fig. 10.2-1)

$$J = \int_{\Gamma} \left(W dx_2 - \sigma_{ij} \frac{\partial u_i}{\partial x_j} n_j ds \right) \quad (10.2-1)$$

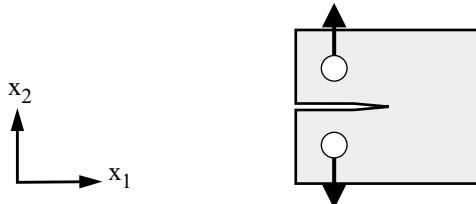
where

- Γ = line contour enclosing the crack tip
- σ_{ij} = components of the stress tensor
- u_i = components of the displacement vector
- n_j = components of the unit vector normal to Γ
- ds = length increment along Γ
- W = stress work density

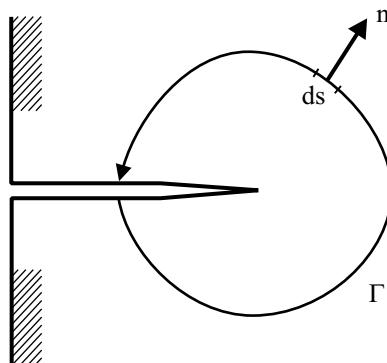
- Note that initial strains are included in the computation of the J-integral.
- Note that the application of the contour integration method to three-dimensional analysis involves some complexity, which makes it cumbersome and unattractive when compared to the virtual crack extension method described in Section 10.3. Therefore, the contour integration method has not been implemented in ADINA for three-dimensional analysis.

- For linear elastic materials, the J-integral is related to the stress intensity factors K_I , K_{II} , and K_{III} by

$$J = \frac{K_I^2}{E'} + \frac{K_{II}^2}{E'} + \frac{K_{III}^2}{G}$$



a) Physical problem



b) Definition of the line contour

Figure 10.2-1: Arbitrary contour around a crack tip for J-integral calculation

with $E' = E$ in plane stress and $E' = \frac{E}{1-\nu^2}$ in plane strain.

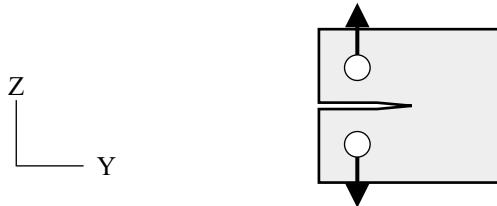
- Note that when the J-integral defined above is employed to characterize cracks in elastic-plastic or nonlinear elastic bodies, the following conditions must theoretically be satisfied:

- ▶ Isothermal analysis

- ▶ No body forces and inertia forces; no pressure on the crack faces
- ▶ Monotonic and proportional loading; no unloading
- ▶ Material homogeneous in the x_1 direction

Definition of the line contours

- In ADINA, the line contour Γ of Eq. (10.2-1) is defined by a series of adjoining segments passing through the elements located on the contour (see Fig. 10.2-2). The integration of the J-integral along each segment is performed numerically using the value of the variables at the integration points which define the segment.



a) Physical problem

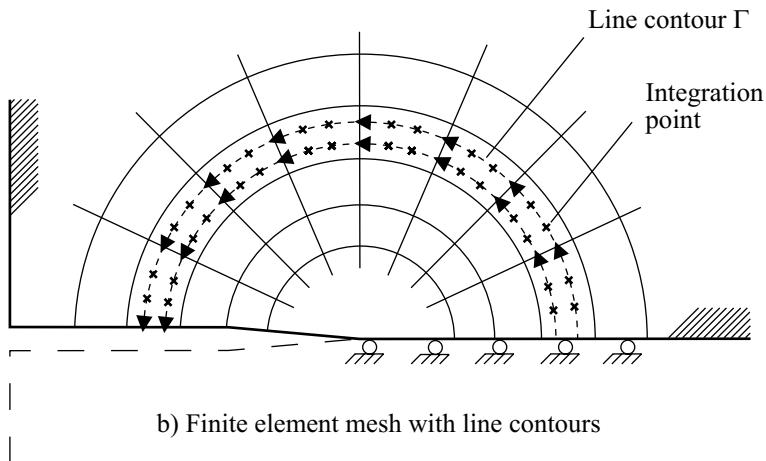


Figure 10.2-2: Definition of line contours in a two-dimensional finite element model

- As many contours as desired can be requested in ADINA. Usually, several contours are used simultaneously in order to assess the J-integral path-independence.
- 2-D elements with 2×2 or 3×3 Gauss integration can be used. All the elements on one contour must have the same integration order. If 2×2 Gauss integration is used, the J-integral is evaluated simultaneously along two line contours (see Fig. 10.2-3). If 3×3 Gauss integration is used, only one J-integral is evaluated, using a middle path through each element (see Fig. 10.2-3).

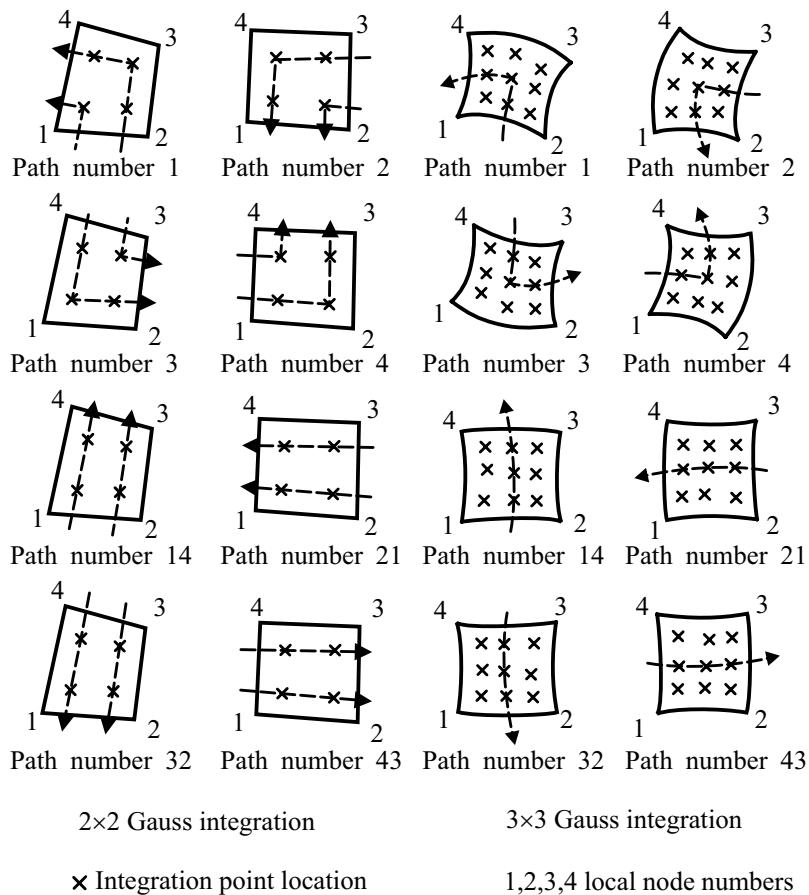


Figure 10.2-3: Convention for the description of element paths

- In order to describe a contour directly in terms of the element numbers, the element numbers (and the group numbers if necessary) must be input in a sequence following the line contour from one end to the other. For the first and last elements on the contour, at least one face must coincide with the physical boundary of the model. The convention used for defining the element paths is given in Fig. 10.2-3.
- The AUI provides commands for the automatic generation of line contours.

Output

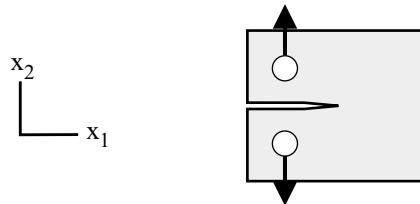
- The material properties at the crack tip location are printed in ADINA at every step of the analysis.
- Each line contour outputs the following information to the porthole file: `LINE_J-INTEGRAL_AVERAGE`, `LINE_J-INTEGRAL_DIFFERENCE`. If 2×2 Gauss integration is used, `LINE_J-INTEGRAL_AVERAGE` is the average of the two J-integrals evaluated and `LINE_J-INTEGRAL_DIFFERENCE` is the difference between these values. If 3×3 Gauss integration is used, `LINE_J-INTEGRAL_AVERAGE` is the value of the J-integral and `LINE_J-INTEGRAL_DIFFERENCE` is zero.

10.3 The virtual crack extension method

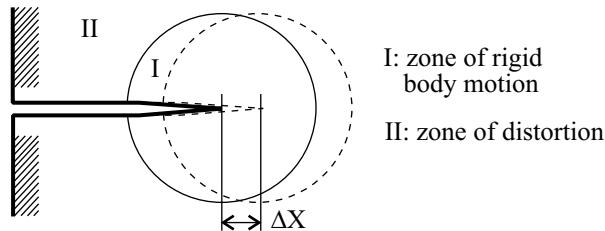
10.3.1 J-integral and energy release rate

- The virtual crack extension method evaluates the J-integral for a given body using the difference of the total potential energy between two configurations with slightly different cracks. The basic idea is shown in Fig 10.3-1.
- The virtual crack extension method requires only one finite element analysis.
- The total potential energy variation is calculated using a "virtual material shift" (zone I in Fig. 10.3-1) obtained by shifting the nodes

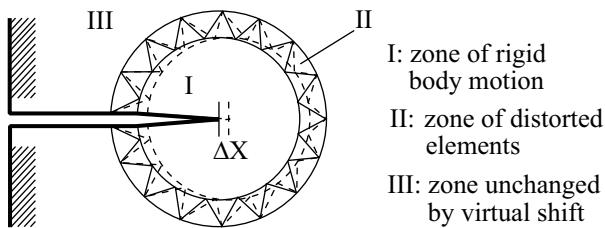
of a domain which includes at least one of the crack front nodes.



a) Physical problem



b) Virtual crack extension (enlarged)



c) Application to finite element analysis

Figure 10.3-1: The virtual crack extension method in two-dimensional analysis

- The equivalence between the J-integral and the ratio of the total potential energy variation to the crack area increase holds only for linear elastic analysis, and elasto-plastic analysis when the deformation theory of plasticity is applicable.
- In the following, the "energy release rate" refers to the ratio of the total potential energy variation to the crack area increase.
- Compared to the line contour method, the virtual crack

extension method is preferred for the calculation of the J-integral.

- The virtual crack extension method is used in ADINA to calculate the generalized expression of the energy release rate including the effects of thermal loads, pressure loads on the faces of the crack and inertia forces.
- Once developed, the expression of the energy release rate reads (see Fig. 10.3-1):

$$G = \frac{1}{A_c} \int_V \left(\left(\sigma_{ij} \frac{\partial u_j}{\partial x_k} - W \delta_{ik} \right) \frac{\partial \Delta X_k}{\partial x_j} - f_i \frac{\partial u_i}{\partial x_j} \Delta X_j \right) dV - \frac{1}{A_c} \int_S t_i \frac{\partial u_i}{\partial x_j} \Delta X_j dS \quad (10.3-1)$$

with

V = volume of the cracked body

S = surface of the cracked body

ΔX_k = components of virtual crack extension vector

A_c = increase in crack area

δ_{ij} = Kronecker delta

f_i = components of the body force vector

t_i = components of the surface traction vector

$W = \int_0^{\varepsilon_{ij}} \sigma_{ij} d\varepsilon_{ij}$ = total stress work density.

Note that initial strains are included in the computation of the energy release rate.

Hoop stress correction: In axisymmetric analysis with 2-D elements, the following corrective term must be added to the right-hand side of equation 10.3-1:

$$G_{hoop} = \frac{1}{A_C} \int_V \Delta X_2 (\sigma_{xx} \varepsilon_{xx} - W) dV \quad (10.3-2)$$

where ΔX_2 is the virtual crack extension component in the y direction.

Thermal correction: For the analysis of structures with temperature dependent material properties, the effect of thermal loads must be taken into account in the calculation of the energy release rate. The following corrective term must be added to the right-hand side of equation 10.3-1:

$$G_{th} = \frac{1}{A_c} \int_V \left(\sigma_{ij} \frac{\partial \varepsilon_{ij}^{th}}{\partial x_k} \Delta X_k \right) dV \quad (10.3-3)$$

in which ε_{ij}^{th} are the components of the thermal strain tensor.

This correction is currently available in ADINA for all temperature dependent material models, except the thermo-orthotropic material model.

Pressure correction: The general expression of the energy release rate includes a contribution from the surface tractions acting on the boundaries of the cracked body. In particular, when a pressure load is applied on the faces of the crack, the surface traction vector reduces to $(0,0,-p)$, and the contribution becomes

$$G_{pr} = \frac{1}{A_c} \int_S t_i \frac{\partial u_i}{\partial x_j} \Delta X_j dS = -\frac{1}{A_c} \int_{S_p} p \left(\frac{\partial u'_3}{\partial x'_1} \Delta X'_1 + \frac{\partial u'_3}{\partial x'_2} \Delta X'_2 \right) \quad (10.3-4)$$

with

- S_p = total surface of the crack faces
- x'_1, x'_2 = local Cartesian directions aligned with the crack surface
- x'_3 = local Cartesian direction normal to the crack surface

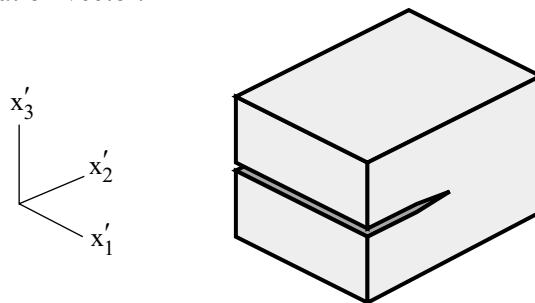
In ADINA, this pressure correction can be requested for linearly varying pressure load sets (see Fig. 10.3-2).

- Note that in the analysis of crack propagation, the pressure load is updated as the crack opens.

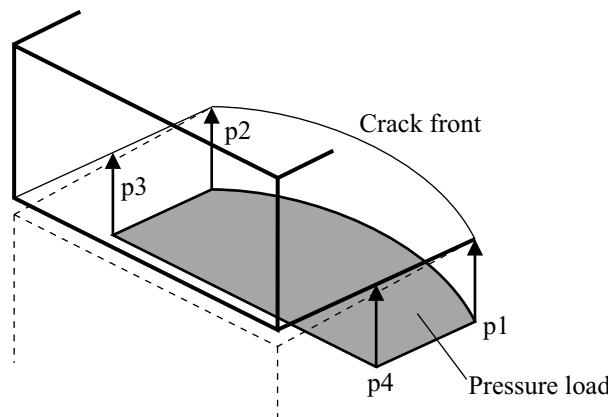
Dynamic correction: In dynamic analysis, the effect of inertia forces is included in the calculation of the energy release rate by adding the following contribution to equation (10.3-1):

$$G_{dyn} = \frac{1}{A_c} \int_V \left(\rho \ddot{u}_i \frac{\partial u_i}{\partial x_j} \Delta X_j \right) dV \quad (10.3-5)$$

in which ρ is the mass density and \ddot{u}_i are the components of the acceleration vector.



a) Geometry of fracture mechanics specimen



b) Pressure load on crack face

Figure 10.3-2: Pressure load on a cracked body

10.3.2 Virtual material shifts

Description of the virtual material shift

- In order to describe a virtual material shift, a specified domain simply connected to the crack tip node must be selected (see Fig. 10.3-3). The nodes inside this domain are subjected to a virtual shift in order to obtain the virtual configuration of the finite element mesh.

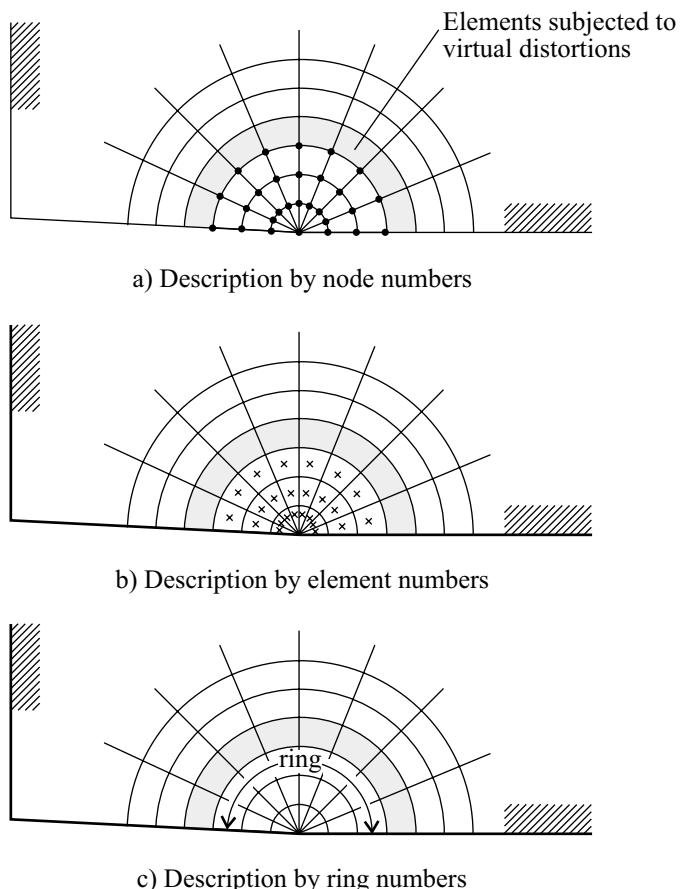


Figure 10.3-3: Description of the virtual material shifts in ADINA

In ADINA, each virtual shift domain can be described using node numbers, using element numbers, or using ring numbers (see Fig. 10.3-3). You should verify that the virtual shift domains are simply connected. If a virtual shift domain does not contain at least one crack tip node, the fracture mechanics calculations are not performed. Several virtual shift domains can be defined simultaneously in one analysis.

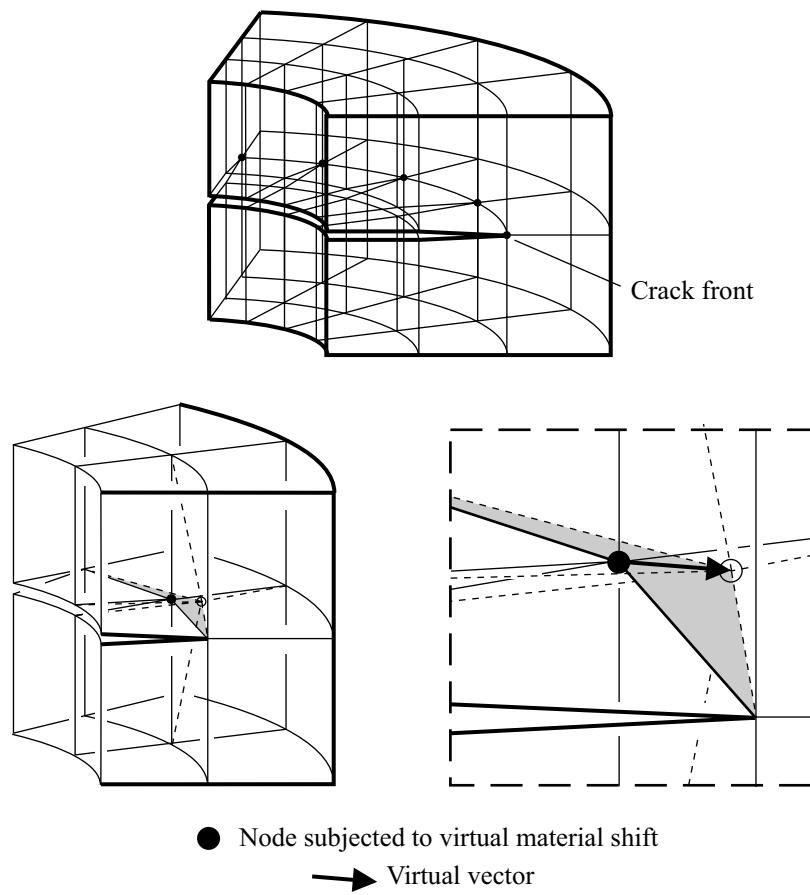


Figure 10.3-3: (continued)

- Midside nodes cannot be used as a virtual shift origin.

- Note that only the distorted elements (zone II in Fig. 10.3-1), between the undeformed zone I and the rigid body motion zone III, add a contribution to the energy release rate. Therefore the choice of the location of zone II in the finite element model will determine the accuracy of the result.
- It is recommended that several virtual material shifts of increasing size be used to evaluate the value of the energy release rate. Each virtual shift should enclose the next smaller virtual shift. The virtual shift size should increase equally in all directions from the plane perpendicular to the crack front.

Virtual vector: The virtual vector defines the amplitude and the direction of the virtual shift (Fig. 10.3-4). The same vector is applied at every node inside the virtual shift domain. If midside nodes are present in distorted elements (zone II in Fig. 10.3-1), the corresponding virtual vectors are calculated by linear interpolation of virtual vectors at corner nodes.

In ADINA, a virtual vector is defined by its components in the global Cartesian system. An automatic calculation of the virtual vector can be requested for a given virtual material shift: in this case, the crack propagation surface must have been defined (see Section 10.4), and the first two nodes on the generation line corresponding to the crack tip node associated with the virtual material shift are used to calculate the virtual vector (see Fig. 10.4-4). The length of the virtual vector is 1/100 of the length of the vector formed by the two nodes. It is recommended that this option be used when a crack propagation analysis is performed.

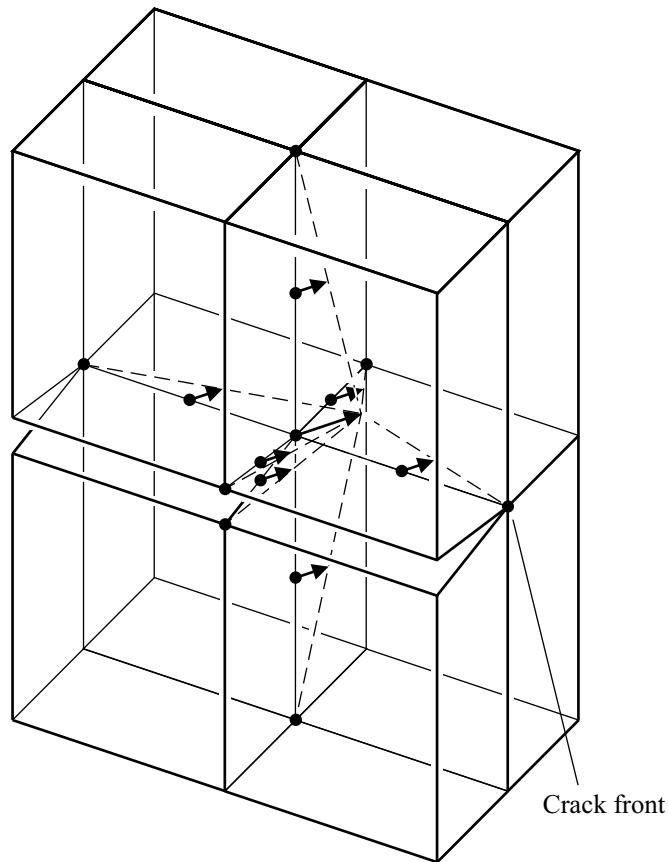
For a symmetric specimen and loading, the virtual vector lies in the plane of the crack. For an asymmetric specimen and/or loading, the virtual vector direction is usually unknown in advance, because the direction of future cracking is not usually in the plane of the crack.

10.3.3 Results

- Each virtual shift prints the following information, and outputs the following information to the porthole file. This information is accessed in the AUI using the given variable names.

J-PARAMETER_1: Energy release rate without corrections (not recommended for general use)

J-PARAMETER_2: Energy release rate without thermal, pressure or dynamic corrections, from equations (10.3-1) and (10.3-2).



- Nodes subjected to a virtual material shift

Figure 10.3-4: Virtual vectors

J-PARAMETER_3: Energy release rate with temperature, pressure and dynamic corrections, from equations (10.3-1) to (10.3-5).

- TEMPERATURE_CORRECTION,
- PRESSURE_CORRECTION,
- DYNAMIC_CORRECTION: The corrections
- Strain energy densities at integration points can be printed and/or saved (see Section 11.7).

10.4 Crack propagation

- The analysis of crack propagation is available in ADINA for 2-D solids. The crack should be planar, on the symmetry line of a symmetric structure submitted to a symmetric loading.
- The "node shift/release" technique used in ADINA combines the shifting and the releasing of the successive crack tip nodes in order to model the propagation of the crack tip through the finite element mesh (see Fig. 10.4-1).
- The "node release-only" technique can also be used, whereby the crack tip node is not shifted but only released. This option provides better convergence when large deformations occur and convergence difficulties arise with the node shift/release technique.

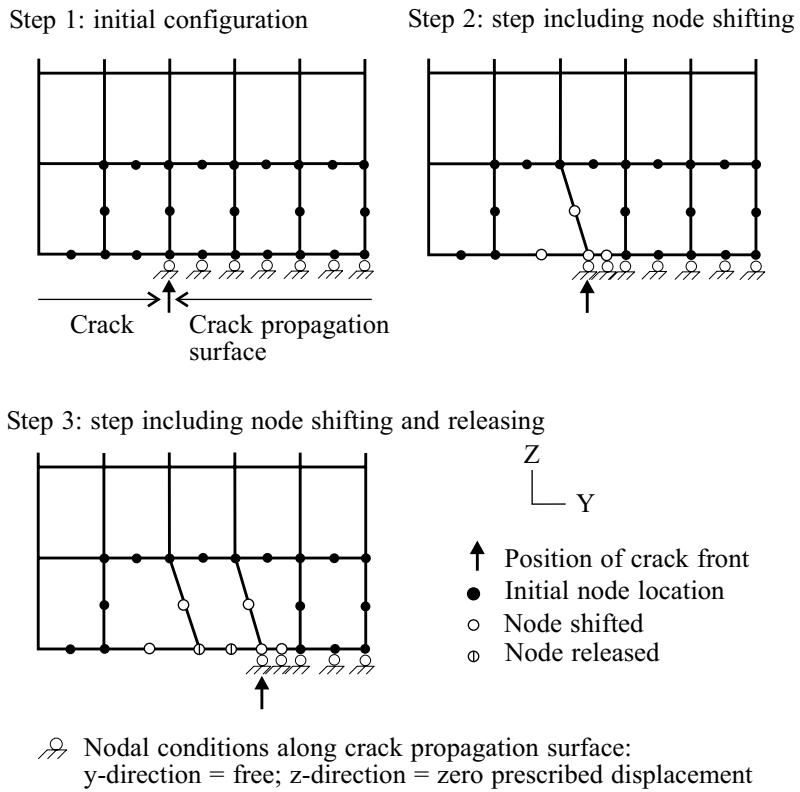


Figure 10.4-1: The node shifting and releasing procedure for crack propagation

Definition of the crack propagation surface: The crack propagation surface, which typically corresponds to the ligament in a fracture mechanics test specimen, is defined in ADINA as the set of nodes which may possibly be released when the crack opens (see Fig. 10.4-2). The crack propagation surface must be planar. In two-dimensional models, the crack propagation surface reduces to a line, though still called a surface.

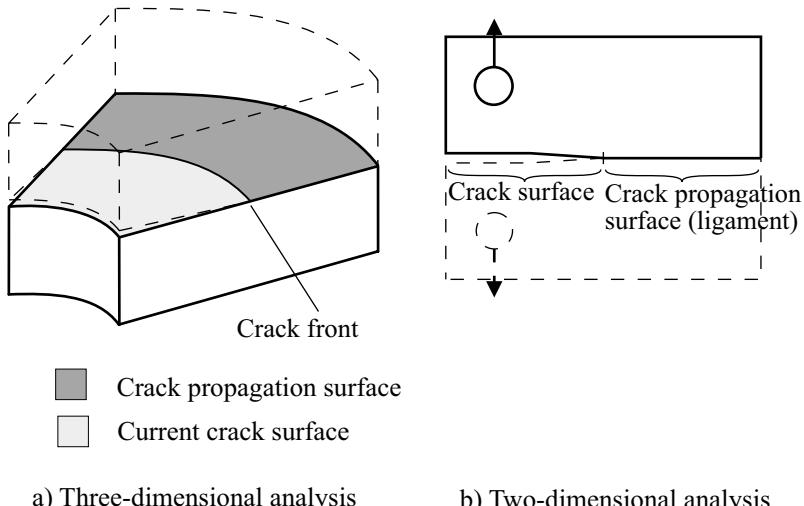


Figure 10.4-2: Definition of the crack propagation surface

The crack propagation surface must always lie either in the X-Y plane of the global Cartesian coordinate system or the a-b plane of a skew system common to all the nodes on the crack propagation surface.

Modeling of the crack advance: When the crack growth control parameter (energy release rate or any other parameter available in ADINA) indicates a crack increase, the crack tip node is automatically shifted to a new position (see Fig. 10.4-1 - step 2). When the advance of the crack tip node becomes larger than the width of the element ahead of the crack tip node in the propagation direction (and connected to the crack propagation surface), the crack tip node is released and the next node on the crack propagation surface starts being shifted (Fig. 10.4-1 - step 3). Midside nodes are shifted accordingly to remain halfway between the corresponding corner nodes, and are released when the preceding corner node on the crack propagation surface is released.

This procedure does not require any specific element size or mesh refinement, apart from general considerations on the accuracy of the analysis results.

- Quadrilateral elements should be used in the portion of the model including the crack tip and the crack propagation surface, in order to enable the automatic remeshing of the crack front as the crack grows.

Pressure loading: In case the faces of the crack are subjected to pressure loads, an automatic updating of the pressure loading is done on the new surfaces generated by the opening of the crack.

Crack growth control parameter: Different parameters can be used in ADINA for the control of the crack advance. Two categories of criteria are available:

- ▶ Displacement parameters such as the crack tip opening displacement (CTOD), the crack mouth opening displacement (CMOD), the load point displacement (see Fig. 10.4-3).
- ▶ Energy release rates with or without thermal, pressure, and dynamic corrections.
- If the energy release rate is chosen as the crack growth control parameter, the associated virtual material shift can be either a spatially fixed shift or a spatially moving shift (see Fig 10.4-4).
In the first case, the virtual shift domain is linked to the moving crack tip node. ADINA selects automatically the nodes which belong to the new virtual shift domain. The virtual shift must be defined by a number of rings of elements connected to the crack tip node.

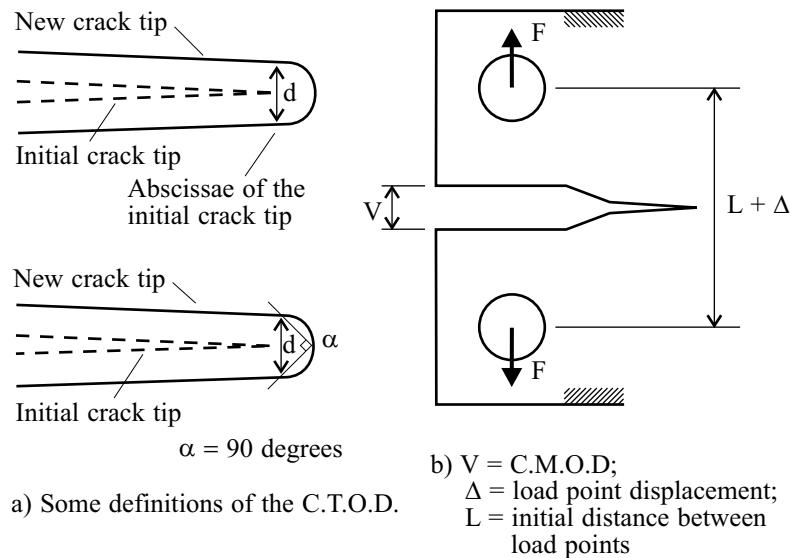


Figure 10.4-3: Definitions of the crack tip opening displacement, crack mouth opening displacement, load point displacement

In the second case, the virtual shift domain must be chosen large enough to include the successive crack tip locations at any time/load step of the analysis.

Crack resistance curve: In order to evaluate the amount of crack advance from the value of the energy release rate (or any other crack growth control parameter) a resistance curve must be input. This curve is a material property.

When the crack growth control parameter is a displacement parameter, then this curve relates the value of the displacement parameter to the crack advance. When the crack growth control parameter is an energy release rate, then this curve relates the value of the energy release rate to the crack advance.

For materials with temperature dependent material properties, a set of temperature dependent resistance curves can be input. The temperature at the crack tip node is then used with the value of the energy release rate (or the crack growth control parameter) to evaluate by linear interpolation the crack increment from the resistance curve (see Fig. 10.4-5).

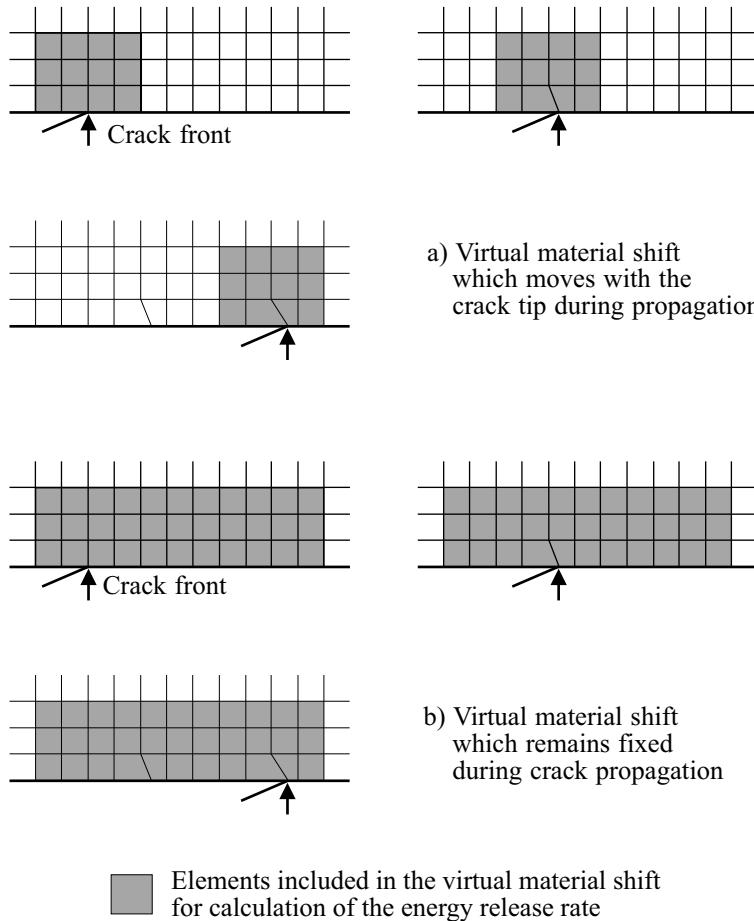


Figure 10.4-4: Usage of virtual material shift in crack propagation analysis

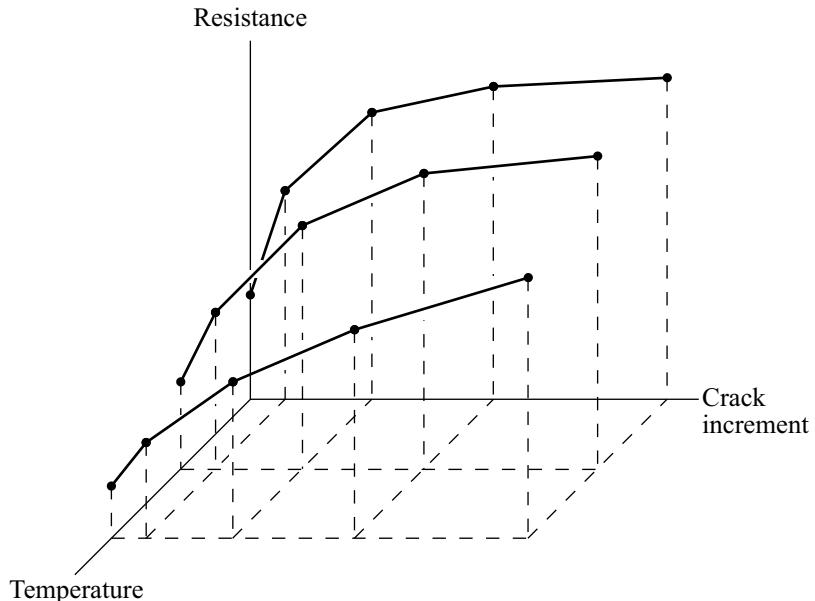


Figure 10.4-5: Resistance curves

10.5 Recommendations on the use of elements and meshes

- In linear elastic analysis, the variation of the stress and strain fields in the proximity of the crack tip is characterized by a $\frac{1}{\sqrt{r}}$ singularity where r is the distance to the crack tip location. Such a variation can be modeled using triangular elements with collapsed nodes at the crack tip location, and 1/4 point midside nodes. Wedge elements with 1/4 point midside and internal nodes can be used for the analysis of three-dimensional elastic bodies (see Fig. 10.5-1).
- For the analysis of elastic-plastic structures, triangular elements with 3 coincident but untied nodes at the crack tip location are recommended, in order to reproduce a $1/r$ variation of the stress/strain field. In 3-D analysis, wedge elements with coincident but untied nodes on the crack front will also improve the modeling of the stress/strain singularity (see Fig. 10.5-1).

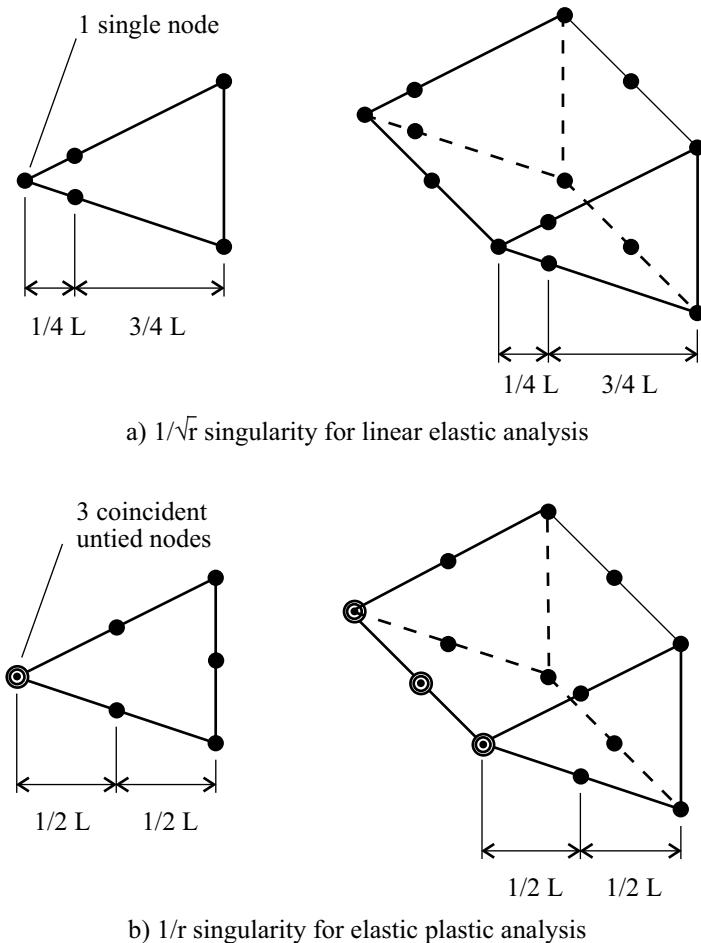


Figure 10.5-1: Element configurations for the modeling of stress and strain field singularities

- Typical meshes for the evaluation of crack tip blunting, or for the calculation of local crack opening parameters such as the CTOD, are presented in Fig. 10.5-2. They can easily be generated using the specific meshing features of the AUI.

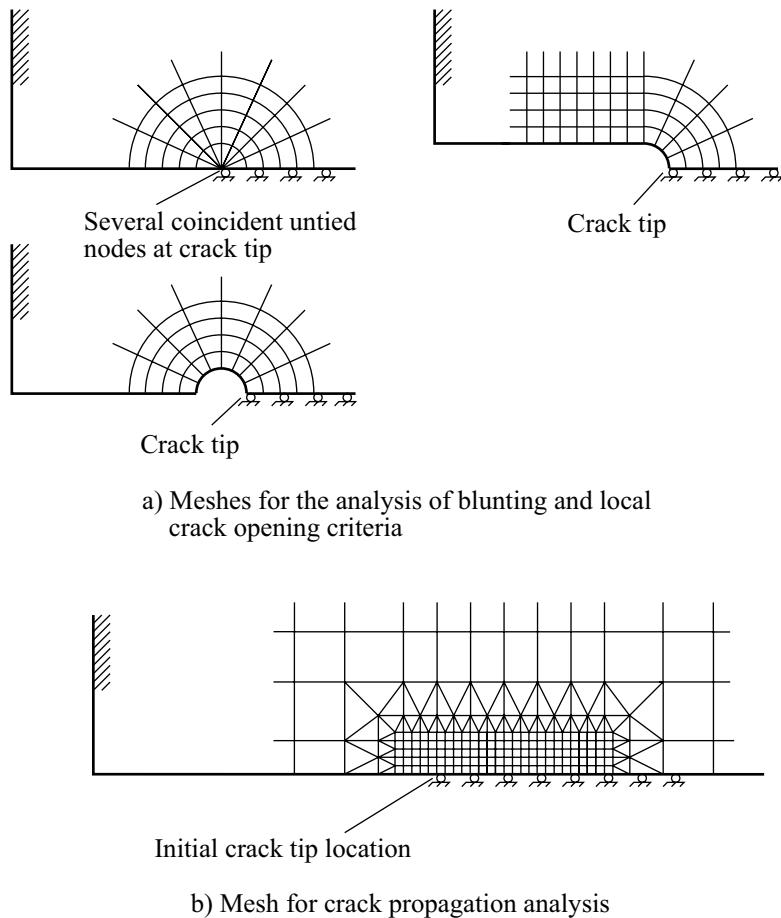


Figure 10.5-2: Meshes used in fracture mechanics

- A typical mesh for 2-D crack propagation analysis is also shown in Fig. 10.5-2. Quadrilateral elements with any number of nodes can be used. To ensure good results, the mesh in the area where the crack propagates should be made of a regular density of elements with regular shapes.
- Transition meshes should be used to connect the portion of refined mesh at the crack tip zone to the mesh used for the rest of the structure.

10.6 User-supplied fracture mechanics

10.6.1 General considerations

- The user interface for fracture mechanics analysis gives you the ability to implement new conservation criteria and new crack propagation models for two- and three-dimensional analysis.
- The user interface can be used in conjunction with the MNO, TL or ULH formulations.
- Any material model can be used in the user-supplied fracture mechanics calculations, including the user-supplied material model.
- The user-supplied coding for fracture mechanics must be inserted in subroutine UFR2 for 2-D analysis, and in subroutine UFR3 for 3-D analysis. These subroutines can be found in files ovl30u.f and ovl40u.f, respectively.

10.6.2 Usage of the user-supplied fracture mechanics subroutines

- Subroutines UFR2 and UFR3 can be used to read, generate, and write input data, to calculate conservation criteria and crack advances at any time/load step using the nodal and element results, to save history dependent quantities for later use, and to print and save the fracture mechanics results. These different operations are controlled by the integer label KEY.

KEY=1: Initial call at the beginning of the analysis. Master control data for fracture mechanics can be read, generated, written, and stored for later use. Note that the variable length arrays VUS1, VUS2 and VUS3 are automatically initialized to zero by ADINA prior to this call, but that your coding can initialize them differently at this stage.

KEY=2: One call for each element during the input data phase. Element data can be read, generated, and stored for later use.

KEY=3: One call at the beginning of every time/load step. Can be used to initialize temporary variables.

KEY=4 (linear analysis) or **KEY = 5** (nonlinear analysis): One call every time/load step, for each integration point of each element. The quantities available (displacements, stresses, strains, ...) correspond to the current step.

KEY = 6: (2-D analysis only) One call every time/load step. The key enables your coding to calculate a crack advance from the standard J-integral result. Only valid for combined user-supplied/standard fracture mechanics.

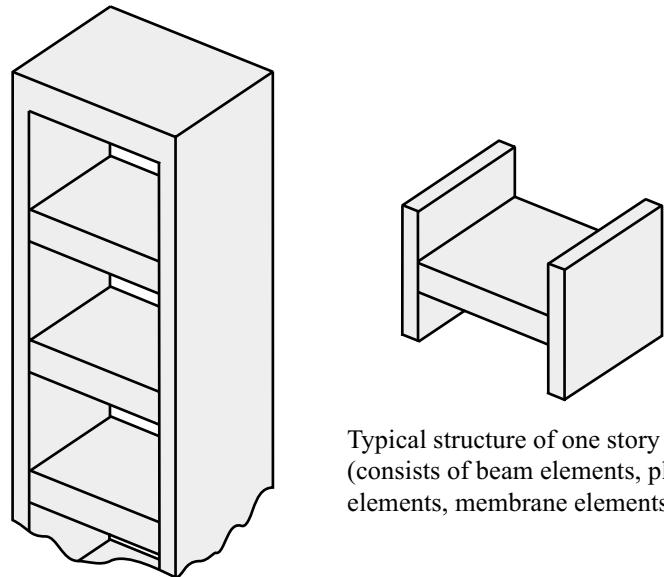
- For additional descriptions of the parameters passed to subroutines UFR2 and UFR3, please contact ADINA R&D, Inc.

11. Additional capabilities

11.1 Substructuring

11.1.1 Substructuring in linear analysis

- As an alternative to assembling and directly solving the complete system in linear static analysis, substructuring can be employed. Through the use of substructuring, in essence, large new finite elements are established that are then assembled in the same way as finite elements. Figure 11.1-1 illustrates an application of substructuring. The computational effort saved through substructuring can be significant. Also, the time required for the preparation of the input data for a large problem may be significantly less when using substructuring.



Structure considered

Typical structure of one story
(consists of beam elements, plate
elements, membrane elements, etc...)

Figure 11.1-1: Substructuring in the analysis of a multistory building

- Each substructure can be reused one or more times; each use of the substructure is called a "reuse". In Figure 11.1-1, one substructure is defined, corresponding to one story of the building. The substructure is reused for each story.
- The nodes of a substructure are grouped into two categories, condensed and retained. A condensed node is one that is not connected to any nodes in the main structure, a retained node is connected to at least one node in the main structure.
- Substructuring in dynamic analysis can be employed when the trapezoidal rule of time integration is used in a direct step-by-step solution (see Section 7.3).
- Substructuring can be used in two different ways:

ref. KJB
Section 8.2.4

- If the mass is lumped to a relatively few degrees of freedom, the massless degrees of freedom can be statically condensed out by substructuring. Figure 11.1-2 illustrates schematically the analysis approach taken. When selecting this approach, you must have good experience with the physical problem to be solved in order to be able to assign masses to only a few degrees of freedom, and yet obtain an accurate solution to the physical problem considered.
- Substructuring in dynamic analysis using ADINA can also be employed without "lumping masses" to certain degrees of freedom. In this case all degrees of freedom are treated as mass degrees of freedom (corresponding to a lumped mass matrix), and exactly the same solution is obtained as when no substructuring is employed. The only difference in a solution without substructuring is the cost of the numerical solution.

ref. K.J. Bathe and S. Gracewski, "On Nonlinear Dynamic Analysis Using Substructuring and Mode Superposition," *J. Computers and Structures*, Vol. 13, pp. 699-707, 1981.

This solution scheme can be very effective when large systems need to be solved. Note again that in this case no error is introduced in the solution using the substructuring.

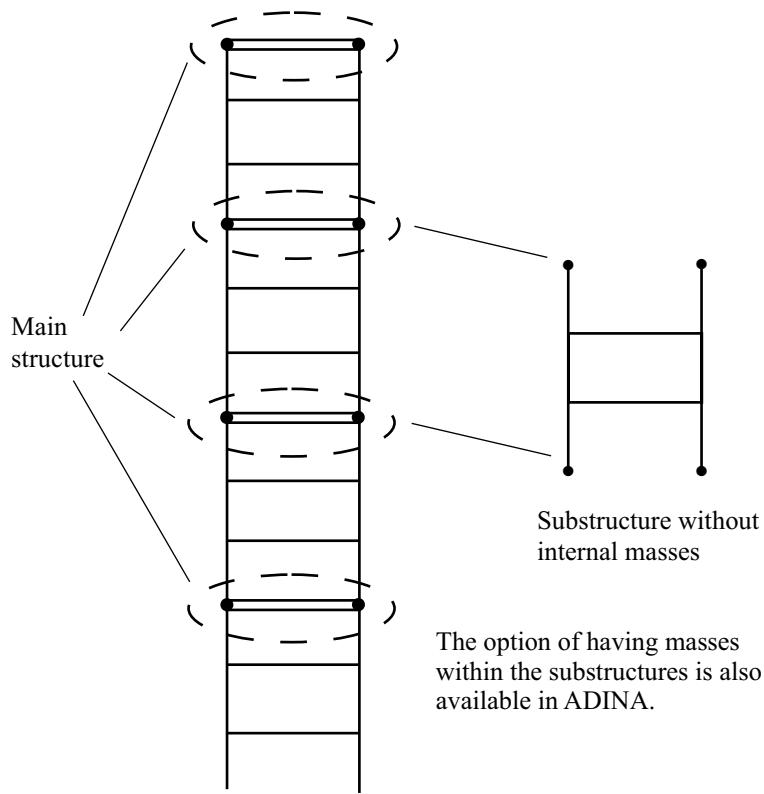


Figure 11.1-2: Substructuring with no masses internal to substructures

11.1.2 Substructuring in nonlinear analysis

- The remarks made regarding substructuring in linear analysis are also applicable when considering the nonlinear analysis of a structural system.
- An important additional consideration (beyond those for linear analysis) is that it can be particularly effective to perform substructuring in a nonlinear analysis when there are only local nonlinearities. In such case, the linear degrees of freedom can be allocated to the linear substructures and the nonlinear degrees of freedom to the master structure. This way the equilibrium iteration is performed in an effective manner.

Some applications are presented in the following reference.

ref. K.J. Bathe and S. Gracewski, "On Nonlinear Dynamic Analysis Using Substructuring and Mode Superposition," *J. Computers and Structures*, Vol. 13, pp. 699-707, 1981.

11.1.3 Substructuring restrictions

The following restrictions apply to substructures:

- Cyclic symmetric analysis cannot be performed if substructures are present.
- Nonlinear element groups cannot be employed by the substructure, i.e. the material model must be linear, the displacements must be small and the element birth/death option must not be activated. The main structure can, however, consist of both linear and nonlinear element groups.
- Contact surfaces cannot be formed by nodes belonging to a substructure.
- Any midsurface director vector sets that are manually defined must be defined separately for each substructure. Skew systems cannot, however, be defined separately for each substructure since the skew systems defined for the main structure can also be referenced by any of the substructures.
- When substructures are used, constraint equations cannot be employed by any substructure or between substructures and the main structure. Hence, constraint equations can only be employed within the main structure.
- Potential-based fluid element groups cannot be employed by the substructure.
- Substructure loads can only consist of concentrated loads, element pressure loads, prescribed displacements and mass proportional loading. Thus, centrifugal, electromagnetic, temperature loading and pipe internal pressure loading cannot be

employed by a substructure.

- Frequency analysis (and consequently mode superposition analysis, response spectrum analysis, harmonic or random vibration analysis) cannot be carried out when substructures are present.
- Buckling eigenvalues and mode shapes cannot be calculated when substructures are present.
- Substructures cannot be used in explicit dynamic analysis.
- Coupling nodes of a substructure must have the same relative locations and the nodal degrees of freedom must have the same directions as for the corresponding main structure nodes. If skew systems or director vectors are used for the coupling nodes, then care must be exercised so that the corresponding skew systems or director vectors for the main structure nodes result in the same directions for the nodal degrees of freedom.
- Skew systems are not allowed at condensed substructure nodes.
- Substructures must be used with a lumped mass assumption.

11.2 Cyclic symmetry analysis

- Cyclic symmetry analysis is useful for components when the geometry and boundary conditions are rotationally symmetric. The complete structure must be composed of N parts obtained by rotating a fundamental part about an axis of cyclic symmetry. Figure 11.2-1 shows a schematic example.
- Note that there is no approximation in the analysis when the cyclic symmetry option is used: the same result is obtained as if the complete structure is modeled.

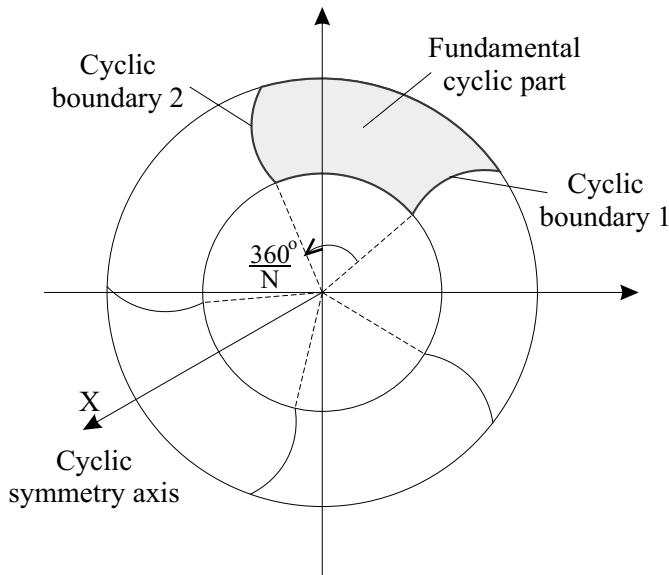


Figure 11.2-1: Schematic of a cyclically symmetric component

- Although the geometry, material properties and displacement boundary conditions must be cyclically symmetric, the external applied loads can be arbitrary, including pressure loads, concentrated loads, centrifugal and mass proportional loading, etc. Prescribed displacement degrees of freedom have to be the same on all cyclic parts. However, the amplitude of the prescribed displacement can vary from one part to the next (see Figure 11.2-2).
- If the loads are also cyclically symmetric, the model also possesses *periodic symmetry*. This assumption further simplifies the analysis (see Figure 11.2-2).
- For general cyclic symmetry, linear static, linear dynamic, and frequency analysis can be performed. For dynamic analysis, mode superposition or implicit time integration can be used.

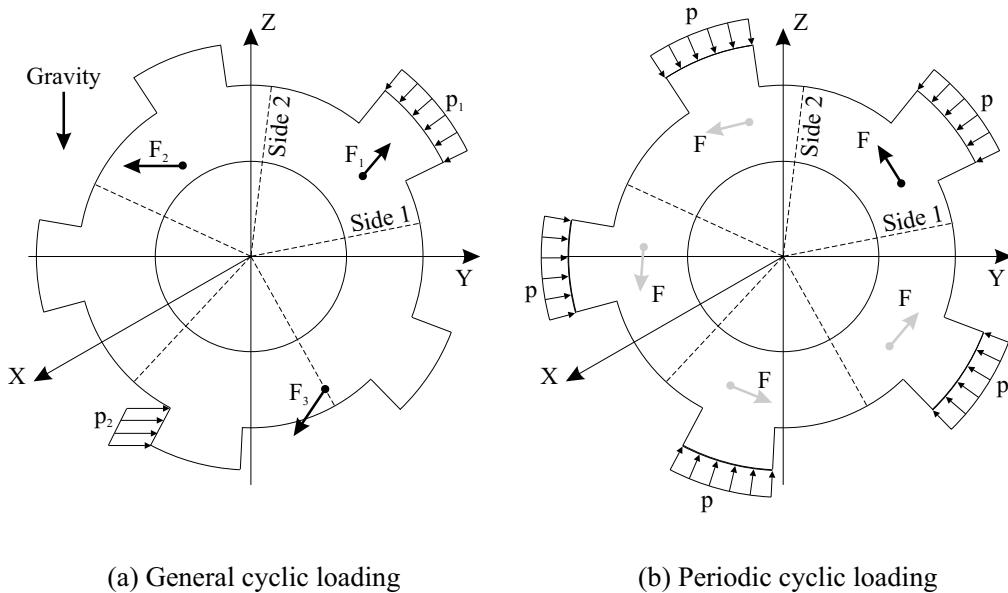


Figure 11.2-2: Applied loads in cyclic symmetry analysis

- For periodic symmetry, a nonlinear analysis can be performed as well (including large deformation, large strains, contact, etc.). A periodic frequency analysis only obtains the periodic and natural frequencies. A dynamic analysis with explicit time integration can also be performed for periodic symmetry.
- A nonlinear periodic static or dynamic analysis can be followed by a full cyclic symmetry frequency analysis to obtain all the natural frequencies and node shapes of the deformed structure.
- The cyclic symmetric analysis option has three advantages:
 - ▶ Only a small part of the complete structure needs to be modeled.
 - ▶ For general cyclic symmetry, the storage required is reduced up to a factor of $N/2$, where N is the number of cyclic parts. For periodic symmetry, the storage required is reduced up to a factor of N .

- ▶ The solution time can be significantly less than when the complete structure is modeled. As an estimate, the cyclic symmetric analysis will reduce the solution time by a factor of up to $N/2$ (N for periodic symmetry).
- Only the finite element model of the fundamental cyclic part should be created with the AUI. The displacement boundary conditions are only defined on the fundamental cyclic. Note that the displacements on the lines and/or surfaces connecting the fundamental part to the adjacent cyclic parts (the "cyclic boundaries") are unknown except for those corresponding to external rigid supports, and are free degrees of freedom.
- The nodes of cyclic boundary 1 must coincide with those of cyclic boundary 2 when rotated counterclockwise $360/N$ degrees about the axis of cyclic symmetry.
- The axis of cyclic symmetry can have any spatial orientation.
- Note that if structural elements (truss, beam, iso-beam, plate, shell or pipe elements) are defined exactly on the cyclic boundaries, then the complete structural elements must be input in the definition of the fundamental cyclic part.
- The loads on each of the N cyclic parts must be defined. Loads on a cyclic boundary can be input as loads onto either of the adjacent cyclic parts.
- In dynamic analysis, the initial conditions for the N cyclic parts must be defined.
- The analysis is performed using the theory of cyclic groups in which N analyses are performed using the finite element model of the fundamental cyclic part and different boundary conditions along the cyclic boundaries corresponding to the different load cases. The solutions are then superimposed to obtain the result for the whole structure. For more details about the theory, see

ref. W. Zhong and C. Qiu, "Analysis of Symmetric or Partially Symmetric Structures," *Computer Methods in*

Applied Mechanics and Engineering, Vol. 38, pp. 1-18,
1983.

- The following analysis options are not possible using the cyclic symmetric solution procedure:
 - ▶ Potential-based fluid elements
 - ▶ Piping analysis with ovalization/warping degrees of freedom
 - ▶ Substructure analysis
 - ▶ Response spectrum analysis
 - ▶ Harmonic or random vibration analysis
 - ▶ Fracture mechanics analysis
 - ▶ Explicit time integration

11.3 Reactions calculation

- Computations of the reaction forces and moments can be requested for all degrees of freedom where fixity boundary conditions are applied and where displacements are prescribed. No reactions are calculated for the degrees of freedom corresponding to deleted master boundary conditions.
Reaction calculations are on by default.
- Note that applied loads applied to the fixed degrees of freedom do not contribute to the displacement and stress solutions. However, these loads are accounted for in the reaction calculations.
- Reaction forces and moments at a node are computed using the consistent force vectors (calculated from the element internal stresses) of elements attached to the node. Hence, a check on the balance of the support reactions and the applied loads often provides a good measure on the accuracy of the solution (in terms of satisfying equilibrium in a nonlinear analysis).

- Reaction calculations in dynamic analysis with consistent mass matrix and direct time integration take into account the mass coupling to the deleted degrees of freedom. The reactions exactly equilibrate the applied forces in all cases.
- Reaction calculations in dynamic mode superposition analysis do not take mass coupling to the deleted degrees of freedom into account.
- Reaction calculations in dynamic analysis do not include contributions from the damping matrix.
- Reaction calculations are not available when the Wilson theta method of time integration is used in dynamic analysis.

11.4 Element birth and death option

- The element birth and death option is available in ADINA to be able to model processes during which material is added to and/or removed from the structure. Such process is, for example, encountered in the construction of a structure (structural members are added in succession), the repair of a structure (structural components are removed and new ones are added) or during the excavation of geological materials (a tunnel is excavated). If the element birth and death option is used, the corresponding element groups become automatically nonlinear. Figure 11.4-1 illustrates some analyses using the element birth and death option.
- The main feature of the element birth and death option is as follows:
 - ▶ If the element birth option is used, the element is added to the total system of finite elements at the time of birth and all times thereafter.
 - ▶ If the element death option is used, the element is taken out of the total system of finite elements at times larger than the time of death.
 - ▶ If the element birth-then-death option is used, the element is

added to the total system of finite elements at the time of birth and remains active until the time of death. The time of death must be greater than the time of birth. The element is taken out of the total system of finite elements at all times larger than the time of death.

- Figure 11.4-2 shows the physical removal and addition of elements along with the algebraic consequences. Note that once a finite element is born, the element mass matrix, stiffness matrix and force vector are for all solution times (before the time of death if the birth-then-death option is used) larger or equal to the time of birth added to the mass matrix, stiffness matrix and force vector of the total element assemblage. Similarly, when an element has died, the element mass matrix, stiffness matrix and force vector are not added to the total assembled mass matrix, stiffness matrix and force vectors for all solution times larger than the time of death of the element.
- Note that an element is born stress free. Hence, even if the nodal points to which the new element is connected have already displaced at the time of birth, these displacements do not cause any stresses in the element, and the stress-free configuration is defined to occur at the nearest solution time less than or equal to the one at which the element is born.

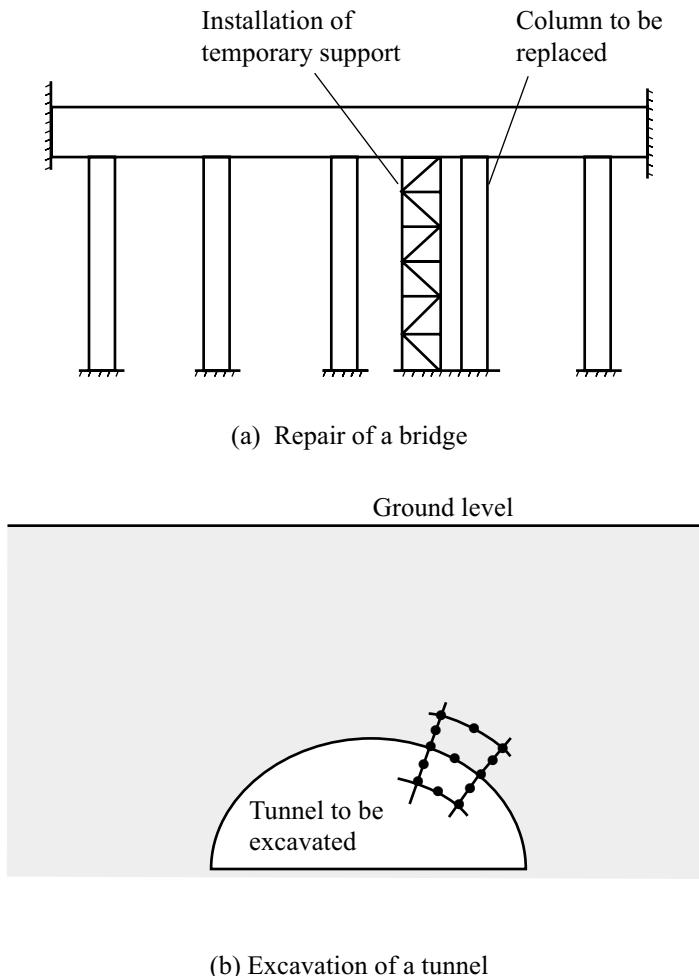
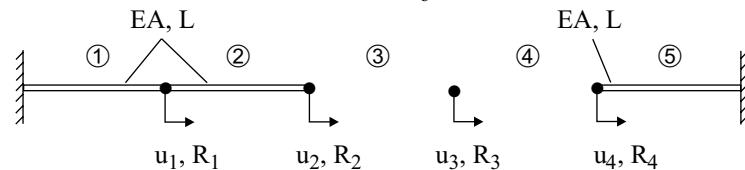


Figure 11.4-1: Sample analyses using the element birth and death options

- Figure 11.4-2 shows that when the element birth/death option is used, the tangent stiffness matrix may at some solution times contain zero rows and corresponding columns. When the element birth/death option is used, the equation solver disregards any zero diagonal element in the tangent stiffness matrix. Hence it is important that these zero diagonal elements are only due to element birth/death.

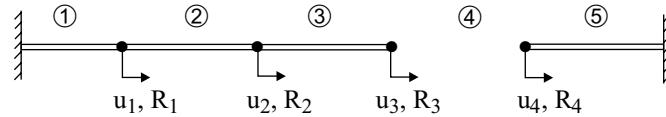
Matrices used to obtain solution at time t_b :



$$t_b - \Delta t \mathbf{K} = \frac{EA}{L} \begin{bmatrix} 2 & -1 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$t_b \mathbf{F}(0) = \begin{bmatrix} X \\ X \\ 0 \\ X \end{bmatrix} \quad \text{Non-zero}$$

Matrices used to obtain solution at time $t_b + \Delta t$:



$$t_b \mathbf{K} = \frac{EA}{L} \begin{bmatrix} 2 & -1 & 0 & 0 \\ -1 & 2 & 1 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$t_b + \Delta t \mathbf{F}(0) = \begin{bmatrix} X \\ X \\ 0 \\ X \end{bmatrix} \quad \text{El ③ is born stress free}$$

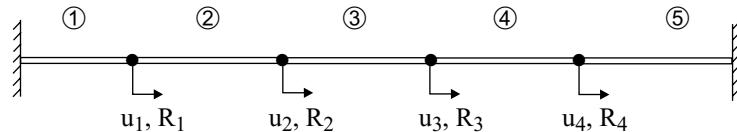
u_i =displacement; R_i =forces Time at birth: $TBIRTH = t_b + (\frac{\Delta t}{1000})$

(a) Element birth option

Figure 11.4-2: Evaluation of element matrices with element birth and death option active

As an example, Fig. 11.4-2(a) shows that, disregarding the one zero row and column, the stiffness matrix at time t corresponds to a stable truss structure.

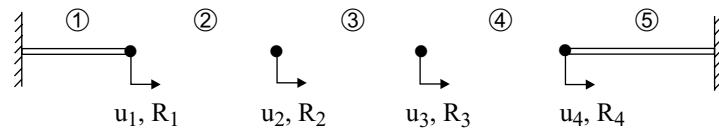
Matrices used to obtain solution at time $t_d - \Delta t$:



$$t_d - 2\Delta t \mathbf{K} = \frac{EA}{L} \begin{bmatrix} 2 & -1 & 0 & 0 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & 2 \end{bmatrix} \quad t_d - \Delta t \mathbf{F}^{(i)} = \begin{bmatrix} X \\ X \\ X \\ X \end{bmatrix}$$

EA, L for all elements

Matrices used to obtain solution at time t_d :



$$t_d - \Delta t \mathbf{K} = \frac{EA}{L} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad t_d \mathbf{F}^{(i)} = \begin{bmatrix} X \\ 0 \\ 0 \\ X \end{bmatrix} \text{ Due to death of elements } ②, ③ \text{ and } ④$$

$$\text{Time of death: TDEATH} = t_d - \left(\frac{\Delta t}{1000} \right)$$

(b) Element death option

Figure 11.4-2: (continued)

- ADINA includes an option for the element stiffness to be gradually reduced to zero using the element birth/death feature. A reduction factor multiplying the element stiffness is applied at the element death time. This factor decreases linearly from 1 to 0 between the element death time and a vanishing time. The vanishing time is obtained as the sum of the death time and a user-input decay time.

- The element birth/death option applies to any mass effect i.e., gravity loading, centrifugal loading and inertia forces. The mass matrix, therefore, does not remain constant throughout the solution.
- To provide the appropriate concentrated and element loading that takes into account the element birth/death option, you have to specify time functions on the loading that correspond to the proper elements (however, see also Section 11.5).
- The time at which an element becomes active or inactive is specified by the parameters TBIRTH and TDEATH respectively. If an element is required to be born at time t_b , i.e., the configuration of the element at time t_b corresponds to the stress-free

configuration, you should input $TBIRTH = t_b + \varepsilon$ where $\varepsilon = \frac{\Delta t}{1000}$

and Δt is the time step between time t_b and the next solution time. If an element is required to be inactive at and after time t_d , you

should input $TDEATH = t_d - \varepsilon$, where $\varepsilon = \frac{\Delta t}{1000}$ and Δt is the

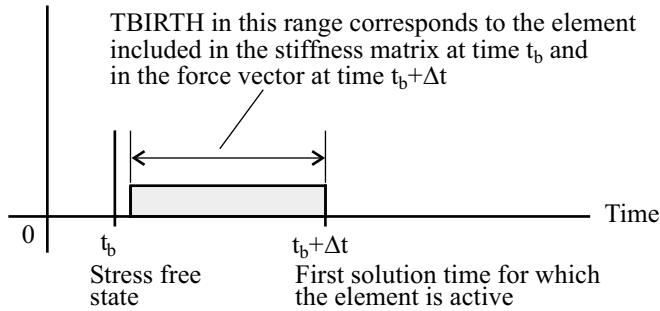
time step between the previous solution time and time t_d .

Birth option active: Figure 11.4-3(a) shows the activity of an element for which the birth option is active. Note that if $TBIRTH$ is input for the range shown (where $TBIRTH > t_b$ and $TBIRTH \leq t_b + \Delta t$), then the stress-free configuration of the element is at time t_b , and the element is first active at time $t_b + \Delta t$. Note that the stress in the element is based on the displacements measured from the configuration at time t_b , irrespective of the input value of $TBIRTH$, provided that $TBIRTH$ is input in the range shown in Fig. 11.4-3(a). See also the example given below.

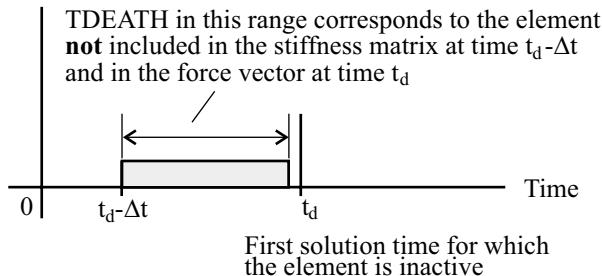
Death option active: Figure 11.4-3(b) shows the activity of an element for which the death option is active. Note that if $TDEATH$ is input for the range shown (where $TDEATH \geq t_d - \Delta t$ and $TDEATH < t_d$), then the element is first inactive at time t_d .

Birth-then-death option active: This is a direct combination of the birth and death options. Initially some elements in an element group are inactive. At a particular solution time determined by the

time of birth TBIRTH, the elements become active and remain so until a subsequent solution time determined by the time of death TDEATH, where $TDEATH > TBIRTH$.



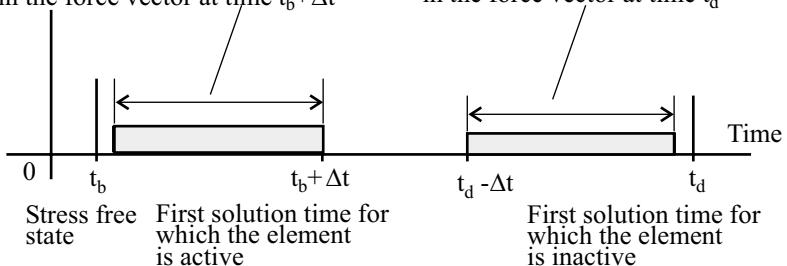
(a) Birth option active



(b) Death option active

TBIRTH in this range corresponds to the element included in the stiffness matrix at time t_b and in the force vector at time $t_b + \Delta t$

TDEATH in this range corresponds to the element **not** included in the stiffness matrix at time $t_d - \Delta t$ and in the force vector at time t_d



(c) Birth and death options active

Figure 11.4-3: Use of element birth and death option

Figure 11.4-3(c) shows the activity of an element for which the element birth and death option is active. Note that if TBIRTH is input for the range shown (where $TBIRTH > t_b$ and $TBIRTH \leq t_b + \Delta t$), the stress-free configuration of the element is the configuration at time t_b ; the element is first active at time $t_b + \Delta t$.

If TDEATH is input for the range shown (where $TDEATH \geq t_d - \Delta t$, and $TDEATH < t_d$, $t_d > t_b$), then the element, after birth, is first inactive at time t_d .

Example of the element birth option: Consider the materially linear truss element model shown in Fig. 11.4-4(a) in which the time of birth for element 2 is . At time t , the length ' L ' corresponding to the load ' R ' is determined as shown in Fig. 11.4-4(b). Note that ' L ' corresponds to the length at which element 2 is stress-free.

At the time of birth, element 2 with length ' L ' is added to the system which was already in equilibrium, see Fig. 11.4-4(c). Note that the force in element 2 is exactly zero after its addition to the system.

At time $t + \Delta t$, element 2 is now active. The force in element 2 is determined based on its deformation with respect to the stress-free state, see Fig. 11.4-4(d). Hence, the total increment in displacement from time t to time $t + \Delta t$ determines the force in the truss. Identically, the same solution would be obtained using any value for TBIRTH which satisfies the relation $TBIRTH > t$ and $TBIRTH \leq t + \Delta t$.

- Birth/death is available for contact pairs in contact analysis.

11.5 Element "death upon rupture"

- For the materials and elements listed in Table 11.5-1, element death is automatically activated when rupture is detected at any integration point of the element. The element is then considered as "dead" for the remainder of the analysis, and, in essence, removed from the model (mass and stiffness contributions).
- As elements become dead, contactor segments connected to these elements are also removed from the model.

- Load penetration can be activated to ensure that pressure loads are properly redistributed inside voids created by "dead" elements (see Section 5.3.1).

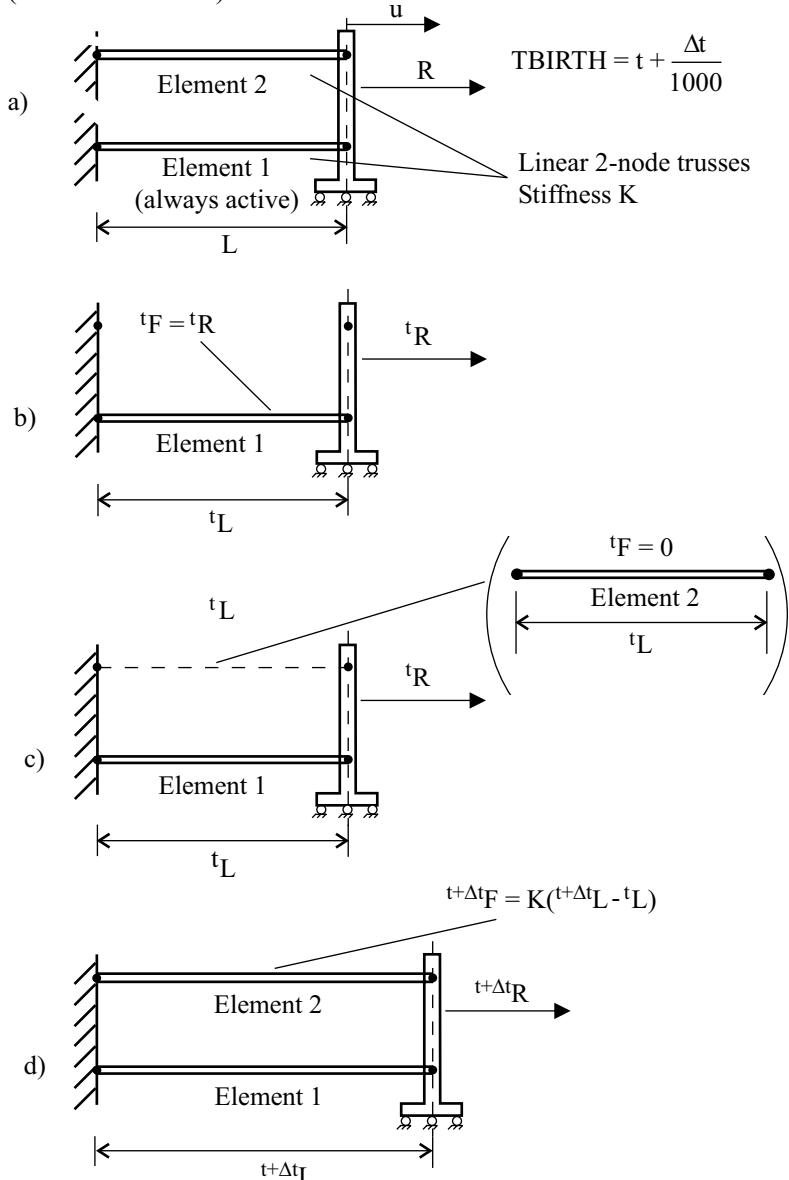


Figure 11.4-4: Example on the use of the element birth option

Table 11.5-1: Elements and material models that include "death upon rupture"

	2-D solid	3-D solid	Beam	Iso- beam	Shell	Pipe
Plastic-bilinear, plastic-multilinear	✓	✓	-	✓	✓	✓
Thermo-plastic, creep, plastic-creep, multilinear-plastic-creep, creep-variable, plastic-creep-variable, multilinear- plastic-creep-variable	✓	✓	-	✓	✓	✓
Plastic-orthotropic	✓	✓	-	-	✓	-
Moment-curvature	-	-	✓	-	-	-

11.6 Initial conditions

11.6.1 Initial displacements, velocities and accelerations

- You can specify the initial displacements, velocities and accelerations of the nodes. Any initial displacements, velocities and accelerations specified in a restart run are ignored.

11.6.2 Initial temperatures and temperature gradients

- The formulas used to determine the initial nodal temperatures are as follows:

Initial temperature explicitly specified: ${}^0\theta_i = {}^0\theta_i^E$

Initial temperature not explicitly specified: ${}^0\theta_i = \text{TINIT}$

in which ${}^0\theta_i$ is the initial nodal temperature at node i , TINIT is the overall initial temperature (also referred to as the default initial temperature) and ${}^0\theta_i^E$ is the explicitly specified initial temperature at the node.

Initial temperatures can also be read from a temperature file. In this case, any directly specified initial temperatures are added to those read from the temperature file.

Initial temperature gradients are handled in an analogous manner. These are only used at axisymmetric shell nodes and at shell mid-surface nodes in order to calculate initial temperatures in these elements.

Initial temperatures and temperature gradients specified in a restart run are ignored.

- When temperatures are directly prescribed (as in Section 5.6), the directly prescribed temperatures are not used to determine the initial temperatures. Initial temperatures must be specified as described above.
- The thermal strains are always assumed to be zero initially, see Section 3.5.

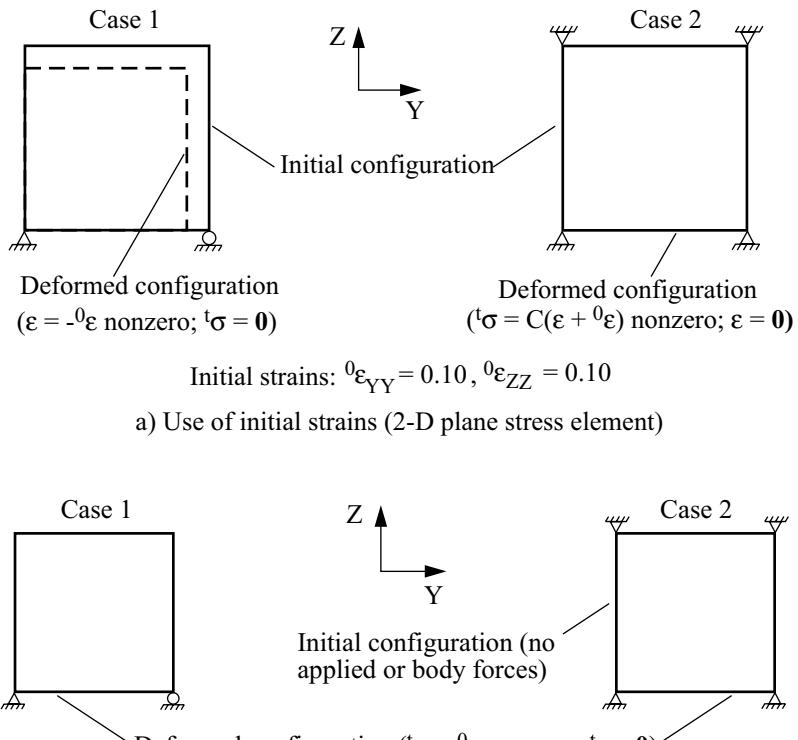
11.6.3 Initial pipe internal pressures

- The initial pipe internal pressures consist of the sum of the explicitly defined pipe internal pressure loads (evaluated at time TSTART) and the pipe internal pressures read from file, if any. If there are no explicitly defined pipe internal pressure loads and no pressure loads read from file, the initial pipe internal pressures are set to zero.

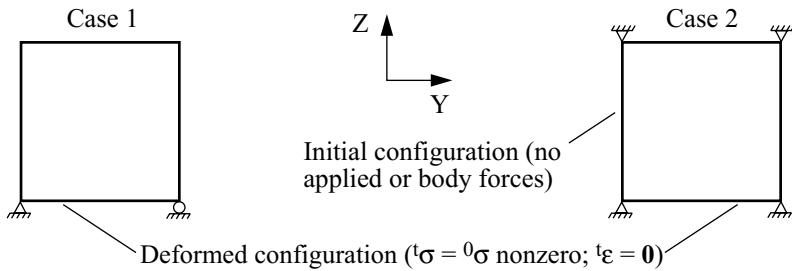
Initial pipe internal pressures specified in a restart run are ignored.

11.6.4 Initial strains

- The initial strain capability allows the definition of an initial strain field (see Fig. 11.6-1). The initial strains are added to the other strain components.
- The initial strain option is available for all solid and structural elements, but only for nonlinear element groups. If you apply the initial strain option to an element group that would otherwise be linear, the element group is automatically classified as nonlinear.
- All material models except the Ilyushin material model can be used.
- The initial strains can be provided as nodal point initial strains or element initial strains.



a) Use of initial strains (2-D plane stress element)



b) Use of initial stresses that do not cause deformations (2-D plane stress element)

Figure 11.6-1: Schematic use of initial strains and stresses

- All initial strain components input by nodal point strain components or element strain components are referred to the initial strain axes of the elements. These axes are denoted (1-2-3), so the initial strain components are e_{11}, e_{22} , etc.

The initial strain axes for the truss, beam, iso-beam, and pipe elements correspond to the axial directions of the elements and only axial initial strain (component 11) is considered.

The initial strain axes for the 2-D solid, 3-D solid, plate and shell elements are user-defined. For the 2-D solid, plate and shell elements, the 1-2 axes must lie in the plane of the element and the 3 axis lies out of the plane of the element.

One way to define initial strain axes in the AUI is using axes-systems.

- Note that nodal initial strains defined at a node are used differently by the elements connected to that node, because the initial strain axes for these elements can be different.
- A summary of initial strain input for different element types is shown in Table 11.6-1. Note that for the plate elements, there is a distinction between membrane and flexural initial strain components. Also for the shell element, membrane and gradient initial strain components can be input.
- For the truss, beam, iso-beam, plate, shell and pipe elements, the element initial strains are specified as element data (possibly assigned to the geometry in the AUI).

Table 11.6-1: Initial strains input for nodes and elements

Element	Nodal point strain components						Element strain components					
	e_{11}	e_{22}	e_{33}	γ_{12}	γ_{13}	γ_{23}	e_{11}	e_{22}	e_{33}	γ_{12}	γ_{13}	γ_{23}
Truss	✓	-	-	-	-	-	✓	-	-	-	-	-
2-D solid	✓	✓	✓ ²⁾	✓	-	-	✓	✓	✓ ²⁾	-	-	-
3-D solid	✓	✓	✓	✓	✓	✓	✓	✓	✓	-	-	-
Beam	✓	-	-	-	-	-	✓	-	-	-	-	-
Iso-beam	✓	-	✓ ²⁾	-	-	-	✓	-	✓ ²⁾	-	-	-
Plate: membrane, flexural	✓	✓	-	✓	-	-	✓	✓	-	✓	-	-
Shell: membrane, gradient	✓	✓	-	✓	✓	✓	✓	✓	-	✓	✓	✓
Pipe	✓	-	-	-	-	-	✓	-	-	-	-	-

- 1) The symbol "✓" means that the strain component can be defined for that element type, either by node or by element input.
- 2) Axisymmetric elements only

- For the 2-D and 3-D solid elements, the element initial strains are specified using strain field coefficients A, B, C, D, E, F and the

following formulas:

2-D solid elements:

$$e_{22} = A + Bz$$

$$e_{11} = Ce_{22} + D$$

$$e_{33} = Ee_{22} + F$$

3-D solid elements:

$$e_{33} = A + Bz$$

$$e_{11} = Ce_{33} + D$$

$$e_{22} = Ee_{33} + F$$

where z is the global z coordinate.

Initial strains in the updated Lagrangian Hencky formulation (large strains): The initial strains are assumed to be small, and the initial spin tensor is assumed to be equal to zero, so that

$${}^{t_0}_0 \mathbf{X} = \mathbf{I} + {}^{t_0} \boldsymbol{\epsilon}$$

in which ${}^{t_0} \boldsymbol{\epsilon}$ is the initial strain tensor and ${}^{t_0}_0 \mathbf{X}$ is the initial deformation gradient. Then a multiplicative decomposition of the deformation gradient is used at every step, i.e.:

$${}^t_0 \mathbf{X} = {}^t_{t_0} \mathbf{X} {}^{t_0}_0 \mathbf{X}$$

in which ${}^t_{t_0} \mathbf{X}$ is obtained from the displacements calculated at time t .

11.6.5 Initial stresses

- An initial stress option can be used with all element types. The following material models can be used: **elastic-isotropic**, **elastic-orthotropic**, **thermo-elastic**, **thermo-orthotropic**, **Drucker-Prager**, **Cam-clay**, **Mohr-Coulomb**.

The initial stress capability allows initial stresses corresponding to an unknown loading to be input. These stresses can be provided as nodal point initial stresses or element initial stresses.

Whenever the constitutive relation is used, the initial stresses are added to the current stresses caused by externally applied loading. The initial stresses are included in the printed and saved stresses.

- The following equations are used:

$${}^t\mathbf{F} = \int {}^t\mathbf{B}^T \left({}^t\boldsymbol{\tau} + \boldsymbol{\tau}^I \right) d{}^tV \quad (11.6-1)$$

in which $\boldsymbol{\tau}^I$ are the initial stresses (corresponding to the formulation/material model used, i.e. Cauchy or 2nd Piola-Kirchhoff), and ${}^t\boldsymbol{\tau}$ are calculated as usual from the mechanical strains. Equation (11.6-1) yields at time 0

$$\mathbf{R}^I = \mathbf{F}^I = \int {}^0\mathbf{B}^T \boldsymbol{\tau}^I d{}^0V$$

The step-by-step incrementation is then carried out using

$${}^{t+\Delta t}\mathbf{K}^{(i-1)}\Delta\mathbf{U}^{(i)} = {}^{t+\Delta t}\mathbf{R} - {}^{t+\Delta t}\mathbf{F}^{(i-1)} + \mathbf{R}^I$$

in which ${}^{t+\Delta t}\mathbf{R}$ is the externally applied load vector.

- Please note:
 - ▶ The stresses $\boldsymbol{\tau}^I$ are in the \mathbf{K}_{NL} matrix (stiffening effect)
 - ▶ The stresses $\boldsymbol{\tau}^I$ are in the material law.
 - ▶ The stresses $\boldsymbol{\tau}^I$ are effects that do not change in time.
- Note that either initial stresses or initial strains can be accepted by the program but can not be mixed.
- All initial stress components input by nodal point stress components or element stress components (see Table 11.6-2) are referred to the initial stress axes of the elements defined in the same

way as the initial strain axes.

Note that nodal initial stresses defined at a node can be used differently by the elements that are connected to that node, because the initial stress axes for these elements can be different.

- In linear or materially-nonlinear-only analysis, the initial stresses as stated above do not create deformations unless the element birth/death option is used (e.g., tunneling). It can be requested that initial stresses create deformations. In this case, the initial stress load vector \mathbf{R}^I is not used, but all other equations are applicable.
- Another option is that the \mathbf{R}^I vector shown above is not included in the equilibrium equations. In this case, the initial stresses cause deformations. This option can be used for linear, materially-nonlinear-only and large displacement/small strain analysis.

Restrictions

- The initial stress option can be used for the Drucker-Prager, Cam-clay and Mohr-Coulomb material models only if the effective stress corresponding to the initial stresses is smaller than the yield stress (that is, the initial stress state must lie within the initial yield surface).
- The initial stress option should not be used in large displacement/large strain analysis.

Table 11.6-2: Initial stress input for nodes and elements

Element	Nodal point stress components						Element stress components					
	τ_{11}^I	τ_{22}^I	τ_{33}^I	τ_{12}^I	τ_{13}^I	τ_{23}^I	τ_{11}^I	τ_{22}^I	τ_{33}^I	τ_{12}^I	τ_{13}^I	τ_{23}^I
Truss	✓	-	-	-	-	-	✓	-	-	-	-	-
2-D solid	✓	✓	✓ ²⁾	✓	-	-	✓	✓	✓ ²⁾	-	-	-
3-D solid	✓	✓	✓	✓	✓	✓	✓	✓	✓	-	-	-
Beam	✓	-	-	-	-	-	✓	-	-	-	-	-
Iso-beam	✓	-	✓ ²⁾	-	-	-	✓	-	✓ ²⁾	-	-	-
Plate: membrane, flexural	✓	✓	-	✓	-	-	✓	✓	-	✓	-	-
Shell: membrane, gradient	✓	✓	-	✓	✓	✓	✓	✓	✓	-	✓	✓
Pipe	✓	-	-	-	-	-	✓	-	-	-	-	-

- 1) The symbol "✓" means that the stress component can be defined for that element type, either by node or by element input.
 2) Axisymmetric elements only

11.7 Strain energy densities

- The computation of strain energy densities at integration points can be requested for 2-D solid and 3-D solid elements. The strain energy density is given by

$$W(\varepsilon_{ij}) = \int_0^{\varepsilon_{ij}} \sigma_{ij} d\varepsilon_{ij}$$

in which ε_{ij} are the total strain components and σ_{ij} are the stress components. The strain energy density includes participations from the elastic, inelastic, and thermal strains.

- The strain energy densities are available for all linear and nonlinear formulations. The strain energy densities are available for the displacement-based and mixed (u/p) element formulations.

11.8 Element group inertial properties

- The following element group properties can be requested in ADINA for printout/saving: **total mass, total volume, moments of inertia, products of inertia, centroid, center of mass**.
- In ADINA, the moments and products of inertia are calculated with respect to the origin of the global coordinate system.
- The AUI can combine the element group quantities over the whole model or over a selected zone in the model. The AUI also includes the concentrated masses in the calculations. The moments and products of inertia are calculated and printed with respect to the center of mass of the whole model or zone.
- It is not possible for the AUI to compute the above-mentioned properties for a selected part of an element group.

11.9 Restart option

- Restart is a useful feature in the ADINA System. It can be used when the user wishes to continue an analysis beyond its previous end point, or change the analysis type, loads or boundary conditions or tolerances. A restart analysis is set via the MASTER command.
- All relevant solution data needed for a restart run are saved in a file (extension .res) in case they are needed in a restart analysis. The frequency of data writing to a restart file is set via the MASTER command.
- Note that multiple restart data can be appended to the restart file. This enables the restart analysis to be based on a solution step different from the last converged solution. Saving multiple time step solutions to a restart file can be expensive, however, as it leads to a large restart file size.
- To perform a restart analysis, the .res file from the first model must be copied to the second model. If running in interactive

mode, the AUI will first look for the restart file, and if it is not found, the user will be prompted to locate the restart file.

- The geometry, and most element data cannot be changed in a restart analysis. However, the following changes are allowed:
 - ▶ Type of analysis can change. Static to dynamic and dynamic to static restarts are allowed.
 - ▶ Solution type can change. Implicit analysis to explicit analysis restarts are allowed and vice-versa. Some restrictions are imposed however when changing solution type:
 - Features not available in either solution type cannot be used.
 - Incompatible modes cannot be used.
 - ▶ Solution control variables can change. The flags, constants and tolerances for the iteration method, convergence, time integrations, automatic time stepping and load-displacement-control can be changed.
 - ▶ Externally applied loads and enforced displacements can be modified.
 - ▶ The material constants can be changed. However, note that in a restart run the same material model (with the same number of stress-strain points and the same number of temperature points, if applicable) must be used for each element as in the preceding run.
 - ▶ Boundary conditions can change.
 - ▶ Constraint equations and rigid elements can change.
 - ▶ Contact settings can change. This includes most contact set, contact pair and contact surface parameters. See Section 4.7.5 for restrictions.
 - ▶ Rayleigh damping coefficients can change.

- ▶ Time increment and number of solution steps can be modified.

- ▶ Time functions describing the load variations can be changed.

Restart in frequency analysis

- For large systems, if the subspace iteration method for frequency analysis does not converge, it can be efficient to use the converged eigenvectors as a subset of the starting vectors and redo the analysis with modified input. The advantage is that the amount of computational effort needed to compute the remaining eigenvectors is less than the amount of computational effort needed to compute all of the requested eigenvectors.
- In the following discussion, the first ADINA run is the one in which not all of the requested eigenvectors are obtained and the second ADINA run is the one in which the remaining requested eigenvectors are obtained. Note that the second ADINA run is not considered to be a restart run.

You supply the converged eigenvectors from the first ADINA run to the second ADINA run as user-supplied starting vectors. You also generate a new data file for the second ADINA run, in which you can change the number of starting vectors and the maximum number of iterations. It is recommended that the number of starting vectors be greater than or equal to the number of converged vectors from the first ADINA run.

Before starting the second ADINA run, you need to copy the eigenvectors from the first run to the second run. The eigenvectors are stored in a file with extension .stv.

11.10 Parallel processing

- ADINA supports parallel processing on the following UNIX platforms: Compaq, HP (operating system HP-UX 11 only), Linux, IBM, SGI and Sun. ADINA also supports parallel processing on PCs running Windows NT, 2000 or XP.
- The following coding is parallelized: assembly of the global system matrices (loop over element groups), the in-core sparse solver, the out-of-core sparse solver, and contact search and check for the rigid target contact algorithm. The parallelization of the assembly of the global system matrices is available only for the HP, Linux and SGI platforms. The parallelization of the in-core sparse solver, out-of-core sparse solver and the rigid target contact algorithm is available for all computers for which parallel processing is supported.
- On SGI computers you can select a scheduler type. The scheduler allocates iterations of a loop to the available processors. For the assembly of the global system matrices, the parallelized loop is the loop over all element subgroups and, within each iteration of this loop, ADINA processes an element subgroup. Let N be the number of subgroups, P be the number of processors and C be the “chunk” size (currently 1). Then the scheduler possibilities supported by ADINA are

Simple: Each processor executes N/P loop iterations. This method works well when N is small and the calculation effort is the same for all loop iterations. However, unless N is evenly divided by P , there will be a time at the end of the loop when fewer than P processors are working.

Dynamic: Each processor executes C loop iterations. When the processor is finished, the processor executes the next undone C loop iterations. This method works well when the calculation effort is different for the loop iterations.

Gss (guided self-scheduling): Each processor executes $N/2P$ loop iterations. When the processor is finished, the processor executes the next undone $N/4P$ loop iterations, and so on.

The default scheduler type is gss.

Note that the sparse solvers also are parallelized, hence the choice of scheduler affects the sparse solver.

11.11 Usage of memory and disk storage

- Depending on the size of the problem and the memory allocated to ADINA (MTOT), ADINA can perform the solution either in-core (entirely within virtual memory) or out-of-core (reading and writing disk files). Whenever possible the solution is performed in-core.
- Explicit dynamic analysis can only run in-core. Enough memory must be available.
- When the solution is performed in-core, the stiffness (and mass) matrices, as well as the element group information, is stored in memory. In this case, the parameter IOPTIM, which is printed in the ADINA output file, is set to 3.
- If MTOT is too small to keep the element group information in-core, ADINA will only keep the stiffness (and mass) matrices in-core, and the parameter IOPTIM is set to 2.
- If MTOT is too small to keep the stiffness (and mass) matrices in-core, the solution is performed out-of-core and the parameter IOPTIM is set to 1.
- When using the iterative or sparse solver, the solution is always performed in-core. The out-of-core solution procedure can only be used by the skyline solver (COLSOL) and it is not effective for practical engineering problems.
- The disk files that may become large are

Unit 1: Linear element group data

Unit 2: Nonlinear element group data

Unit 4: Linear stiffness matrix

Unit 7: Implicit dynamic analysis, effective stiffness matrix

Unit 8: Restart file

Unit 10: Nonlinear stiffness matrix (skyline solver only)

Unit 11: Mass/damping matrices

On UNIX computers, these disk files, and other temporary disk files, are stored in a subdirectory within the directory from which the problem is run. The subdirectory name starts with a “.”, so a directory listing with `ls` will not show the name.

- You can request that ADINA save intermediate disk files:

Factorized linear stiffness matrices: If substructures are present and/or the main structure is linear (only linear material models, only small displacements and strains and no element birth/death options or contact surfaces), the factorized linear stiffness matrices can be saved for use in restart jobs, so that in the restart jobs, the matrix assemblages and factorizations need not be performed. This results in improved solution efficiency.

If the main structure is linear, the record index array for unit 10, on which the **L** and **D** factors of the main structure are stored, is written to the restart file (unit 8). Unit 10 must be saved in addition to unit 8, and later supplied to the restart job.

If substructures are present, the **L** and **D** factors of all substructures are stored on unit 16 and must be saved in addition to unit 8 (and also unit 10 when applicable), and later supplied to the restart job.

The option of saving the factorized linear stiffness matrices is not allowed in an eigenvalue solution, or in explicit analysis.

Global stiffness and mass matrices: The global stiffness and mass matrices can be written to unit 70 in formatted form. The format is as follows:

```
      WRITE (70,1000) ((I,MAXA(I)),I=1,NEQ+1)
      WRITE (70,1001) ((I,A(I)),I=1,NELMA)
      IF ((the mass matrix is lumped)) THEN
          WRITE (70,1001) (I,D(I)),I=1,NEQ)
      ELSEIF ((the mass matrix is consistent)) THEN
          WRITE (70,1001) (I,B(I)),I=1,NELMA)
      ENDIF
      1000 FORMAT (1X,I8,5X,I8)
      1001 FORMAT (1X,I8,5X,E20.13)
```

where

A	=	global stiffness matrix
B	=	consistent mass matrix
D	=	lumped mass matrix
MAXA =		auxiliary array that stores the addresses of the diagonal entries of the stiffness (and consistent mass) matrices in A (and B)
NEQ =		total number of equations to be solved
NELMA	=	total number of matrix entries under the skyline

See Section 12.2.3 in reference KJB for an example of the storage of the stiffness matrix in skyline form.

In nonlinear analysis, you can select the time step at which the nonlinear stiffness matrix is written to unit 70. The stiffness matrix is stored at the beginning of the time step (e.g., if you select time step 1, the stiffness matrix corresponding to time step 0 is stored).

- In certain cases, you can specify data to ADINA using disk files:

Pipe internal pressures: Pipe internal pressures can be read from unit 24 in the following format (unformatted sequential Fortran file):

```
READ (24) TIME, (PIPEIP(I), I=1, NUMNP)
```

Here NUMNP is the number of nodes in the main structure and TIME is the solution time for the pipe internal pressures.

You must specify whether the times in unit 24 correspond exactly with the ADINA solution times, or whether they are different than the ADINA solution times. In the latter case, ADINA interpolates the pipe internal pressures to the ADINA solution times and writes the interpolated data to unit 24.

Temperatures: Temperatures can be read from unit 56 in a manner entirely analogous to the reading of pipe internal pressures. Temperatures read from unit 56 are added to the other specified temperatures (initial or prescribed), except at the beginning of a restart run.

Temperature gradients: Temperature gradients can be read from unit 57 in a manner entirely analogous to the reading of temperatures. Temperature gradients read from unit 57 are added to the other specified temperature gradients (initial or prescribed), except at the beginning of a restart run.

Forces and displacements: Forces can be read from unit 58 in the following format (unformatted sequential Fortran file):

```
READ (58) TIME, ((F(I,J), I=1,NUMNP), J=1, NOFC)  
Here NOFC is the number of force components. You must specify which force component corresponds to each value of J.
```

You must specify whether the times in unit 58 correspond exactly with the ADINA solution times, or whether they are different than the ADINA solution times. In the latter case, ADINA interpolates the forces to the ADINA solution times and writes the interpolated data to unit 58.

The forces read from unit 58 are added to the forces from the other specified loadings.

At the beginning of the solution analysis (or at the beginning of the restart run), the total nodal point forces are always defined by the displacement, velocity and accelerations at the beginning of the solution analysis (or at the beginning of the restart run), and the nodal point forces read from unit 58 are not used.

Displacements can be written to unit 58 in the format given above.

11.12 Remeshing options

- In some cases (e.g. rolling, forging), a finite element mesh might become distorted to such an extent that the solution stops. A restart analysis cannot be used to continue the solution; usually a new mesh has to be created in order to solve the problem. This section describes the remeshing options available in ADINA.
- The mapping option allows you to map the last obtained solution from the current mesh onto a new mesh. The mapping option is applicable to 2-D solid elements in large strain analysis only, when the plastic-bilinear or plastic-multilinear material models are used.
- When you anticipate that you will use the mapping option, you

must create a mapping file (extension .map) in the first run. In the AUI user interfaces, choose Control→Mapping and select “Create Mapping File” or from the AUI command line, use the command MASTER MAP-OUTPUT=YES. You must also save the AUI database file.

- To create a new mesh on the deformed geometry and continue the analysis, follow these steps:
 - ▶ Open the AUI database file.
 - ▶ In the AUI user interfaces, choose Control→Mapping and select “Remesh and Continue Analysis”, or from the AUI command line, use the command MASTER MAP-OUTPUT = REMESH.
 - ▶ Read the nodal deformations into the AUI. The nodal deformations are stored in a file with extension .modes (UNIX versions) or .mod (Windows version). To read the nodal deformations in the AUI user interfaces, specify the nodal deformations file in the Control→Mapping dialog box, or from the AUI command line, use the command MASTER NODAL-DEFORMATION-FILE = (filename).
 - ▶ When the AUI reads the nodal deformations, it creates new geometry lines corresponding to the boundaries of the deformed body. Note, in this AUI version, the AUI does not create 3-D deformed surfaces.
 - ▶ Now use the dialog boxes and/or commands of the AUI to create a new mesh. Set the solution start time to the last obtained solution time. You must use the same element types, material models and contact conditions as you used in the previous mesh. However, you can modify the loading, time stepping and boundary conditions.
 - ▶ Create a new data file and run ADINA. You do not have to provide the restart file from the previous run to the current ADINA run, but you must provide the mapping file from the previous run to the current ADINA run.

11.13 Analysis zooming

- The analysis zooming feature is used to analyze details of models starting from a relatively coarse mesh and subsequently zooming in on specific regions of interest. The feature is available in the ADINA program only.
- The initial (“coarse model”) analysis requires the user to specify that the analysis zooming feature might be used in subsequent runs. All other options are the same as usual. The ADINA program creates a mapping file which can be used in subsequent layered models.
- The analysis zooming feature is based on St. Venant’s principle that if an actual distribution of forces is replaced by a statically equivalent system, the distribution of stresses and strains is altered only near the regions of load applications.
- The advantages of this zooming method are
 - ▶ the need for transition regions is eliminated
 - ▶ a user can change the zoomed region and still use the full model initial solutions for every zooming calculation, thereby enabling the user to experiment with different designs of the region of interest
 - ▶ the method can help to demonstrate the adequacy of mesh refinements.
- When using the analysis zooming feature, note that
 - ▶ A user can zoom on any part of the model. Zooming can be repeated on the already zoomed parts. For every layer, a new mapping file is created, which can be used in further zooming on the model.
 - ▶ There can be more than one zoomed part.
 - ▶ Boundary conditions, material models and loading are

derived from the initial model, but they can be modified.

- ▶ The *.dat* file for the zoomed part must have a different name from the full model *.dat* file.
- ▶ Only 2-D solid and 3-D solid elements can be used in the zoom modeling.
- ▶ ADINA performs a step-by-step analysis. The *.map* file from the full model solution must be used in the analysis of the zoomed model. Although this is not a restart analysis, the *.map* file should be regarded as the restart file as in a standard restart run.
- ▶ For every time step, displacements from the full model solutions are mapped to the boundary nodes of the zoomed model. This process should not take a long time since only boundary nodes are mapped.
- ▶ The mapped displacements are treated in ADINA as prescribed displacements in the zoomed model.
- ▶ The user should be able to verify that the distance between the cut boundaries and the stress concentration region is adequate. This can be done by using the AUI (ADINA-PLOT). The AUI can overlay stress contour plots of the zoomed model on the same stress contour plots of the whole (coarse) model. In other words, the AUI can display the zoomed model on the (coarse) full model for all plotting options, namely, band plots and vector plots.
- ▶ The cut boundaries are assumed to be far enough away from stress concentration regions.

11.14 Stiffness stabilization

- Stiffness stabilization is a useful feature for stabilizing static problems (mainly involving contact) where there are insufficient constraints to remove rigid body motions, and yet there are no forces that will cause rigid body motion. This is sometimes the case
-

in contact problems right before contact starts and right after it ends.

- This feature adds a stabilizing effect to the stiffness matrix by scaling all diagonal stiffness terms (except those belonging to contact equations). At the same time, the right-hand-side load vector is not modified. The diagonal stiffness terms are modified as follows

$$\mathbf{K}_{ii} = (1 + \varepsilon_{STAB}) \mathbf{K}_{ii}$$

where ε_{STAB} is the stabilization factor. A default stabilization factor is selected by ADINA that is suitable for most stabilization problems. The user can override the default factor.

- Since the internal force vector is not affected by this stabilization, the final converged solution is the same as without stabilization. However, the iteration history, or the displacements at each iteration can be affected.
- Insufficiently supported static problems can alternatively be treated by adding weak springs at various locations in the model. Stiffness stabilization is generally more beneficial for the following reasons:
 - ▶ Determining the number, location and stiffness of the springs requires a lot of user intervention. This is not an issue in stiffness stabilization.
 - ▶ There may be no suitable location for the springs.
 - ▶ The stiffness of the spring has to be entered as an absolute value while the stiffness stabilization factor is normalized with respect to the stiffness matrix.
 - ▶ The springs generate an internal force, which affects the final solution while the stiffness stabilization does not. It is sometimes hard to assess how much the springs affect the final solution.
- Note that even if no supports are present in the model, the

stiffness matrix will be positive definite once the stabilization factor is applied.

- Stiffness stabilization can only be used for nonlinear analyses.

This page intentionally left blank.

12. Postprocessing considerations

- In this chapter we discuss those aspects of the AUI (ADINA User Interface) which require an in-depth discussion.
- This chapter discusses the postprocessing of results from all five solution programs ADINA, ADINA-T, ADINA-F, ADINA-FSI, ADINA-TMC.

12.1 Calculation of results within the AUI

- The ADINA system programs generate a wide variety of results, for example, stresses, displacements, reactions, heat fluxes, contact segment tractions. You access all of these results using several quite general AUI commands. To understand the inputs to these commands, the following concept is important to remember.
- The AUI must know the answers to the following four questions before it can obtain a result.
 - ▶ What result is to be obtained?
 - ▶ Where (in the model) is the result to be evaluated?
 - ▶ When (for which load step, mode shape, etc) is the result to be evaluated?
 - ▶ How is the result to be evaluated?

You can supply answers to each of these questions within the AUI. "What" is controlled by variables. "Where" is controlled by model points, model combination points, model lines, zones and result grids. "When" is controlled by response depictions and response-range depictions. "How" is controlled by smoothing depictions and result control depictions.

- It is also important to understand that the AUI can transform results computed by the solution programs (the available results) to the results that you request (the requested results). The AUI can make the following transformations:

Spatial transformations:

- ▶ The AUI can interpolate results available at the element integration points to other points within the elements, such as the element local nodes.
- ▶ The AUI can combine the results from all of the elements attached to a node into a single result (smoothing).
- ▶ The AUI can interpolate the results from the nodes to points within elements.

Time transformations:

- ▶ The AUI can interpolate results available at two solution times to solution times between these two times.

Resultant transformations:

- ▶ The AUI can combine results at a single point into a single result. For example, the AUI can compute the effective stress at a point from the stress components at the point.

Combination transformations:

- ▶ The AUI can combine results according to response spectrum formulas, harmonic and random vibration formulas or user-supplied combinations (known as response-combinations). The AUI can also scan for the most extreme result for given solution times and return the most extreme value (envelope combinations).

These transformations are organized in a hierarchy such that each transformation can request information from the transformation below it. The hierarchy is described in the following examples.

Example 1: As an example involving only the spatial transformation, consider the evaluation of a smoothed stress component at a point within an element (not necessarily a point at which the results were computed by ADINA) for an available time (a time for which results were computed by ADINA). This stress component is calculated by the spatial transformation as follows. The spatial transformation interpolates the available stress component data from the integration points to the nodes, performs smoothing, then interpolates the stress component to the requested point.

Example 2: As an example involving only the time and spatial transformations, consider the evaluation of a smoothed stress component at a point within an element for a requested time (not necessarily a time for which results were computed by ADINA). This stress component is calculated by the time transformation. The time transformation requests the stress component for the available solution times nearest to the requested time, then interpolates in time to the requested solution time. The stress components at the available solution times are calculated by the spatial transformation as in Example 1.

In the above example, the spatial transformation acts as a subroutine that is called by the time transformation.

Example 3: As an example involving the resultant, time and spatial transformations, consider the evaluation of smoothed effective stress at a point within an element for a requested time. The effective stress is computed by the resultant transformation. The resultant transformation requests the stress components from the time transformation, then computes the effective stress. The time transformation computes the stress components as in Example 2.

In the above example, the time transformation acts as a subroutine that is called by the resultant transformation.

Example 4: As an example involving the combination, time and spatial transformations, consider the evaluation of stress component difference at a point within an element between two requested times. (E.g., $\sigma_{xx}(\text{time } 2.0) - \sigma_{xx}(\text{time } 1.0)$.) The stress component difference is computed by the combination transformation. The combination transformation requests the stress components from the time transformation, then computes the difference. The time

transformation computes the stress components as in Example 2.

In the above example, the time transformation acts as a subroutine that is called by the combination transformation.

- In the above examples, the AUI automatically determines the hierarchy order. However, when considering examples involving both the resultant and the combination transformations, you can control the order of these transformations. There are two cases:

Resultants before combinations: The combination transformation can request results from the resultant transformation, the resultant transformation can request results from the time transformation, the time transformation can request results from the spatial transformation.

In other words, the combination transformation acts on resultants.

Resultants after combinations: The resultant transformation can request results from the combination transformation, the combination transformation can request results from the time transformation, the time transformation can request results from the spatial transformation.

In other words, the combination transformation acts on the variables contained within the resultant.

The default is “resultants before combinations”.

The following examples show cases in which the hierarchy order is important.

Example 5: Consider the calculation of the radial stress in response spectrum analysis. The response spectrum calculation is a combination transformation and the radial stress calculation is a resultant transformation.

If “resultants before combinations” is used, the combination transformation requests the radial stresses for each mode shape from the resultant transformation, then the combination transformation combines the modal radial stresses using the response spectrum formulas.

If “resultants after combinations is used”, the resultant transformation requests the stress components from the combination transformation, then the resultant transformation computes the radial stress. The combination transformation calculates the stress components using the modal stress components and the response spectrum formulas.

In this example, “resultants before combinations” is correct, and it would be incorrect to use “resultants after combinations”.

Example 6: Consider the calculation of the effective stress for a response combination involving the addition of two load cases. The response combination calculation is a combination transformation and the effective stress calculation is a resultant transformation.

If “resultants before combinations” is used, the combination transformation requests the effective stresses for each load case from the resultant transformation, then the combination transformation adds the effective stresses.

If “resultants after combinations is used”, the resultant transformation requests the stress components from the combination transformation, then the resultant transformation computes the effective stress. The combination transformation adds the stress components from the two load cases.

In this example, “resultants after combinations” is correct, and it would be incorrect to use “resultants before combinations”.

Example 7: Consider the calculation of the effective stress in harmonic vibration analysis, in which you enter the value of ωt . The harmonic vibration analysis calculation is a combination transformation and the effective stress calculation is a resultant transformation.

If “resultants before combinations” is used, the combination transformation requests the effective stresses for each mode shape from the resultant transformation, then the combination transformation combines the modal effective stresses using the harmonic vibration formulas.

If “resultants after combinations is used”, the resultant transformation requests the stress components from the combination transformation, then the resultant transformation computes the effective stress. The combination transformation calculates the stress components using the modal stress components and the harmonic vibration formulas.

In this example, “resultants after combinations” is correct, and it would be incorrect to use “resultants before combinations”.

Example 8: Consider the calculation of the resultant $1+u_i$, where u_i is a displacement component, in harmonic vibration analysis.

If “resultants before combinations” is used, the combination transformation requests the quantity $1+\phi$ for each mode shape from the resultant transformation (where ϕ is the corresponding eigenvector component for each mode shape), then the combination transformation combines the quantities $1+\phi$ using the harmonic vibration formulas.

If “resultants after combinations” is used”, the resultant transformation requests u_i from the combination transformation, then the resultant transformation computes $1+u_i$. The combination transformation calculates u_i using ϕ for each mode shape and the harmonic vibration formulas.

In this example, “resultants after combinations” is correct, and it would be incorrect to use “resultants before combinations”.

Example 9: Consider the calculation of effective stress in random vibration analysis.

If “resultants before combinations” is used, the combination transformation requests the effective stresses for each mode shape from the resultant transformation, then the combination transformation combines the modal effective stresses using the random vibration formulas.

If “resultants after combinations is used”, the resultant transformation requests the stress components from the combination transformation, then the resultant transformation computes the effective stress. The combination transformation calculates the stress components using the modal stress components and the random vibration formulas.

In this example, neither “resultants after combinations” nor “resultants before combinations” is correct. It would be better if the AUI could estimate the effective stress by changing the signs of the stress components output from the combination transformation, so as to maximize the effective stress.

- The general rule of thumb is, if both the combination transformation and the resultant transformation are linear, it doesn't matter which rule to use; both will give the correct results. If the combination transformation is nonlinear and the resultant transformation is linear, use "resultants before combinations". If the combination transformation is linear and the resultant transformation is nonlinear, use "resultants after combinations". If both transformations are nonlinear, then neither rule will give the correct results.
- There is an additional consideration when the model point is a combination point, for example a node combination point (in which the results at the nodes in the combination point are summed).

When "resultants before combinations" is used, the combination transformation requests the summed resultant (the summing being performed by the combination point).

When "resultants after combinations" is used, the summing (by the combination point) is performed on the variables referenced by the resultant, then the resultant is taken.

Example 10: Consider the calculation of the effective stress for a response combination involving the subtraction of two load cases. Here the effective stress is evaluated at a node combination point.

If "resultants before combinations" is used, the combination transformation requests the summed effective stresses for each load case, then the combination transformation subtracts the summed effective stresses.

If "resultants after combinations" is used", the combination point requests the effective stresses from the resultant transformation and then sums the effective stresses. The resultant transformation computes the effective stresses from the stress components. The combination transformation subtracts the stress components from the two load cases.

- In the following, we discuss what results are available, where they can be evaluated and how they are evaluated.

12.1.1 What results are available

- The results computed by the ADINA system programs are made

available to the AUI when the corresponding porthole file is loaded. In many cases you control the results that are available when you define the model during preprocessing. For example, reactions are computed if you check the "Calculate Reaction Forces/Moments" box within the Control→Miscellaneous Options dialog box.

Regarding results output along with the elements, in general you can either output stresses or forces, but not both. This choice is made within the element group definitions dialog box. For example, for a 2-D solid element group, the choice is made with the "Calculated Element Results" buttons.

When you choose to output stresses, the actual results output depend upon the material model and upon the stress reference system employed. These results are described in Chapter 2 of this manual for each element group.

Regarding results output along with the contact surfaces, you can output either segment tractions or forces. This choice is made within the contact group definitions dialog box. However, ADINA calculates segment tractions only for contactor contact surfaces. The actual results output are described in Chapter 4 of this manual.

- Each type of result output by ADINA is referred to by a variable name.

The variables can be classified according to where (in the model) they are output by the solution program and how they are interpolated within the model. The variable location types are:

Location-independent: Variables such as time or frequency that are not associated with specific points in the model.

Node field: Variables such as displacement, velocity or acceleration that are output at node points and that can be interpolated within elements (using the nodal shape functions).

Element/layer field: Variables such as stresses that are output within elements or element layers and that can be interpolated within elements. These variables are output by the solution program at the integration points.

Contact: Variables such as normal traction that are output at contact segments (ADINA only).

Draw bead: Variables such as restraining traction and uplifting traction that are output at draw bead segments (ADINA only).

Section field: Variables such as moments that are output within the sections of elements (midsurface of shells or neutral line of Hermitian beams) and that can be interpolated within element sections (ADINA only).

Radiation field: Variables that are output at radiosity segments (ADINA-T only).

Line contour: Variables that are output for each fracture mechanics line contour (ADINA only).

Virtual shift: Variables that are output for each fracture mechanics virtual shift (ADINA only).

Node discrete: Variables such as reaction that are output at node points and that cannot be interpolated within elements.

Element local node: Variables such as consistent nodal forces that are output at the element local nodes and that cannot be interpolated within elements.

Element local node field: Variables such as shell thicknesses that are output at the element local nodes and that can be interpolated within elements.

Coordinate: Variables such as coordinate.

Mesh surface: Variables, such as the surface normal, that can only be evaluated at mesh integration or mesh extreme points. The variables can also be classified according to function. The classifications used in the AUI are: *displacement, velocity, acceleration, temperature, fluid, latent heat, stress, strain, force, traction, reaction, flux, electric field, fracture, failure criterion, eigenvector, prescribed load, coordinate, thickness, time, frequency/mode, miscellaneous, user-defined*.

Tables 12.1-1 to 12.1-23 show all of the variables. Each table

shows all of the variables of the given function category. In the tables, we compress the variable names using characters enclosed in parentheses. The compression rules are as follows:

Numerical lists: Entries enclosed in parentheses are to be expanded with one variable per character in the parentheses. For example, OVALIZATION- (123456) represents the six variables OVALIZATION-1, OVALIZATION-2, OVALIZATION-3, OVALIZATION-4, OVALIZATION-5, OVALIZATION-6.

Vectors: Entries enclosed in parentheses are to be expanded with one variable per character in the parentheses. For example, (ABC) -DISPLACEMENT represents the three variables A-DISPLACEMENT, B-DISPLACEMENT, C-DISPLACEMENT.

Symmetric tensors: Entries enclosed in parentheses are to be expanded to give components as in this example:
STRESS- (XYZ) represents the six variables STRESS-XX, STRESS-YY, STRESS-ZZ, STRESS-XY, STRESS-XZ, STRESS-YZ.

Other cases: Entries enclosed in parentheses and separated by commas are to be expanded as in this example:
CURVATURE- (RB, SB, RBSB, T) represents the four variables CURVATURE-RB, CURVATURE-SB, CURVATURE-RBSB, CURVATURE-T.

We also use the * to denote any number starting from 1, as in this example: TIME_FUNCTION_* represents the variables TIME_FUNCTION_1, TIME_FUNCTION_2, etc.

The tables give notes for those variables that are not self-explanatory.

Table 12.1-1: Displacement variables

Variable name	Variable location type	Notes
(ABC) -DISPLACEMENT	Node field	1
(XYZ) -DISPLACEMENT	Node field	
DISPLACEMENT_MAGNITUDE	Node field	2
FLUID_P0	Node field	
FLUID_POTENTIAL	Node field	
OVALIZATION-(123456)	Node field	
(ABC)-ROTATION	Node field	
(V1,V2)-ROTATION	Node field	
(XYZ)-ROTATION	Node field	
WARPING-(123456)	Node field	

1) The A, B and C components refer to the skew coordinate system (if one is used). The V1, V2 components refer to the shell rotation directions (for shell midsurface nodes with five degrees of freedom). The X, Y and Z components always refer to the global coordinate system. The AUI can transform results that were calculated by ADINA in one coordinate system to another coordinate system, see Table 12.1-24.

This comment holds for velocities, accelerations and eigenvectors as well as displacements.

2) DISPLACEMENT_MAGNITUDE is the square root of the sum of the squares of the displacement components, as computed by the AUI.

Table 12.1-2: Velocity variables

Variable name	Variable location type	Notes
(ABC)-ANGULAR_VELOCITY	Node field	
(V1,V2)-ANGULAR_VELOCITY	Node field	
(XYZ)-ANGULAR_VELOCITY	Node field	
ELEMENT_(XYZ)-VELOCITY	Element field	
FLUID_P0_VELOCITY	Node field	1
FLUID_POTENTIAL_VELOCITY	Node field	2
NODAL_SLIP_VELOCITY	Node field	7
NODAL_SLIP_VELOCITY-(XYZ)	Node field	8
OVALIZATION_VELOCITY-(123456)	Node field	3
PORE_PRESSURE_VELOCITY	Node field	5
(ABC)-VELOCITY	Node field	

(XYZ) -VELOCITY	Node field	
VELOCITY_MAGNITUDE	Node field	6
WARPING_VELOCITY- (123456)	Node field	4

- 1) This is the 1st time derivative of the fluid P_0 degree of freedom.
 - 2) This is the 1st time derivative of the fluid potential degree of freedom.
 - 3) These are the 1st time derivatives of the ovalization degrees of freedom.
 - 4) These are the 1st time derivatives of the warping degrees of freedom.
 - 5) This is the 1st time derivative of the pore pressure (see Table 12.1-7 for the pore pressure).
 - 6) VELOCITY_MAGNITUDE is the square root of the sum of the squares of the velocity components, as computed by the AUI.
 - 7) NODAL_SLIP_VELOCITY is the magnitude of the contact slip velocity. It is output when the “new” contact segments are used in frictional contact.
 - 8) NODAL_SLIP_VELOCITY- (XYZ) are the components of the contact slip velocity. These are output when the “new” contact segments are used in frictional contact.
-

Table 12.1-3: Acceleration variables

Variable name	Variable location type	Notes
(ABC) -ACCELERATION	Node field	
(XYZ) -ACCELERATION	Node field	
ACCELERATION_MAGNITUDE	Node field	6
(ABC) -ANGULAR_ACCELERATION	Node field	
(V1, V2) -ANGULAR_ACCELERATION	Node field	
(XYZ) -ANGULAR_ACCELERATION	Node field	
FLUID_P0_ACCELERATION	Node field	1
FLUID_POTENTIAL_ACCELERATION	Node field	2
OVALIZATION_ACCELERATION- (123456)	Node field	3
PORE_PRESSURE_ACCELERATION	Node field	5
WARPING_ACCELERATION- (123456)	Node field	4

- 1) This is the 2nd time derivative of the fluid P_0 degree of freedom.
 - 2) This is the 2nd time derivative of the fluid potential degree of freedom.
 - 3) These are the 2nd time derivatives of the ovalization degrees of freedom.
 - 4) These are the 2nd time derivatives of the warping degrees of freedom.
 - 5) This is the 2nd time derivative of the pore pressure (see Table 12.1-7 for the pore pressure).
 - 6) ACCELERATION_MAGNITUDE is the square root of the sum of the squares of the acceleration components, as computed by the AUI.
-

Table 12.1-4: Temperature variables

Variable name	Variable type	Notes
ELEMENT_TEMPERATURE	Element field	1
TEMPERATURE	Node field	

- 1) This is the temperature as output at the element integration points.
-

Table 12.1-5: Fluid variables

Variable name	Variable location type	Notes
CELL_PECLET_NUMBER	Element field	
CELL_REYNOLDS_NUMBER	Element field	
EFFECTIVE_VISCOSITY	Element field	
MASS_RATIO_*	Node field	
NODAL_VISCOSITY	Node field	
OMEGA-(XY,XZ,YZ)	Element field	
STREAM_FUNCTION	Node field	
TOTAL_HEAD	Node field	
TURBULENCE_EPSILON	Node field	
TURBULENCE_K	Node field	
TURBULENCE_W	Node field	

Table 12.1-6: Latent heat variables

Variable name	Variable location type	Notes
LATENT_HEAT_INTERFACE *	Node field	

Table 12.1-7: Stress variables

Variable name	Variable location type	Notes
AXIAL_STRESS	Element field	18
DISTORTIONAL_STRESS	Element field	
EFFECTIVE_STRESS	Element field	1,3
EQUIV_INTERNAL_AXIAL_PRESSURE	Element field	
EQUIV_INTERNAL_HOOP_PRESSURE	Element field	
FE_EFFECTIVE_STRESS	Element field	2,3
FE_MAX_SHEAR_STRESS	Element field	2
FE_PRESSURE	Element field	2,4
FE_SIGMA-P1	Element field	2,5
FE_SIGMA-P1_ANGLE	Element field	2,5
FE_SIGMA-P1_DIRECTION-(XYZ)	Element field	2,5
FE_SIGMA-P2	Element field	2,5
FE_SIGMA-P2_DIRECTION-(XYZ)	Element field	2,5
FE_SIGMA-P3	Element field	2,5
FE_SIGMA-P3_DIRECTION-(XYZ)	Element field	2,5
FLUID_REFERENCE_PRESSURE	Element field	
GASKET_PRESSURE	Element field	
GASKET_STRESS-(ABC)	Element field	17
GASKET_YIELD_STRESS	Element field	
GENERAL_ELEMENT_STRESS	Element field	
GRAVITY_IN-SITU_PRESSURE	Element field	
HILL_EFFECTIVE_STRESS	Element field	
MAX_SHEAR_STRESS	Element field	1,7
MEAN_STRESS	Element field	
NODAL_PRESSURE	Node field	4
PORE_PRESSURE	Node field	15
PRESSURE	Element field	1,4
PRESSURE_GRADIENT_MAGNITUDE	Element field	16
SIGMA-NORM2	Element field	1,8
SIGMA-(P1,P2,P3)	Element field	1,6
SIGMA-(P1,P2,P3)_DIRECTION-(XYZ)	Element field	1,14

STRESS- (123)	Element field	9
STRESS- (ABC)	Element field	10
STRESS- (IJK)	Element field	11
STRESS- (RST)	Element field	12
STRESS- (XYZ)	Element field	13
STRESS_THRU_THICKNESS	Element field	
YIELD_STRESS	Element field	

- 1) These variables are computed by the AUI from data output by the solution program.
- 2) Variables that begin with FE_ denote quantities that are output directly from the solution program. For example, FE_EFFECTIVE_STRESS is the effective stress as output from ADINA for certain elements/material models.
These variables should be distinguished from variables defined in the AUI as predefined resultants (see Note 1).
- 3) Note that, at the integration points, EFFECTIVE_STRESS and FE_EFFECTIVE_STRESS are numerically exactly the same. But the AUI interpolates (and possibly smooths) EFFECTIVE_STRESS from the integration points differently than it interpolates FE_EFFECTIVE_STRESS. The AUI interpolates EFFECTIVE_STRESS by interpolating the stress components (possibly smoothing them as well) individually; then, after interpolation, the AUI computes EFFECTIVE_STRESS using the formula

$$\sigma_e = \sqrt{1/2 \left((\tau_{xx} - \tau_{yy})^2 + (\tau_{xx} - \tau_{zz})^2 + (\tau_{yy} - \tau_{zz})^2 + 6(\tau_{xy}^2 + \tau_{xz}^2 + \tau_{yz}^2) \right)}$$

However, the AUI interpolates and smooths FE_EFFECTIVE_STRESS using the same algorithms that it uses for any other stress component.

- 4) The comments given in Note 3 apply to the variables PRESSURE (computed as $p = -\frac{\tau_{xx} + \tau_{yy} + \tau_{zz}}{3}$ by the AUI) and FE_PRESSURE. The variable NODAL_PRESSURE is output by ADINA-F (since pressure is calculated at the nodes in ADINA-F), and, when postprocessing ADINA-F results, you can use NODAL_PRESSURE to plot or list the nodal pressures.
- 5) These are the principal stresses output from ADINA. Please note: for the concrete

and curve description models, these are the true principal stresses only before cracking has occurred. After cracking, the directions are fixed corresponding to the crack directions and these variables are no longer principal stresses.

For 2-D elements, `FE_SIGMA-P3` is always out of the plane of the element and `FE_SIGMA-P1`, `FE_SIGMA-P2` are always in the plane of the element ($\text{FE_SIGMA-P1} \geq \text{FE_SIGMA-P2}$).

- 6) These are the principal stresses as computed by the AUI from the stress components ($\text{SIGMA-P1} \geq \text{SIGMA-P2} \geq \text{SIGMA-P3}$).
- 7) This is the maximum shear stress, defined as one-half the difference between the maximum and minimum principal stress, as computed by the AUI from the stress components.
- 8) This is the 2-norm of the stress tensor (the larger of the absolute values of the maximum and minimum principal stress) as computed by the AUI from the stress components.
- 9) These are the stresses transformed by the AUI into a user-specified coordinate system. More information is given at the end of this section.
Note that these variables have nothing to do with the initial stresses entered in the initial strain directions.
- 10) These are the stress components in the axes of the material coordinate system, for orthotropic materials. Note that the AUI cannot transform stresses output in other coordinate systems to the material coordinate system.
- 11) These are equal to the global stress components for those elements in which the stresses are output in the global coordinate system, equal to the isoparametric coordinate system stress components for those elements in which the stresses are output in the isoparametric coordinate system, and equal to the material coordinate system stress components for those elements in which the stresses are output in the material coordinate system. These variables are mainly used for constructing resultants that are invariants (that is, do not depend on the coordinate system direction). For example, resultant variable `EFFECTIVE_STRESS` is defined in terms of `STRESS-II`, `STRESS-JJ`, etc.
- 12) These are the stress components in the axes of the isoparametric coordinate system. For shell elements, component directions R and S refer to directions \bar{r} and \bar{s} (see Section 2.7.1 and Figure 2.7-6). Note that the AUI cannot transform stresses output in other coordinate systems to the isoparametric coordinate system.

- 13) These are the stress components in the global coordinate system. Note that the AUI cannot transform stresses output by ADINA in other coordinate systems to the global coordinate system.
- 14) These are the directions of the principal stresses SIGMA-P1, SIGMA-P2, SIGMA-P3 (see note 6).
- 15) This is the value of the pore pressure, used in the porous media formulation.
- 16) PRESSURE_GRADIENT_MAGNITUDE is implemented for the ADINA-F 3- and 4-node 2D elements, and for the 4- and 8-node 3D elements. The purpose of PRESSURE_GRADIENT_MAGNITUDE is to provide an ADINA-F variable that should be smooth in the mathematical solution but that may not be smooth in the finite element solution. A band plot of PRESSURE_GRADIENT_MAGNITUDE will show jumps between elements.
- 17) In 2-D analysis, the BB, AB and CC components are defined; in 3-D analysis, the BB, CC, AB, AC, BC components are defined.
- 18) In band plots, AXIAL_STRESS can be used instead of STRESS-RR.

Table 12.1-8: Strain variables

Variable name	Variable location type	Notes
ACCUM_EFF_CREEP_STRAIN	Element field	11
ACCUM_EFF_PLASTIC_STRAIN	Element field	5
ACCUM_EFF_TRANSF_STRAIN	Element field	12
ACCUM_PLASTIC_AXIAL_STRAIN	Element field	
ACCUM_PLASTIC_CURVATURE-(ST)	Section field	
ACCUM_PLASTIC_TWIST	Section field	
AUSTENITE_FRACTION	Element field	12
AXIAL_STRAIN	Section field	
CREEP_STRAIN-(ABC)	Element field	1
CREEP_STRAIN-(RST)	Element field	1
CREEP_STRAIN-(XYZ)	Element field	1
CREEP_STRAIN_THRU_THICKNESS	Element field	
CURVATURE-(RB,SB,RBSB)	Section field	
CURVATURE-(ST)	Section field	

CURVATURE- (XL, YL, XLYL)	Element field	
DEFORMATION_GRADIENT- (XYZ)	Element field	6
DETWINNED_MARTENSITE_FRACTION	Element field	12
EFFECTIVE_CREEP_STRAIN	Element field	
GASKET_CLOSURE_STRAIN	Element field	
GASKET_PLASTIC_CLOSURE_STRAIN	Element field	
GASKET_STRAIN- (ABC)	Element field	9
GASKET_THERMAL_STRAIN	Element field	
GASKET_THERMAL_STRAIN- (ABC)	Element field	9
IRRADIATION_STRAIN	Element field	
LOGSTRAIN_P1	Element field	10
LOGSTRAIN_P2	Element field	10
LOGSTRAIN_P3	Element field	10
LSTRETCH- (XYZ)	Element field	7
MEMBRANE_STRAIN- (RB, SB, RBSB)	Section field	
MEMBRANE_STRAIN- (XL, YL, XLYL)	Element field	
NEUTRAL_AXIS_POSITION- (RB, SB)	Section field	
PLASTIC_AXIAL_STRAIN	Section field	
PLASTIC_CURVATURE- (ST)	Section field	
PLASTIC_STRAIN- (ABC)	Element field	1
PLASTIC_STRAIN- (RST)	Element field	1
PLASTIC_STRAIN- (XYZ)	Element field	1
PLASTIC_STRAIN_THRU_THICKNESS	Element field	
PLASTIC_TWIST	Section field	
RSTRETCH- (XYZ)	Element field	8
SHEAR_STRAIN- (RB, SB)	Section field	
STRAIN- (123)	Element field	3
STRAIN- (ABC)	Element field	1,4
STRAIN- (RST)	Element field	1,4
STRAIN- (XYZ)	Element field	1,4
STRAIN_THRU_THICKNESS	Element field	
STRETCH- (XYZ)	Element field	2
THERMAL_STRAIN	Element field	
THERMAL_STRAIN- (ABC)	Element field	1
THERMAL_STRAIN- (RST)	Element field	1
THERMAL_STRAIN- (XYZ)	Element field	1
THERMAL_STRAIN_THRU_THICKNESS	Element field	
TRANSFORMATION_STRAIN- (RST)	Element field	12
TRANSFORMATION_STRAIN- (XYZ)	Element field	12

TWINNED_MARTENSITE_FRACTION	Element field	12
TWIST	Section field	
VOLUMETRIC_STRAIN	Element field	

1) The off-diagonal strain terms are engineering strains; i.e. twice the tensorial strains.

2) These are the Cartesian components of the left stretch tensor \mathbf{V} , as output by ADINA. See Section 3.1, equation 3.1-5 for the definition and meaning of \mathbf{V} .

The principal stretches can be computed in the AUI by defining a resultant using functions SIGP1, SIGP2 and SIGP3. For example, you can define, for 3-D problems,

```
MAX_STRETCH = SIGP1 (<STRETCH-XX>, <STRETCH-YY>,
                      <STRETCH-ZZ>, <STRETCH-XY>,
                      <STRETCH-XZ>, <STRETCH-YZ>)
```

The left stretch tensor is also used in the AUI for plotting the stretches as element vectors.

3) These are the strain components transformed by the AUI into a user-specified coordinate system. See the end of this section for more information.

Note that these strains have nothing to do with the initial strains entered in the initial strain directions.

4) The same comments as are given for the stress variables (notes 10, 12, 13 of Table 12.1-7) apply to the strain variables.

5) The definition of the accumulated effective plastic strain is given in Section 3.4.1.

6) These are the components of the deformation gradient tensor ${}^t_0\mathbf{X}$, as output from ADINA. Note that there are 9 components, since the deformation gradient tensor is in general not symmetric.

7) These are the Cartesian components of the right stretch tensor ${}^t_0\mathbf{V}$, as computed by the AUI from the deformation gradients output by ADINA. See Section 3.1, equation 3.1-5 and Figure 3.1-3 for the definition and meaning of ${}^t_0\mathbf{V}$.

The principal stretches can be computed in the AUI by defining a resultant using functions SIGP1, SIGP2 and SIGP3. For example, you can define, for 3-D problems,

```
MAX_STRETCH = SIGP1(<LSTRETCH-XX>, <LSTRETCH-YY>,
                     <LSTRETCH-ZZ>, <LSTRETCH-XY>,
                     <LSTRETCH-XZ>, <LSTRETCH-YZ>)
```

The left stretch tensor is also used in the AUI for plotting the stretches as element vectors.

8) These are the Cartesian components of the right stretch tensor \mathbf{U} , as computed by the AUI from the deformation gradients output by ADINA. See Section 3.1, equation 3.1-1 for the definition and meaning of \mathbf{U} .

The principal stretches can be computed in the AUI by defining a resultant using functions SIGP1, SIGP2 and SIGP3. For example, you can define, for 3-D problems,

```
MAX_STRETCH = SIGP1(<RSTRETCH-XX>, <RSTRETCH-YY>,
                     <RSTRETCH-ZZ>, <RSTRETCH-XY>,
                     <RSTRETCH-XZ>, <RSTRETCH-YZ>)
```

9) In 2-D analysis, the BB, AB and CC components are defined; in 3-D analysis, the BB, CC, AB, AC, BC components are defined.

10) These are the maximum, intermediate and minimum principal logarithmic strains. They are available in large strain analysis.

11) ACCUM_EFF_CREEP_STRAIN is defined in Section 3.6.3.

12) These variables are used with the SMA material defined in Section 3.14.

Table 12.1-9: Force variables

Variable name	Variable location type	Notes
AVERAGE_BENDING_MOMENT	Section field	1,6
AVERAGE_MEMBRANE_FORCE	Section field	2,6
AVERAGE_SHEAR_FORCE	Section field	6
AXIAL_FORCE	Section field	
BENDING_MOMENT-(RB, SB, RBSB)	Section field	3
BENDING_MOMENT-(ST)	Section field	
BENDING_MOMENT-(XL, YL, XLYL)	Element field	4
CONTACT_FORCE-(ABC)	Node discrete	
CONTACT_FORCE-(XYZ)	Node discrete	
DAMPING_FORCE	Element field	
EFFECTIVE_BENDING_MOMENT	Section field	

EFFECTIVE_MEMBRANE_FORCE	Section field	
ELASTIC_FORCE	Element field	
FORCE-(ABC)	Element field	
FORCE-(RST)	Element field	
FORCE-(XYZ)	Element field	
FORCE_RST_MAGNITUDE	Element field	11
MAX_PRINCIPAL_BENDING_MOMENT	Section field	6, 7
MAX_PRINCIPAL_MEMBRANE_FORCE	Section field	6, 8
MEMBRANE_FORCE-(RB, SB, RBSB)	Section field	3
MEMBRANE_FORCE-(XL, YL, XLYL)	Element field	4
MIN_PRINCIPAL_BENDING_MOMENT	Section field	6, 9
MIN_PRINCIPAL_MEMBRANE_FORCE	Section field	6, 10
MOMENT-(ABC)	Element field	
MOMENT-(RST)	Element field	
MOMENT-(XYZ)	Element field	
MOMENT_RST_MAGNITUDE	Element field	11
NODAL_FORCE-(RST)	El local node	5
NODAL_FORCE-(XYZ)	El local node	5, 14
NODAL_FORCE_RST_MAGNITUDE	El local node	11
NODAL_MOMENT-(RST)	El local node	5
NODAL_MOMENT-(XYZ)	El local node	5, 14
NODAL_MOMENT_RST_MAGNITUDE	El local node	11
RESTRAINING_FORCE-(XYZ)	Node discrete	12
SHEAR_FORCE-(RB, SB)	Section field	3
SOLITARY_CONTACT_FORCE-(ABC)	Node discrete	
SOLITARY_CONTACT_FORCE-(XYZ)	Node discrete	
TORSIONAL_MOMENT	Section field	
UPLIFTING_FORCE-(XYZ)	Node discrete	13
YIELD_AXIAL_FORCE	Section field	
YIELD_BENDING_MOMENT-(ST)	Section field	
YIELD_TORSIONAL_MOMENT	Section field	

- 1) This is evaluated by the AUI as $0.5(M_{\bar{r}\bar{r}} - M_{\bar{s}\bar{s}})$, which is an invariant (does not depend on the directions of \bar{r} and \bar{s}).
- 2) This is evaluated by the AUI as $0.5(N_{\bar{r}\bar{r}} - N_{\bar{s}\bar{s}})$, which is an invariant (does not

depend on the directions of \bar{r} and \bar{s} .

- 3) RB denotes the \bar{r} direction, SB denotes the \bar{s} direction.
- 4) XL denotes the element local x direction, YL denotes the element local y direction.
- 5) These are the consistent nodal point forces corresponding to the stresses within the element. The AUI does not transform these from the coordinate system in which they are output by ADINA.
- 6) These are computed by the AUI from the data output by ADINA.
- 7) This is evaluated by the AUI as the maximum principal value of $\begin{bmatrix} -M_{\bar{s}\bar{s}} & M_{\bar{r}\bar{s}} \\ M_{\bar{r}\bar{s}} & M_{\bar{r}\bar{r}} \end{bmatrix}$.
- 8) This is evaluated by the AUI as the maximum principal value of $\begin{bmatrix} N_{\bar{r}\bar{r}} & N_{\bar{r}\bar{s}} \\ N_{\bar{r}\bar{s}} & N_{\bar{s}\bar{s}} \end{bmatrix}$.
- 9) This is evaluated by the AUI as the minimum principal value of $\begin{bmatrix} -M_{\bar{s}\bar{s}} & M_{\bar{r}\bar{s}} \\ M_{\bar{r}\bar{s}} & M_{\bar{r}\bar{r}} \end{bmatrix}$.
- 10) This is evaluated by the AUI as the minimum principal value of $\begin{bmatrix} N_{\bar{r}\bar{r}} & N_{\bar{r}\bar{s}} \\ N_{\bar{r}\bar{s}} & N_{\bar{s}\bar{s}} \end{bmatrix}$.
- 11) Each of these variables is the square root of the sum of the squares of the components, as computed by the AUI.
- 12) RESTRAINING_FORCE- (XYZ) is the restraining forces on the nodes of the drawbead segments.
- 13) UPLIFTING_FORCE- (XYZ) is the uplifting forces on the nodes of the drawbead segments.
- 14) For 2D, 3D, plate, shell and spring elements, these variables are output in the coordinate system of the node to which the element is attached (global system or skew system). For example, if a 3D element is attached to nodes with skew systems, variable NODAL_FORCE-X is the nodal force in the skew A direction.

Table 12.1-10: Traction variables

Variable name	Variable location type	Notes
NODAL_NORMAL_TRACTION	Node field	6, 7
NODAL_NORMAL_TRACTION- (XYZ)	Node field	6
NODAL_TANGENTIAL_TRACTION	Node field	8, 9
NODAL_TANGENTIAL_TRACTION- (XYZ)	Node field	8
NORMAL_TRACTION	Contact	1
NORMAL_TRACTION- (XYZ)	Contact	
RESTRAINING_TRACTION- (XYZ)	Drawbead segment	4
SURFACE_TRACTION- (XYZ)	Mesh surface	2
TANGENTIAL_TRACTION	Contact	3
TANGENTIAL_TRACTION- (XYZ)	Contact	
UPLIFTING_TRACTION- (XYZ)	Drawbead segment	5

1) This is the magnitude of the normal traction vector.

2) These are computed by the AUI as predefined resultants using the formulas

$$\begin{aligned} \text{SURFACE_TRACTION-X} &= <\text{STRESS-XX}> * <\text{SURFACE_NORMAL-X}> + \\ &\quad <\text{STRESS-XY}> * <\text{SURFACE_NORMAL-Y}> + \\ &\quad <\text{STRESS-XZ}> * <\text{SURFACE_NORMAL-Z}> \end{aligned}$$

$$\begin{aligned} \text{SURFACE_TRACTION-Y} &= <\text{STRESS-XY}> * <\text{SURFACE_NORMAL-X}> + \\ &\quad <\text{STRESS-YY}> * <\text{SURFACE_NORMAL-Y}> + \\ &\quad <\text{STRESS-YZ}> * <\text{SURFACE_NORMAL-Z}> \end{aligned}$$

$$\begin{aligned} \text{SURFACE_TRACTION-Z} &= <\text{STRESS-XZ}> * <\text{SURFACE_NORMAL-X}> + \\ &\quad <\text{STRESS-YZ}> * <\text{SURFACE_NORMAL-Y}> + \\ &\quad <\text{STRESS-ZZ}> * <\text{SURFACE_NORMAL-Z}> \end{aligned}$$

See Table 12.1-17 for the definitions of the surface normal variables.

3) This is the magnitude of the tangential traction vector.

4) RESTRAINING_TRACTION- (XYZ) is the restraining tractions on the drawbead segments.

5) UPLIFTING_TRACTION- (XYZ) is the uplifting tractions on the drawbead segments.

- 6) These variables are output when the “new” contact segments are used.
- 7) NODAL_NORMAL_TRACTION is the magnitude of the normal traction.
- 8) These variables are output when the “new” contact segments are used in frictional contact.
- 9) NODAL_TANGENTIAL_TRACTION is the magnitude of the tangential traction.

Table 12.1-11: Reaction variables

Variable name	Variable location type	Notes
(ABC)-MOMENTREACTION	Node discrete	1
(V1,V2)-MOMENTREACTION	Node discrete	
(XYZ)-MOMENTREACTION	Node discrete	
(ABC)-REACTION	Node discrete	
(XYZ)-REACTION	Node discrete	
REACTION_MAGNITUDE	Node discrete	2

- 1) Reactions are only output for those nodes with fixities. Reactions that are numerically zero might not be output by ADINA.
- 2) REACTION_MAGNITUDE is the square root of the sum of the squares of the reaction components, as computed by the AUI.

Table 12.1-12: Flux variables

Variable name	Variable location type	Notes
AXIAL_HEAT_FLUX	Element field	
HEAT_FLUX_SURFACE	Mesh surface	1
HEAT_FLUX-(XYZ)	Element field	
HEAT_FLUX_MAGNITUDE	Element field	6
KINETIC_ENERGY_FLUX_SURFACE	Mesh surface	2
MASS_FLUX_SURFACE	Mesh surface	3
MOMENTUM_FLUX-(XYZ)_SURFACE	Mesh surface	4
RADIATION_HEAT_FLUX-(XYZ)	Radiation field	
SEEPAGE_FLUX-(XYZ)	Element field	
SEEPAGE_FLUX_MAGNITUDE	Element field	6

VOLUME_FLUX_SURFACE	Mesh surface	5
---------------------	--------------	---

1) This is computed by the AUI as a predefined resultant using the formula

$$\text{HEAT_FLUX_SURFACE} = <\text{HEAT_FLUX-X}> * <\text{SURFACE_NORMAL-X}> + \\ <\text{HEAT_FLUX-Y}> * <\text{SURFACE_NORMAL-Y}> + \\ <\text{HEAT_FLUX-Z}> * <\text{SURFACE_NORMAL-Z}>$$

See Table 12.1-17 for the definitions of the surface normal variables.

2) This is computed by the AUI as a predefined resultant using the formula

$$\text{KINETIC_ENERGY_FLUX_SURFACE} = \\ \text{KINETIC_ENERGY_DENSITY} * \text{VOLUME_FLUX_SURFACE}$$

3) This is computed by the AUI as a predefined resultant using the formula

$$\text{MASS_FLUX_SURFACE} = \text{ELEMENT_DENSITY} * \\ (<\text{X-VELOCITY}> * <\text{SURFACE_NORMAL-X}> + \\ <\text{Y-VELOCITY}> * <\text{SURFACE_NORMAL-Y}> + \\ <\text{Z-VELOCITY}> * <\text{SURFACE_NORMAL-Z}>)$$

See Table 12.1-17 for the definitions of the surface normal variables. You must define variable ELEMENT_DENSITY before using variable MASS_FLUX_SURFACE.

4) These are computed by the AUI as predefined resultants using the formulas

$$\text{MOMENTUM_FLUX-X_SURFACE} = \\ <\text{X-VELOCITY}> * <\text{MASS_FLUX_SURFACE}>$$

$$\text{MOMENTUM_FLUX-Y_SURFACE} = \\ <\text{Y-VELOCITY}> * <\text{MASS_FLUX_SURFACE}>$$

$$\text{MOMENTUM_FLUX-Z_SURFACE} = \\ <\text{Z-VELOCITY}> * <\text{MASS_FLUX_SURFACE}>$$

You must define variable ELEMENT_DENSITY before using these variables. See Table 12.1-17 for the definitions of the surface normal variables.

5) This is computed by the AUI as a predefined resultant using the formula

```
VOLUME_FLUX_SURFACE = <X-VELOCITY>*<SURFACE_NORMAL-X> +
<Y-VELOCITY>*<SURFACE_NORMAL-Y> +
<Z-VELOCITY>*<SURFACE_NORMAL-Z>
```

See Table 12.1-18 for the definitions of the surface normal variables.

- 6) Each of these variables is the square root of the sum of the squares of their components, as computed by the AUI.
-

Table 12.1-13: Electric field variables

Variable name	Variable location type	Notes
ELECTRIC_POTENTIAL	Node field	
CURRENT_DENSITY-(XYZ)	Element field	
CURRENT_DENSITY_MAGNITUDE	Element field	1
HEAT DISSIPATION	Element field	
HEAT DISSIPATION RATE	Element field	
TOTAL_HEAT DISSIPATION	Loc-independent	
TOTAL_HEAT DISSIPATION RATE	Loc-independent	

- 1) CURRENT_DENSITY_MAGNITUDE is the square root of the sum of the squares of the current density components, as computed by the AUI.
-

Table 12.1-14: Fracture variables

Variable name	Variable location type	Notes
CRACKED_SURFACE	Virtual shift	1
CRACK_BOUNDARY_FLAG	Node discrete	2
CRACK_LENGTH	Loc-independent	
DYNAMIC_CORRECTION	Virtual shift	3
J-PARAMETER_1	Virtual shift	3,4
J-PARAMETER_2	Virtual shift	3,5
J-PARAMETER_3	Virtual shift	3,6
LINE_CONTOUR_NUMBER	Line contour	7
LINE_J-INTEGRAL_AVERAGE	Line contour	7,8

LINE_J-INTEGRAL_DIFFERENCE	Line contour	7
PRESSURE_CORRECTION	Virtual shift	3
TEMPERATURE_CORRECTION	Virtual shift	3
VIRTUAL_CRACK_SURFACE	Virtual shift	3,9
VIRTUAL_SHIFT_NUMBER	Virtual shift	3
VIRTUAL_SHIFT_*_PART_FACTOR	Node field	3,10
(XYZ)-VIRTUAL_SHIFT	Virtual shift	3,11

1) This variable is used to plot the cracked surface in 3-D crack propagation analysis (ADINA fracture mechanics analysis). It is equal to 1.0 if the node is on the crack front or has been released; otherwise it is not defined. This variable can be plotted in a band plot to shade the area that has cracked during the analysis.

2) This variable indicates the current boundary condition applied to a node on a crack propagation surface (used in ADINA fracture mechanics analysis). It has the following values: 0=node is released, 1=node is on the crack front and is fixed, 2=node is on the crack front and is constrained, 11=node is unreleased and is fixed, 12=node is unreleased and is constrained.

The crack boundary flag is not defined on nodes that are not on the crack propagation surface.

3) These variables are used in ADINA fracture mechanics analysis, virtual crack extension method, see Section 10.3.

4) This value of the J-integral is not recommended for general use.

5) This value of the J-integral does not include temperature, pressure or dynamic corrections.

6) This value of the J-integral includes temperature, pressure and dynamic corrections.

7) These variables are used in ADINA fracture mechanics analysis, line contour method.

8) This is the average of the J-integrals for the outer and inner paths within elements that use 2×2 Gauss integration.

9) This is the crack surface area.

10) This variable gives, for each node, the amount of participation in the virtual shift,

from "undefined" (node is not included in virtual shift) to 0.0 (node is included, but does not shift) to 1.0 (node is included and shifts completely). It can be used in a band plot to show the virtual shift.

- 11) These are the components of the virtual shift vector.

Table 12.1-15: Failure criterion variables

Variable name	Variable location type	Notes
CAP_LOCATION	Element field	
CRACK_FLAG	Element field	1
EFFECTIVE_STRESS_RATIO	Element field	
FAILURE_CRITERION	Element field	2,3
FAILURE_CRITERION_COMP-FIBER	Element field	2,3
FAILURE_CRITERION_COMP-MATRIX	Element field	2,3
FAILURE_CRITERION_SURFACE- (1234)	Element field	2,3
FAILURE_CRITERION_TENS-FIBER	Element field	2,3
FAILURE_CRITERION_TENS-MATRIX	Element field	2,3
FAILURE_FLAG	Element field	2,4
FAILURE_FLAG_COMP-FIBER	Element field	2,5
FAILURE_FLAG_COMP-MATRIX	Element field	2,5
FAILURE_FLAG_COMPRESSION- (ABC)	Element field	2,5
FAILURE_FLAG_SHEAR- (AB, AC, BC)	Element field	2,5
FAILURE_FLAG_SURFACE- (1234)	Element field	2,5
FAILURE_FLAG_TENS-FIBER	Element field	2,5
FAILURE_FLAG_TENS-MATRIX	Element field	2,5
FAILURE_FLAG_TENSION- (ABC)	Element field	2,5
GASKET_DEFORMATION_MODE	Element field	11
GASKET_STATUS	Element field	12
NUMBER_OF_CRACKS	Element field	6
PLASTIC_FLAG	Element field	7
PLASTIC_FLAG_2	Element field	8
PLASTIC_FLAG_AXIAL	Section field	
PLASTIC_FLAG_BENDING- (ST)	Section field	
PLASTIC_FLAG_TORSION	Section field	
SMA_FLAG	Element field	13
SPECIFIC_VOLUME	Element field	

<u>VOID_RATIO</u>	Element field	
<u>VOID_VOLUME_FRACTION</u>	Element field	
<u>WRINKLE_FLAG</u>	Element field	9
<u>YIELD_FUNCTION</u>	Element field	10
<u>YIELD_SURFACE_DIAMETER_P</u>	Element field	

1) This is a three digit number (P1)(P2)(P3) interpreted as follows. Each digit represents a principal stress coordinate direction. If the digit equals 0 (zero), there is no crack; if the digit equals 1, there is an open crack; if the digit equals 2, there is a closed crack. If the crack flag equals 333, the material is crushed.

The related variable NUMBER_OF_CRACKS returns the number of cracks at the point, as computed from this variable.

2) The failure variables are used in analyses employing the anisotropic failure criteria within shell element groups, see Section 2.7.8.

3) Each failure criterion variable stores the value of the failure criterion used in the analysis.

4) The failure flag has the following values: 0=material at the point considered has not failed, 1=material at the point considered has failed.

5) All of the failure flags with a qualifier have the following values: 0=material at the point considered has not failed in that mode, 1=material at the point considered has failed in that mode.

6) This variable returns the number of cracks at the point, as computed by the AUI from the variable CRACK_FLAG.

7) The plastic flag has the following values: 1=elastic conditions, 2=plastic conditions.

8) This variable is used for the Drucker-Prager material model. It has the following values: 1=elastic conditions, 2=Drucker-Prager yielding, 3=cap yielding, 4=vertex yielding, 5=tension cutoff yielding.

9) This variable has the following values: 0=no wrinkling, 1=wrinkling in one direction, 2=wrinkling in two directions.

10) This variable is the value of the yield function.

11) GASKET_DEFORMATION_MODE has the following values: 1=elastic, 2=plastic, 3=rupture.

12) GASKET_STATUS has the following values: 1=open, 2=closed, 3=sealed, 4=leaked, 5=crushed.

13) SMA_FLAG is used with the SMA material. It has the following values: 1=elastic state; 2=martensite transformation; 3=austenite transformation.

Table 12.1-16: Eigenvector variables

Variable name	Variable location type	Notes
(ABC) -EIGENVECTOR	Node field	
(XYZ) -EIGENVECTOR	Node field	
(ABC) -EIGENVECTOR_ROTATION	Node field	
(V1, V2) -EIGENVECTOR_ROTATION	Node field	
(XYZ) -EIGENVECTOR_ROTATION	Node field	
EIGENVECTOR_MAGNITUDE	Node field	1
FLUID_P0_EIGENVECTOR	Node field	
FLUID_POTENTIAL_EIGENVECTOR	Node field	
OVALIZATION_EIGENVECTOR-(123456)	Node field	
PORE_PRESSURE_EIGENVECTOR	Node field	
THERMAL_EIGENVECTOR	Node field	
WARPING_EIGENVECTOR-(123456)	Node field	

1) EIGENVECTOR_MAGNITUDE is the square root of the sum of the squares of the eigenvector components, as computed by the AUI.

Table 12.1-17: Prescribed load variables

Variable name	Variable location type	Notes
(ABC) -PRESCRIBED_ACCELERATION	Node field	
(XYZ) -PRESCRIBED_ACCELERATION	Node field	
(ABC) -PRESCRIBED_ANG_ACCELERATION	Node field	
(V1,V2) -PRESCRIBED_ANG_ACCELERATION	Node field	
(XYZ) -PRESCRIBED_ANG_ACCELERATION	Node field	
PRESCRIBED_CONC_HEAT_FLOW	Node field	
PRESCRIBED_CONC_SEEPAGE_FLOW	Node field	
PRESCRIBED_CONV_TEMPERATURE	Node field	
(ABC) -PRESCRIBED_DISPLACEMENT	Node field	
(XYZ) -PRESCRIBED_DISPLACEMENT	Node field	
(ABC) -PRESCRIBED_FLUID_TRACTION	Node field	
(XYZ) -PRESCRIBED_FLUID_TRACTION	Node field	
PRESCRIBED_FLUID_P0	Node field	
PRESCRIBED_FLUID_POTENTIAL	Node field	
(ABC) -PRESCRIBED_FORCE	Node field	
(XYZ) -PRESCRIBED_FORCE	Node field	
PRESCRIBED_MASS_RATIO_*	Node field	
(ABC) -PRESCRIBED_MOMENT	Node field	
(V1,V2) -PRESCRIBED_MOMENT	Node field	
(XYZ) -PRESCRIBED_MOMENT	Node field	
PRESCRIBED_OVALIZATION- (123456)	Node field	
PRESCRIBED_PRESSURE	Node field	
PRESCRIBED_RAD_TEMPERATURE	Node field	
(ABC) -PRESCRIBED_ROTATION	Node field	
(V1,V2) -PRESCRIBED_ROTATION	Node field	
(XYZ) -PRESCRIBED_ROTATION	Node field	
PRESCRIBED_TEMPERATURE	Node field	
PRESCRIBED_TOTAL_HEAD	Node field	
(ABC) -PRESCRIBED_VELOCITY	Node field	
(XYZ) -PRESCRIBED_VELOCITY	Node field	
PRESCRIBED_WARPING- (123456)	Node field	

Table 12.1-18: Coordinate variables

Variable name	Variable location type	Notes
(XYZ) -COORDINATE	Coordinate	1
(XYZ) -COORDINATE_INCREMENT	Coordinate	2
(XYZ) -ORIGINAL	Coordinate	7
DIRECTOR_VECTOR-(XYZ)_CURRENT	Element local node field	9
DIRECTOR_VECTOR-(XYZ)_ORIGINAL	Element local node field	10
DISTANCE	Coordinate	3
DISTANCE_POSITION	Coordinate	3
(XYZ)-LEVER	Coordinate	4
(XYZ)-POSITION	Coordinate	5
SURFACE_NORMAL-(XYZ)	Coordinate	6
SURFACE_NORMAL-(XYZ)_ORIGINAL	Coordinate	8

- 1) These are the initial coordinates (coordinates of the original configuration of the model), when used with responses of type load-step. These are the reference coordinates (coordinates of the reference configuration), when used with responses of type mode-shape, response-spectrum, harmonic and random (recall that these responses include a reference time).
- 2) These give the changes in the X, Y and Z coordinates between the original configuration and the current configuration. They are used when the coordinates of the model change during the analysis, for example, in fracture mechanics crack propagation analysis.
- 3) These variables, when used in a line list or line graph, are defined as follows:

`DISTANCE = last computed DISTANCE + (distance between previous point and current point),`

`DISTANCE of first point = 0.0`

with a similar definition for `DISTANCE-POSITION`. (The difference between `DISTANCE` and `DISTANCE-POSITION` is that `DISTANCE` is computed based on the original configuration of the model and `DISTANCE-POSITION` is computed based on the deformed configuration of the model.) These variables are path dependent and can only be used in a line list or line graph.

- 4) The lever variables are defined as follows: $x^L = (x - x_{REF})^N$ with similar

definitions for the y and z lever variables. (x, y, z) are either the original or current coordinates, $(x_{REF}, y_{REF}, z_{REF})$ are the coordinates of a reference point and N is an integer greater than or equal to 1. The choice of original or current coordinate, $(x_{REF}, y_{REF}, z_{REF})$ and N are entered as part of a result control depiction.

The purpose of the lever variables is to make it easier to compute moments about a reference point.

5) These are the current coordinates (coordinates of the current configuration of the model), when used with responses of type load-step. These are the reference coordinates (coordinates of the reference configuration), when used with responses of type mode-shape, response-spectrum, harmonic and random (recall that these responses include a reference time).

6) These variables are used to access the face outwards normal of mesh integration or mesh extreme points.

These variables can also be used when plotting bands (for example, a resultant can contain these variables when plotting bands). When plotting bands, these variables refer to the normal at the current time (when the response is of type load-step), or at the reference time (when the response is of type mode-shape, response-spectrum, harmonic or random).

7) These are the initial coordinates (coordinates of the original configuration of the model), when used with any response type.

8) These variables can be used when plotting bands (for example, a resultant can contain these variables), except when plotting bands on cutting surfaces. When plotting bands, these variables refer to the normal of the original configuration of the model.

9) DIRECTOR_VECTOR-(XYZ)_CURRENT are the current components of the shell nodes director vectors. These variables are available from ADINA at the local nodes of shell elements.

10) DIRECTOR_VECTOR-(XYZ)_ORIGINAL are the original components of the shell nodes director vectors. These variables are available from ADINA at the local nodes of shell elements.

Table 12.1-19: Thickness variables

Variable name	Variable location type	Notes
THICKNESS	Element local node field	1, 2
THICKNESS_CHANGE	Element local node field	1, 3
THICKNESS_CURRENT	Element local node field	1, 4
THICKNESS_ORIGINAL	Element local node field	1, 2
THICKNESS_STRAIN	Element local node field	1
THINNING	Element local node field	1

- 1) These variables are available from ADINA at the local nodes of ADINA shell elements.
- 2) THICKNESS and THICKNESS_ORIGINAL are the shell thicknesses in the original configuration of the model.
- 3) THICKNESS_CHANGE = THICKNESS_CURRENT - THICKNESS_ORIGINAL.
- 4) THICKNESS_CURRENT is the shell thickness in the current configuration. THICKNESS_CURRENT is different than THICKNESS and THICKNESS_ORIGINAL only for the top-bottom nodes of transition shell elements and for large strain shell elements.
- 5) THICKNESS_STRAIN is the natural logarithm of THICKNESS_CURRENT divided by THICKNESS_ORIGINAL.
- 6) THINNING=-(THICKNESS_CHANGE/THICKNESS_ORIGINAL).

Table 12.1-20: Time variables

Variable name	Variable location type	Notes
LAMBDA	Loc-independent	1
LOAD_STEP	Loc-independent	
TIME	Loc-independent	
TIME_FUNCTION *	Loc-independent	

- 1) LAMBDA is the load vector multiplier in the LDC method.

Table 12.1-21: Frequency/mode variables

Variable name	Variable location type	Notes
BUCKLING_LOAD_FACTOR	Loc-independent	1
EIGENVALUE	Loc-independent	2
FREQUENCY	Loc-independent	3
(XYZ)-MODAL_PARTICIPATION_FACTOR	Loc-independent	4
MODE_NUMBER	Loc-independent	
NATURAL_FREQUENCY	Loc-independent	5
PHYSICAL_ERROR_BOUND	Loc-independent	6

- 1) This is the buckling load factor computed in linearized buckling analysis.
- 2) This is the eigenvalue as computed by ADINA. The natural frequency is the square root of the eigenvalue.
- 3) The dimensions of frequency are cycles/(unit time)
- 4) These variables refer to modal participation factors for ground motions acting in the X, Y and Z directions.
- 5) The dimensions of natural frequency are radians/(unit time).
- 6) The physical error bound is used to measure the accuracy of the computed eigenvalues, see Section 6.2.

Table 12.1-22: Miscellaneous variables

Variable name	Variable location type	Notes
CENTER_OF_MASS-(XYZ)	Loc-independent	1,2
CENTROID-(XYZ)	Loc-independent	1,3
CONTACT_STATE_FLAG	Node discrete	4
COORDINATE_SYSTEM_NUMBER	Loc-independent	5
FLUID_DOF_TYPE	Node discrete	6
INERTIA-P1	Loc-independent	1,7
INERTIA-P1_DIRECTION-(XYZ)	Loc-independent	1,8
INERTIA-P2	Loc-independent	1,9
INERTIA-P2_DIRECTION-(XYZ)	Loc-independent	1,10
INERTIA-P3	Loc-independent	1,11

INERTIA-P3_DIRECTION-(XYZ)	Loc-independent	1,12
KINETIC_ENERGY_DENSITY	Node field	13
LINE_INDEX_NUMBER	Loc-independent	14
MASS	Loc-independent	1,15
MOMENT_OF_INERTIA-(XX,YY,ZZ)	Loc-independent	1,16
NODAL_CONTACT_GAP	Node field	26
NODAL_CONTACT_STATUS	Node field	27
NORMAL-(XYZ)	Contact	
NUMBER_OF_SUBINCREMENTS	Element field	
PIPE_CROSS-SECTION_AREA	Node field	17
PIPE_SKIN_AREA	Node field	18
PRODUCT_OF_INERTIA-(XY,XZ,YZ)	Loc-independent	1,19
RADIOSITY	Node field	
REACTION_MOMENT-(XYZ)	Node discrete	25
ROTATIONAL_SYSTEM	Node discrete	20
SEGMENT_AREA	Contact	
SEGMENT_LENGTH	Contact	
STRAIN_ENERGY_DENSITY	Element field	23
SURFACE_MOMENT-(XYZ)	Mesh surface	24
TRANSLATIONAL_SYSTEM	Node discrete	21
VOLUME	Loc-independent	1,22

- 1) These variables are available for ADINA models when mass properties calculations are requested in ADINA-IN for all element groups in the model. They represent the given properties for the entire structure (or for that part of the structure selected in the AUI) and are therefore location-dependent.
- 2) This is the center of mass in the global coordinate system.
- 3) This is the centroid (center of volume) in the global coordinate system.
- 4) The contact state flag has the following values: 0=unknown, 1=free, 2=sticking, 3=sliding. This variable is defined only for node to node contact.
- 5) This variable stores the number of the coordinate system into which stresses/strains are transformed, see Section 12.1.3.
- 6) The fluid dof type has the following values: 0=free velocity potential, 1=fixed, 2=constrained, 3=free velocity potential, 4=free P_0 degree of freedom.

7) This is the largest principal inertia, given in a coordinate system with origin at the center of mass and axes parallel to the global coordinate system.

8) This is the principal axis associated with the largest principal inertia.

9) This is the intermediate principal inertia, given in a coordinate system with origin at the center of mass and axes parallel to the global coordinate system.

10) This is the principal axis associated with the intermediate principal inertia.

11) This is the smallest principal inertia, given in a coordinate system with origin at the center of mass and axes parallel to the global coordinate system.

12) This is the principal axis associated with the smallest principal inertia.

13) This is computed by the AUI as a predefined resultant using the formula

```
KINETIC_ENERGY_DENSITY = 0.5*ELEMENT_DENSITY*
(<X-VELOCITY>**2 + <Y-VELOCITY>**2 + <Z-VELOCITY>**2)
```

You must define variable ELEMENT_DENSITY before using variable

KINETIC_ENERGY_DENSITY.

14) This is used when listing or graphing results along a line.

15) This is the total mass.

16) These are the moments of inertia, given in a coordinate system with origin at the center of mass and axes parallel to the global coordinate system. The XX component of the inertia tensor is equal to the XX moment of inertia.

17) This is the sum (over all pipe elements attached to the node) of the pipe internal cross-sectional area, based upon the current geometry.

18) This is the sum (over all pipe elements attached to the node) of the internal pipe skin area, based upon the current geometry.

19) These are the products of inertia, given in a coordinate system with origin at the center of mass and axes parallel to the global coordinate system. The XY component of the inertia tensor is the negative of the XY product of inertia.

20) Rotational system has the following values: 0=rotational results are stored in the

global coordinate system, 1=rotational results are stored in a skew system, 2=rotational results are stored in a shell midsurface coordinate system (the shell node has five degrees of freedom).

21) Translational system has the following values: 0=translational results are stored in the global coordinate system, 1=translational results are stored in a skew system.

22) This is the total volume.

23) This variable is defined only in ADINA fracture mechanics analysis.

24) These are computed by the AUI as predefined resultants using the formulas

SURFACE_MOMENT-X =

$$\langle Y\text{-LEVER} \rangle * \langle SURFACE_TRACTION-Z \rangle - \langle Z\text{-LEVER} \rangle * \langle SURFACE_TRACTION-Y \rangle$$

SURFACE_MOMENT-Y =

$$\langle Z\text{-LEVER} \rangle * \langle SURFACE_TRACTION-X \rangle - \langle X\text{-LEVER} \rangle * \langle SURFACE_TRACTION-Z \rangle$$

SURFACE_MOMENT-Z =

$$\langle X\text{-LEVER} \rangle * \langle SURFACE_TRACTION-Y \rangle - \langle Y\text{-LEVER} \rangle * \langle SURFACE_TRACTION-X \rangle$$

26) NODAL_CONTACT_GAP is output when the “new” contact segments are used.

27) NODAL_CONTACT_STATUS is output when the “new” contact segments are used. It has the following values: =1, node is dead (on a dead contact segment), =2, node is open, =3, node is closed, frictionless contact, =4, node is closed, slipping contact, =5, node is closed, sticking contact.

See Table 12.1-18 for the definitions of the lever variables and Table 12.1-10 for the definitions of the traction variables.

25) These are computed by the AUI as predefined resultants using the formulas

REACTION_MOMENT-X =

$$\langle Y\text{-LEVER} \rangle * \langle Z\text{-REACTION} \rangle - \langle Z\text{-LEVER} \rangle * \langle Y\text{-REACTION} \rangle$$

REACTION_MOMENT-Y =

$$\langle Z\text{-LEVER} \rangle * \langle X\text{-REACTION} \rangle - \langle X\text{-LEVER} \rangle * \langle Z\text{-REACTION} \rangle$$

REACTION_MOMENT-Z =

$$\langle X\text{-LEVER} \rangle * \langle Y\text{-REACTION} \rangle - \langle Y\text{-LEVER} \rangle * \langle X\text{-REACTION} \rangle$$

See Table 12.1-18 for the definitions of the lever variables and Table 12.1-11 for the definitions of the reaction variables.

Table 12.1-23: User-defined variables

Variable name	Variable location type	Notes
INT_USER_VARIABLE_*	Element field	1
ONE	Loc-independent	2
USER_VARIABLE *	Element field	1

- 1) These variables are used in conjunction with the user-supplied material model.
 - 2) This variable has the constant value “1”. It is included so that you can determine the total area of a domain that is defined as a mesh integration point.
 - 3) Resultants that you define in the AUI are added to the list of user-defined variables. The variable location type of a resultant is determined by the variables that define the resultant.
-

- The AUI can transform node field and node discrete results calculated by ADINA in one coordinate system to another coordinate system as shown in Table 12.1-24.

Table 12.1-24: Coordinate transformations for node field and node discrete variables performed by the AUI

Coordinate system of ADINA/ADINA-F result	Variable prefix		
	X,Y,Z	A,B,C	V1,V2
	1	1	3
Global	1	1	3
Skew	2	1	3
V1,V2	3	1	1

- 1) The result is computed without any coordinate transformation.
 - 2) The result is computed using a transformation from the skew
-

system to the global system.

- 3) No result is computed.

Stress transformations: The AUI can transform stresses and strains to user-defined coordinate systems, including cylindrical and spherical systems.

In order to activate this feature, you set the coordinate system of the result control dialog box to the number of the coordinate system that you define. By default, the coordinate system is 0 (the global Cartesian system).

You access the transformed stresses using variables STRESS-11, STRESS-22, STRESS-33, STRESS-12, STRESS-13, STRESS-23 and the transformed strains using variables STRAIN-11, STRAIN-22, STRAIN-33, STRAIN-12, STRAIN-13, STRAIN-23. These variables can be used in any plotting or listing command.

The AUI evaluates these variables from the stresses and strains in the global system, as computed by the solution program. Therefore the stresses and strains must have been calculated in the global system by the solution program.

The direction numbers in the variable names STRESS-11, STRESS-22, etc. correspond to coordinate directions as shown in the following table:

System type	Direction 1	Direction 2	Direction 3
Cartesian	XL	YL	ZL
Cylindrical	R	THETA	XL
Spherical	R	THETA	PHI

Thus, by default, STRESS-11 = STRESS-XX, STRESS-22 = STRESS-YY, etc.

You can plot the directions used for stress transformations in a mesh plot. This is done from within the Element Depiction dialog box reached using the Modify Meshplot icon. This feature is useful for verifying that the stress transformation directions are correct.

Please note:

You can smooth the transformed stresses. The smoothing is done in the global coordinate system and the smoothed stresses are transformed into the user-specified system.

The transformed off-diagonal strain components (STRAIN-12, STRAIN-13, STRAIN-23) are engineering quantities, not tensorial quantities.

You can use the variables STRESS-11, STRESS-22, etc. within resultants.

12.1.2 Where results can be obtained

- Results can be evaluated at various types of locations within the model.

Location-independent: Used for variables that are location-independent, see above.

Node: locations at nodes.

Element/layer: locations within elements or element layers. These can be the integration points, the element local nodes or other locations within the element.

To specify the element, you need to enter the element group number and element number. If the element is an ADINA multilayer shell element, you also need to enter the layer number. Otherwise you can enter a layer number of 1.

To specify the point within the element (or element layer), there are several conventions, as follows. In ADINA system elements, there are two subtypes of element points: integration points and local nodes. In any given analysis, for any given element, the results are computed by the solution program at only one of these subtypes.

In the AUI, you can request that the results be evaluated at the element points where they were computed by the solution program, or you can request that the results be extrapolated to other element points. This latter option is available for some elements as shown in Table 12.1-25.

Table 12.1-25 Extrapolation of element results
ADINA

Element type	Result location	Dimension for results	Can extrapolate to
Truss	Int. pts.	1-D	$-1.0 \leq r \leq 1.0$

2-D solid or fluid	Int. pts.	2-D	$-1.0 \leq r,s \leq 1.0^*$
3-D solid or fluid	Int. pts.	3-D	$-1.0 \leq r,s,t \leq 1.0^*$
Iso-beam (3-D)	Int. pts.	3-D	$-1.0 \leq r,s,t \leq 1.0$
Iso-beam (2-D)	Int. pts.	2-D	$-1.0 \leq r,s \leq 1.0$
Plate	Int. pts.	2-D	$0.0 \leq r,s \leq 1.0$
Shell	Int. pts.	3-D	$-1.0 \leq r,s,t \leq 1.0$

ADINA-T

Element type	Result location	Dimension for results	Can extrapolate to
1-D	Int. pts.	1-D	$-1.0 \leq r \leq 1.0$
2-D conduction	Int. pts.	2-D	$-1.0 \leq r,s \leq 1.0^*$
3-D conduction	Int. pts.	3-D	$-1.0 \leq r,s,t \leq 1.0^*$
Shell conduction	Int. pts.	3-D	$-1.0 \leq r,s,t \leq 1.0$

ADINA-F:

Element type	Result location	Dimension for results	Can extrapolate to
2-D fluid	Int. pts.	2-D	$-1.0 \leq r,s \leq 1.0^*$
3-D fluid	Int. pts.	3-D	$-1.0 \leq r,s,t \leq 1.0^*$

*) If the element is triangular or tetrahedral, the element uses triangular/tetrahedral area coordinates as the isoparametric coordinates and the coordinates range from 0.0 to 1.0.
(end of table)

You use the Defined By parameter and its related parameters in the element result point dialog box to select the element point as follows:

Grid point: This option is available only if results were calculated at the element integration points by the solution program.

The element conceptually contains an evenly spaced grid in the elements' isoparametric system. The grid can have one, two or three dimensions depending on the element type.

To specify a grid point using a 1-D grid, give a two digit number AB: digit A gives the total number of grid divisions and digit B gives the desired grid point.

The isoparametric coordinate corresponding to the grid point is computed using

$$C = C_{\min} + \frac{B-1}{A-1}(C_{\max} - C_{\min})$$

where C_{\min} and C_{\max} are the minimum and maximum isoparametric coordinates that the elements can contain. For example, AB=52 corresponds to $r = -1/2$ for elements in which r can be between -1 and 1.

To specify a grid point using a 2-D grid, give a four digit number ABCD: digits A and B give the desired grid point in the r direction and digits C and D give the desired grid point in the s direction. For example, ABCD=2121 corresponds to $r = -1$, $s = -1$ and ABCD=3243 corresponds to $r = 0$, $s = 1/3$.

To specify a grid point using a 3-D grid, give a six digit number ABCDEF: digits A and B give the desired grid point in the r direction, digits C and D give the desired grid point in the s direction and digits E and F give the desired grid point in the t direction.

You can also specify the element centroidal point by giving a grid value of 0.

Label point: You specify the label point in the element at which results were computed by the solution program.

If the results were calculated at the integration points, then the label is a 1, 2, 3 or 4 digit number giving the integration point as shown in Table 12.1-26.

If the results were calculated at the element local nodes, then the label is the local node number.

No extrapolation is performed by the AUI when using the label option.

Table 12.1-26: Label point numbering

ADINA:

Element type	Result locations	Label number
Truss	Int. pts.	(INR)
2-D solid or fluid	Int. pts.	10(INR) + INS*
3-D solid or fluid	Int. pts.	100(INR) + 10(INS) + INT*
Hermitian beam	Int. pts.	100(INR) + 10(INS) + INT
Iso-beam (3-D)	Int. pts.	100(INR) + 10(INS) + INT
Iso-beam (2-D)	Int. pts.	10(INR) + INS
Plate	Int. pts.	(ITRI)
Shell (rectangular)	Int. pts.	100(INR) + 10(INS) + INT

Shell (triangular)	Int. pts.	10(ITRI) + (INT)
Pipe	Int. pts.	1000(INA) + 10(INB) + INC
Spring	---	1
General	Stress transformation matrix	(J)

ADINA-T:

Element type	Result locations	Label number
1-D	Int. pts.	(INR)
2-D conduction	Int. pts.	10(INR) + INS*
3-D conduction	Int. pts.	100(INR) + 10(INS) + INT*
Shell (conduction)	Int. pts.	100(INR) + 10(INS) + INT

ADINA-F:

Element type	Result locations	Label number
2-D fluid	Int. pts.	10(INR) + INS*
3-D fluid	Int. pts.	100(INR) + 10(INS) + INT*

Notes:

Each quantity in parentheses represents one integer:

(ITRI) – integration point number in triangular integration

(ITET) - integration point number in tetrahedral integration

(INR), (INS), (INT) – integration point numbers for r, s and t coordinates

(INA), (INB), (INC) – integration point numbers for a, b and c coordinates

(J) – stress component number

*) If the element is triangular, the label number is (ITRI); if the element is tetrahedral, the label number is (ITET).

(end of table)

Node: This option is available only if results were calculated at the element integration points by the solution program.

You enter the global node number of a node connected to the element.

The results are extrapolated to the point within the element

corresponding to the position of the node in the element. For example, if the first node of a 2-D element is number 65, then selecting node 65 is equivalent to selecting the point $(r,s) = (1,1)$ in the element.

It is necessary to specify the element number when this option is used because, when the results are unsmoothed, the results at a node point are different from the different elements attached to the node.

The use of this option does not, by itself, smooth results from neighbor-ring elements to the specified node. But if smoothing is turned on, then any element that contains the specified node will give the same results (because the smoothed results are continuous between the elements).

Isoparametric coordinates: This option is available only if results were calculated at the element integration points by the solution program.

You directly enter the r, s and t coordinates of the desired point. If the element is 1-D or 2-D, the unused coordinates are ignored.

Special considerations for the ADINA shell element: The element is considered to be a 3-D element for the purpose of specifying element points. You can specify that the element point lies on the shell midsurface using the following options:

Node number: Specify the node number of a midsurface node.

Grid location: Specify a grid location of 0, or a grid location in which the 5th and 6th digits are equal to 1.

When the element point lies on the shell midsurface, when the results extrapolated from the element integration points are evaluated at the point, the t coordinate and layer used for the evaluation are selected by the parameters of the result control depiction.

Element section: locations within element sections (ADINA only, beam and shell elements). These can be the section integration points, the element local nodes or other locations within the element section.

Two types of elements can have results output at element sections. The Hermitian beam element can have results output on the neutral line of the element; the neutral line is considered to be the element section. The ADINA shell element can have results output on the neutral surface of the element; the neutral surface is considered to be the element section.

Element section results are very similar to element result points. You can request that results be evaluated at the locations at which they were computed by ADINA or that the results be extrapolated to other locations using the grid, node and isoparametric coordinate parameters. It is important to remember the correct dimension of the element section: the section is a line for beam elements and a surface for shell elements. Hence, for example, a grid location is of the form AB for beam element section points and of the form ABCD for shell element section points.

Note that it is possible to list shell element stresses at shell element section result points if shell midsurface calculations are activated in the result control depiction.

Contact segments: (ADINA only). Locations in contact segments. A contact surface contains one or contact segments, each of which contains a single location where ADINA computes results.

When you request the coordinates of a contact segment, the coordinates of the center of the segment are used.

Within a contact surface, contact segments are numbered sequentially starting from 1. Hence, to specify a contact segment result point location, you must specify three numbers: the contact group number, the contact surface number and the contact segment number.

Note that no results are available for contact segments on target contact surfaces. Those contact results that are output at nodal points (for example, concentrated contact forces) are considered by the AUI to be associated with node points.

Draw bead segments: (ADINA only). Locations in draw bead segments. A contact surface optionally contains one or more drawbead segments, each of which contains a single location where ADINA computes results.

Within a contact surface, draw bead segments are numbered sequentially starting from 1. Hence, to specify a draw bead segment result point location, you must specify three numbers: the

contact group number, the contact surface number and the draw bead segment number.

Radiosity segments: (ADINA-T only). Locations in radiosity segments. A radiosity surface contains one or more radiosity segments, each of which contains integration points at which ADINA-T computes results.

The options for entering locations within radiosity segments are very similar to those used for entering locations within element layers. You can request that results be evaluated at the locations where they were computed by ADINA-T or that the results be extrapolated to other locations within the segment using the grid, node and isoparametric coordinate parameters. It is important to remember the correct dimensions of the radiosity segment. Hence, for example, a grid location is of the form AB for segments within radiosity lines and of the form ABCD for segments within radiosity surfaces.

Virtual shifts: (ADINA only). In order for you to access the results associated with virtual shifts in ADINA fracture mechanics analysis, the AUI allows you to define a result point at a virtual shift. (In this case, the virtual shift does not occupy a point in space.)

Line contours: (ADINA only). In order for you to access the results associated with J-integral line contours in ADINA fracture mechanics analysis, the AUI allows you to define a result point at a J-integral line contour. (In this case, the line contour does not occupy a point in space.)

Mesh integration points: If you want to integrate a quantity over the faces of a mesh plot, define a mesh integration point.

The mesh integration point consists of: 1) a domain over which the integration is to take place and 2) the rules for performing the integration.

Domain: The domain is described by the name of a mesh plot and the name of a zone as follows. If the mesh plot does not have a cutting surface, the domain is the external faces of the meshplot (the “skin” of the meshplot). If the mesh plot has a cutting surface, the domain is the cutting surface faces.

Faces can be, for example, the boundaries of 3-D elements, the midsurfaces of shell or plate elements, or the interiors of 2-D elements. They cannot be the boundaries of shell or 2-D elements.

Another way to think of the domain is in terms of a band plot on the mesh plot. The domain is that area of the mesh plot that is covered by the bands.

Use the zone name to restrict the domain. A face is in the domain only if its associated element is in the specified zone.

Rules: The rules are the integration type, a multiplying factor, an integration order and the configuration over which the integration is performed. Let z be the integrand and S be the domain. Then the integration types are

Integral:
$$\int_S z dS$$

Averaged:
$$z_{av} = \frac{\int_S z dS}{S}$$

Mean-square:
$$z_{ms} = \frac{\int_S z^2 dS}{S}$$

Root-mean-square:
$$\sqrt{\frac{\int_S z^2 dS}{S}}$$

Variance:
$$\frac{\int_S (z - z_{av})^2 dS}{S}$$

Standard deviation:
$$\sqrt{\frac{\int_S (z - z_{av})^2 dS}{S}}$$

$$\text{Relative variance: } \frac{\int_S \left(\frac{z - z_{av}}{z_{av}} \right)^2 dS}{S}$$

$$\text{Relative standard deviation: } \sqrt{\frac{\int_S \left(\frac{z - z_{av}}{z_{av}} \right)^2 dS}{S}} \quad (\text{also referred to as the coefficient of variation})$$

Use the multiplying factor to multiply the result after the indicated integration is performed. Use the integration order to indicate the numerical integration order over each element face. Use the configuration to indicate whether to perform the integration in the original configuration of the model or in the current deformed configuration.

Please note that you do not specify the integrand within the mesh integration point. You specify the integrand when you reference the mesh integration point within another dialog box.

Once you have defined a mesh integration point, you can use it within any command that accepts a model result point. For example, you can plot the time history of an integrated quantity using Graph→Response Curve (Model Point).

There are certain variables and predefined resultants that can only be used in conjunction with mesh integration (or mesh extreme) points. These are all based upon the face outwards normal. The face outwards normal can be accessed through variables SURFACE_NORMAL-X, SURFACE_NORMAL-Y, SURFACE_NORMAL-Z. Useful quantities for which predefined resultants are defined are VOLUME_FLUX_SURFACE, MASS_FLUX_SURFACE, SURFACE_TRACTION-X, SURFACE_TRACTION-Y, SURFACE_TRACTION-Z, MOMENTUM_FLUX-X_SURFACE, MOMENTUM_FLUX-Y_SURFACE, MOMENTUM_FLUX-Z_SURFACE, KINETIC_ENERGY_DENSITY, KINETIC_ENERGY_FLUX_SURFACE, HEAT_FLUX_SURFACE. Any of these variables can be used along with a mesh integration point. For example, you can determine the volume flux of flow through a pipe by defining a mesh integration point corresponding to a cutting surface, then

graphing or listing the variable VOLUME_FLUX at the mesh integration point.

You can also obtain the surface area of the mesh integration point by evaluating variable ONE at the point (variable ONE is defined to have the value 1.0).

You can also obtain the approximate volume of the mesh by use of the divergence theorem (also known as Gauss' theorem or Green's theorem), which is:

$$\int_V \nabla \cdot \mathbf{A} dV = \int_S \mathbf{A} \cdot d\mathbf{S}$$

Simply define resultant ADOTDS to be <X-COORDINATE>*<SURFACE_NORMAL-X> and evaluate it at a mesh integration point. This works because $\nabla \cdot (<X-COORDINATE>, 0, 0) = 1$.

Mesh extreme points: If you want to determine the extreme value of a quantity over the faces of a mesh plot, define a mesh extreme point.

The mesh extreme point consists of: 1) a domain over which the search for the extreme value is to take place and 2) the rules for determining the extreme value.

Domain: The domain is defined in exactly the same way as for a mesh integration point, see above.

Rules: The rules are the extreme type, the multiplying factor, the search grid order and the domain configuration. The extreme types can be absolute maximum, maximum or minimum. Use the multiplying factor to multiply the result after the extreme value is found. Use the search grid order to indicate the number of sampling points within each element face. Use the domain configuration to indicate whether to search for the extreme value using the original or current deformed configuration of the model.

Please note that you do not specify the quantity within the mesh extreme point. You specify the quantity when you reference the mesh extreme point within another dialog box.

Once you have defined a mesh extreme point, you can use it

within any command that accepts a model result point. For example, you can plot the time history of an extreme quantity using Graph→Response Curve (Model Point).

The variables that can be used for mesh integration points can also be used for mesh extreme points.

- Not all results can be obtained at all result locations. Table 12.1-27 shows the result location types for each type of available variable. Notice that certain combinations are allowed only when smoothing is requested (see below).

12.1.3 How results are evaluated (including smoothing)

- If a node field variable is requested within an element, the nodal shape functions are employed to interpolate the variable within the element.
- If an element/layer variable is requested within an element at a point, the variable is interpolated or extrapolated as follows:

RST interpolation: If the variable represents a floating-point value, the variable is interpolated or extrapolated using the nearest integration points and bilinear interpolation. If the variable represents an integer value, the value of the variable at the nearest integration point is used.

Table 12.1-27: Locations at which available variables can be evaluated

Variable location type	Result location types
Location-independent	All
Node field	Node, element/layer, contact segment, radiation segment
Element/layer field	Node (smoothing only), element/layer, section
Contact segment	Contact segment
Drawbead segment	Drawbead segment
Section field	Node (smoothing only), section
Radiation field	Node (smoothing only),

	radiosity segment
Line contour	Line contour
Virtual shift	Virtual shift
Node discrete	Node
Element local node	Element/layer (at local nodes only)
Element local node field	Element/layer, section
Coordinate	Node, element/layer, section, contact segment, radiation segment
Mesh surface	Mesh integration, mesh extreme

Face interpolation: The results at the center of the nearest element face are used.

Centroid interpolation: The results at the element centroid are used.

Integration point interpolation: The results at the nearest integration point are used.

The choice of interpolation is made within the result control dialog box. RST interpolation is the default.

Smoothing

- The element/layer field or section results can be smoothed (that is, made continuous between adjacent elements or element sections). The smoothing process is now described in detail for element results; the smoothing process is similar for element section results.

Smoothing consists of two steps:

1) Smoothing to the node points. For each node,

- Within each element attached to the node, the element integration point results are extrapolated to the node point. The extrapolation process is exactly the same as used when element integration point results are requested at other points

within the element.

- b) The contributions of each element are combined into a single result, as specified by the type of smoothing:

Averaged: averages

Minimum: minimum result is taken

Maximum: maximum result is taken

Difference: difference between the maximum and minimum result is taken

Extreme: extreme result (result furthest from 0.0) is taken

Error: Same as difference, except that only the element corner nodes are considered and the result is divided by a reference value that you enter (the default reference value is 1.0).

2) Extrapolation to the requested point. The extrapolation is exactly the same as used when nodal point results are requested within the elements. If the requested point is a nodal point, no action is taken in this step.

- Normally, all elements attached to a node contribute results to the node. However, you can choose to smooth results using only selected elements. This is done by specifying a zone name in the smoothing depiction. In this case, only elements within the zone contribute results to the node.

This option can be important when, for example, the model has several materials and you do not want to smooth stresses across the material interfaces.

- The following restrictions for smoothing are important to remember:

- ▶ Results from 2-D, 3-D (solid or fluid) and plate elements can be smoothed. Results from shell elements can be smoothed on the midsurface of the shell elements if shell element midsurface calculations are turned on in the result control depiction.

- ▶ You must have requested that results be saved at the element integration points during preprocessing.
 - ▶ Results that evaluate as integer values, such as the crack flag from the concrete model, can be smoothed into non-integer values, even when the non-integer values have no meaning.
 - ▶ Smoothing results across element groups may not be physically appropriate, for example, when the elements groups have different material properties.
 - ▶ Results are smoothed component-by-component without taking into account possible differences in coordinate systems between adjacent elements. This is especially important for results from shell elements.
- Control of smoothing is done by choosing a smoothing depiction, or by defining and using a smoothing depiction. By default, smoothing is turned off in the AUI.

Plotting results onto midsurfaces of shells: The AUI allows you to specify which shell layer and t coordinate value to use when plotting results onto shell midsurfaces. This allows you to plot, for example, the results from the top of the shells onto the shell midsurfaces. You can also choose to plot results evaluated at the shell midsurface onto the shell midsurface. These choices are made using a result control depiction. By default, results from the top of the shell are presented on the shell midsurface.

12.1.4 When (for which load step, etc.) are results available

- Results available from solution programs of the ADINA system are

Load step results: Results associated with load steps (time steps). Each load step has a unique load step number and solution time.

Mode shape results: Results associated with mode shapes. Each mode shape has two numbers that define it uniquely, the mode shape number and the solution time corresponding to the

solution from which the mode shapes were calculated.

Residual results: Results associated with residual (static correction) responses in response spectrum analysis (see Section 9.1 for theoretical background).

- Corresponding to these three types of results, the AUI has three response types:

Load-step: Allows you to reference results from a specific load step. A load step is referenced by its solution time.

If you enter a solution time between two solution times for which results are output by the solution program, the AUI automatically interpolates the results to the entered solution time. This is an example of a time transformation discussed in Section 12.1.

Mode-shape: Allows you to reference results from a specific mode shape. A mode shape is referenced by its mode shape number and solution time.

Residual: Allows you to reference residual results. Residual results are referenced by the ground motion direction and solution time corresponding to the solution for which the residual results were obtained.

- The AUI includes additional response types used in various analyses.

Response spectrum: A calculation method used for computing results in response spectrum analysis. You specify the modal combination method, loading spectrum, modal damping table and other information. See Section 9.1 for theoretical background.

Harmonic: A calculation method used for computing results in harmonic vibration analysis. You specify the desired harmonic response (maximum, phase angle, rms amplitude or signed amplitude), the loading frequency, the loading spectra, the damping table and other information. See Section 9.3 for theoretical background.

Random: A calculation method used for computing results in random vibration analysis. You specify the desired random response (rms amplitude or power-spectral-density), the loading frequency (for power-spectral-density only), the loading spectra, the damping table and other information. See Section 9.4 for theoretical background.

Response-combination: A calculation method for combining previously defined responses. These responses can be of any type except for response-combination and any mixture of the allowed response types can be used. For each previously defined response, you can specify a weighting factor and a combination method.

The AUI uses the following algorithm to evaluate the response-combination:

```
accum=0.0
For (each response in the response-combination) {
    value=(result for the response) * multiplying factor
    if (method = algebraic) {
        accum=accum + value
    }
    else if (method = abs) {
        accum=abs(accum) + abs(value)
    }
    else if (method = srss) {
        accum=sqrt(accum**2 + value**2)
    }
    else if (method = signed 1) {
        if (value >= 0.0) {
            accum=value + abs(accum)
        }
        else {
            accum=value - abs(accum)
        }
    }
    else if (method = signed 2) {
        if (value >= 0.0) {
            accum=accum + abs(value)
        }
        else {
            accum=accum - abs(value)
        }
    }
}
```

```
    }  
  }  
}
```

You might use methods signed 1 and signed 2, for example, when including a static load case along with the combined responses for different spatial directions in response spectrum analysis.

Envelope: At a point in the model, a given result (such as the value of a stress component) has a time history, and there is a solution time at which the result obtains an extreme value at that point. Of course, the magnitude of the extreme value and the corresponding solution time is different from point to point. The envelope of the result is a listing of all of the extreme values of the results; the listing gives the model points, the extreme values and the corresponding solution times.

To obtain an envelope listing in the AUI, define a response of type envelope. This response associates a response name with a calculation method in which, for each calculated value, several responses are evaluated and the most extreme one returned. The type of extreme can be minimum, maximum, absolute value of the maximum or difference between maximum and minimum. There are two options:

Range: The AUI evaluates the value for a range of load-steps and returns the extreme value. You can choose the start and end times, the intermediate steps to sample, and whether to interpolate results to steps for which results are not saved.

Selected: The AUI evaluates the value at each response specified in the data input lines of this command and returns the extreme value.

The AUI performs the calculation independently at each requested point in the model.

- The AUI includes dialog boxes for each of the response types. Each dialog box defines a response of the desired response type. You then use the response in another command when you want to plot or list the results using that response.

For response types response spectrum, harmonic, random, response combination, random or envelope, the AUI does not actually compute the response until the response is referenced by another command.

12.2 Evaluation of meshes used

- Once the solution with a given finite element mesh has been obtained, you may want to identify whether the mesh used has been fine enough. This can be achieved in the AUI by plotting bands of quantities that are the derivatives of the primary element variables.
 - For example, a band plot of pressures (or effective stress or any stress variable) can be produced. The unsmoothed or smoothed pressures can be displayed. Consider that unsmoothed pressures are plotted. If the stresses are to be continuous in the exact solution (which may not be the case when there are discontinuities in material properties), the pressure bands will be reasonably continuous if the mesh has been fine enough. The breaks in bands indicate the severity of the stress discontinuities.
- ref. T. Sussman and K.J. Bathe, "Studies of finite element procedures – stress band plots and the evaluation of finite element meshes," *Engineering Computations*, Vol. 3, No. 3, pp. 178-191, September 1986.
- The breaks in the bands are easily seen if the bands are drawn using two colors that alternate (so that the resulting plot looks like a photoelastic image). This option is available in the AUI when a band table of type repeating is requested.
 - Another error measure based on jumps in results between elements when too coarse a mesh has been used is the maximum difference between unsmoothed results at nodes. This error measure is obtained by using a smoothing depiction of type error; see Section 12.1.3 for a description of the smoothing depictions.
 - See the AUI Primer, Problem 13, for an example showing the evaluation of meshes using band plots.

Index

2

- 2-D solid elements, 15
 - 3-D plane stress, 15
 - axisymmetric, 15
 - element output, 25
 - formulations, 21
 - generalized plane strain, 15
 - mass matrices, 24
 - material models, 21
 - numerical integration, 22
 - plane strain, 15
 - plane stress, 15
 - recommendations for use, 30
 - triangular, 20
- 2^{nd} Piola-Kirchhoff stresses, 245

3

- 3-D solid elements, 34
 - element output, 47
 - formulation, 43
 - mass matrices, 47
 - material models, 42
 - numerical integration, 44
 - prisms, 41
 - pyramids, 41
 - recommendations for use, 53
 - sub-parametric, 40
 - tetrahedra, 41

5

- 5 degrees of freedom node, 118

6

- 6 degrees of freedom node, 120

A

- Acceleration variables, 706
- Accumulated effective plastic strain, 271
- Active column solver, 542
- ADINA System documentation, 4
- Almansi strains, 244
- Analysis zooming, 690
- Analytical rigid targets, 468
- Arrival time, 489
- Arruda-Boyce material model, 336
 - 3-D analysis, 336
 - axisymmetric analysis, 336
 - plane strain analysis, 336
 - plane stress analysis, 336
 - selection of material constants, 337
- ATS method, 487, 551, 564
 - contact surfaces, 551
 - dynamic analysis, 552
- Automatic step incrementation, 547
 - ATS method, 551
 - LDC method, 553
- Axisymmetric shell elements, 84, 87, 95
 - element output, 94, 96
 - formulations, 94
 - linear, 91
 - mass matrices, 96
 - material models, 94
 - nonlinear, 94
 - numerical integration, 89, 95

B

- Beam elements, 54
 - 2-D action, 65
 - 3-D action, 65
 - coefficient of thermal expansion, 63
 - cross-sections for, 59, 64, 65

-
- elastic stability analysis, 64
 - element output, 63, 64, 68
 - internal hinges, 56
 - large displacement elastic, 64
 - linear, 58
 - mass matrices, 83
 - moment curvature, 70
 - nonlinear elasto-plastic, 65
 - numerical integration, 68
 - off-centered, 56
 - rigid ends, 57
 - shear deformations, 66
 - stiffness matrix for, 58
 - warping effects, 62, 64, 67
 - Beam-tied elements, 80
 - Bernoulli-Euler beam theory, 55
 - BFGS method, 546, 549, 562
 - Bolt elements, 82, 166
 - Boundary conditions
 - displacement, 485
 - essential, 485
 - force, 485
 - moment, 485
 - natural, 485
 - C**
 - Cable elements, 7
 - Cam-clay material model, 380
 - initial stiffness, 384
 - initial stresses, 383
 - Cauchy stresses, 241, 245
 - Centers of mass
 - inertial properties, 681
 - Central difference method, 575
 - Centrifugal loads, 504, 505
 - Centroids
 - inertial properties, 681
 - Composite shell elements, 127
 - specification of layer thicknesses, 128
 - transverse shear stresses, 127
 - Computers supported by ADINA, 2
 - Concentrated dampers, 573
 - Concentrated loads, 490
 - Concrete material model, 309, 310, 430
 - compressive failure envelopes, 323
 - material behavior after failure, 324
 - material failure envelopes, 316
 - Poisson's ratio in the compressive region, 327
 - post tensile cracking behavior, 324
 - postcompression crushing behavior, 326
 - stress-strain relations, 311
 - temperature effects, 327
 - tensile failure envelopes, 320
 - Condensed node, 656
 - Consistent contact surface stiffness, 459
 - Consolidation analysis, 394
 - Constraint equations, 525
 - Constraint-function method, 447, 451
 - Contact algorithm, 453
 - selection of, 450
 - Contact analysis features, 470
 - Contact birth/death, 460
 - Contact body, 468
 - Contact compliance, 470
 - Contact definition features, 468
 - Contact detection, 472
 - Contact group properties, 454
 - Contact oscillations, suppressing, 473
 - Contact pairs, 440
 - Contact slip, 460
 - Contact surface compliance, 459
 - Contact surface depth, 456
 - Contact surface offsets, 454
 - Contact surfaces, 437
 - ATS method, 551
 - constraint-function method, 447, 451
-

- convergence, 482
 discretization, 476
 forces, 461
 initial penetration, 456
 output, 460
 rigid target method, 449
 segment method, 448
 three-dimensional, 439
 tractions, 460
 two-dimensional, 438
- Contact surfaces extension, 457
 Contact variables, 702
 Convergence criteria, 557
 contact, 559
 energy, 557, 560, 564, 576
 force and moment, 558, 560, 564, 576
 translation and rotation, 558, 564
- Coordinate variables, 703, 726
 Crack propagation, 630, 644
 crack growth control, 647
 definition of the crack propagation surface, 645
 modeling of the crack advance, 646
 node release-only technique, 644
 pressure loading, 647
 transition meshes, 652
- Crack resistance curves, 648
 Crack tip blunting, 651
 Creep coefficients
 user-supplied, 301
 Creep laws, 299
 eight-parameter, 299
 eight-parameter with variable coefficients, 301
 exponential, 299
 LUBBY2, 300
 LUBBY2 with variable coefficients, 302
 power, 299
- Creep material model, 9, 290, 299
 Creep strains, 299
 O.R.N.L. rules for cyclic loading conditions, 303
 strain hardening, 303
 time hardening, 304
 time offset, 305
- Creep-variable material model, 290, 301
 Curve description material model, 370
 cracking, 374
 cyclic parts, 662
 tension cut-off, 374
- Curve fitting for rubber and foam material models, 340
 Arruda-Boyce model, 344
 hyper-foam model, 344
 Mooney-Rivlin model, 342
 Ogden model, 343
- Cyclic boundaries, 662
 Cyclic symmetry analysis, 659
 periodic symmetry, 660
 restrictions, 663
- D**
- Damping, 573
 consistent, 573
 lumped, 573
 modal, 573
 Rayleigh, 573
- Darcy's law, 394
 Deformation gradient tensor
 elastic, 250
 inelastic, 250
 total, 247
- Deformation-dependent distributed loads, 495
 Deformation-dependent pressure loads, 495
 Determinant search method, 536
 Direct time integration, 238, 570
 Director vectors, 108

- Disk files, 685
displacements, 688
factorized linear stiffness matrices, 686
forces, 688
global mass, 686
global stiffness, 686
pipe internal pressures, 687
temperature gradients, 687
temperatures, 687
- Displacement variables, 705
- Displacement-based finite elements, 16, 36
- Displacement-based fluid elements, 182
3-D, 183
axisymmetric, 183
element output, 184
irrotational conditions, 182
planar, 183
- Distributed loads, 494
beam, 499
deformation dependent, 495
iso-beam, 501
pipe, 501
shell, 504
- Draw bead, 469
- Draw bead variables, 703
- Drucker-Prager material model, 376
- Dynamic analysis, 570, 574
- E**
- Earthquake loading, 573
- Effective plastic strain, 270
- Effective-stress-function algorithm, 282
- Eigenvalue problems, 531
- Eigenvector variables, 724
- Eight-parameter creep law
variable coefficients, 301
- Eight-parameter creep law, 299
- Elastic-isotropic material model, 251, 253
- Elastic-orthotropic material model, 251, 253
2-D solid elements, 253
3-D solid elements, 258
fabric, 389
plate elements, 258
shell elements, 259
- Electric field variables, 720
- Electromagnetic loads, 515
- Element ‘death upon rupture’, 671
- Element birth/death, 547, 664
- Element local node field variables, 703
- Element local node variables, 703
- Element locking, 149
- Element output for
2-D solid elements, 25
3-D solid elements, 47
axisymmetric shell elements, 95
beam elements, 63, 64, 68
displacement-based fluid elements, 182
general elements, 171
iso-beam elements, 96
moment-curvature beam elements, 78
nonlinear spring/damper/mass elements, 176
pipe elements, 167
plate-shell elements, 106
potential-based fluid elements, 239
shell elements, 140
truss elements, 11
- Element/layer field variables, 702
- Elements
2-D solid, 15
3-D solid, 34
axisymmetric shell, 84
cable, 7

- general, 169
 iso-beam, 84
 pipe, 151
 plate/shell, 99
 rebar, 12
 ring, 7
 shell, 107
 spring/damper/mass, 169
 truss, 7
- Energy release rates, 636
 Engineering strains, 241, 244
 Engineering stresses, 241, 245
 Envelope response type, 751
 Equilibrium iterations
 BFGS method, 549
 full Newton method, 549
 modified Newton method, 548
 Explicit dynamic analysis, 581
 Explicit integration
 central difference method, 575
 Explicit time integration, 570, 575
 Exponential creep law, 299
- F**
- Fabric material model, 389
 Failure criteria, 133
 Hashin failure criterion, 138
 maximum strain failure criterion, 135
 maximum stress failure criterion, 133
 tensor polynomial failure criterion, 137
 Tsai-Hill failure criterion, 136
 user-supplied failure criterion, 140
 Failure criterion variables, 722
 Fiber-matrix composites, 128
 Five degrees of freedom node, 118
 Flanges, 153, 157
 Floor response spectrum analysis, 617
 curve smoothing, 621
- damped transfer function method, 627
 peak broadening, 621
 response spectra loading, 622
 spatial combinations, 627
 time history loading, 619
 undamped transfer function, no resonance correction method, 626
 undamped transfer function, resonance correction method, 626
- Fluid variables, 707
 Flux variables, 718
 Follower loads, 493
 Force variables, 714
 Formulations for
 2-D solid elements, 21
 3-D solid elements, 43
 axisymmetric shell elements, 94
 iso-beam elements, 94
 pipe elements, 159
 plate-shell elements, 99
 shell elements, 115
 truss elements, 9
 Fourier analysis, 602
 power spectral density, 603
 Fracture mechanics, 629
 crack propagation, 630, 644
 line contour method, 629
 recommendations, 650
 user-supplied, 653
 virtual crack extension method, 630
 Fracture variables, 720
 Free surface modeling, 186
 Frequency analysis, 534
 error bound, 535
 Lanczos iteration method, 538
 of preloaded structures, 534
 physical error bound, 535

- potential-based fluid elements, 204, 238
 restart, 683
 rigid body modes, 535
 subspace iteration method, 536
- Frequency domain analysis, 591
 Fourier analysis, 591
 harmonic vibration analysis, 591
 random vibration analysis, 591
 response spectrum analysis, 591
 SDOF system response, 591
- Frequency/mode variables, 729
- Friction
 basic models, 462
 user-supplied models, 464
- Full Newton iterations, 547, 549, 562
 line searches, 549
- ## G
- Gasket material model, 399
- General elements, 169, 573
 element output, 170
- Geometric imperfections, 534
- Geotechnical material models, 370
- Green-Lagrange strains, 242, 244
- Gurson material model, 265
- ## H
- Harmonic response types, 749
- Harmonic vibration analysis
 applied forces, 605
 ground motion direction, 611
 ground motions, 606
 initial calculations, 611
 modal damping ratios, 611
 phase angle, 611
 potential-based fluid elements, 207, 238
 quasi-static amplitude, 608
 real part of the response, 611
- RMS amplitude, 611
 signed amplitude, 611
 sweep spectra, 611
- Hermitian beam elements, 54
- Hill yield condition, 278
- Holzapfel model for finite strain viscoelasticity, 349
- Hyper-foam material model, 337
 3-D analysis, 339
 axisymmetric analysis, 339
 plane strain analysis, 339
 plane stress analysis, 339
 selection of material constants, 339
- ## I
- Ilyushin material model, 282
- Implicit time integration, 571, 575
 Newmark method, 571, 575
 trapezoidal rule, 571, 576
 Wilson theta method, 571, 575
- Incompatible modes finite elements, 19, 39
- Inelastic deformations, 249
- Inertial properties, 681
 center of mass, 681
 centroid, 681
 element group, 681
 moments of inertia, 681
 products of inertia, 681
 total mass, 681
 total volume, 681
- Infinite elements, 186
- Initial accelerations, 673
- Initial conditions, 673
 initial accelerations, 673
 initial displacements, 673
 initial pipe internal pressures, 674
 initial strains, 674
 initial stresses, 677
 initial temperatures, 673
 temperature gradients, 673

- Initial displacements, 673
- Initial loads, 519
- Initial pipe internal pressures, 674
- Initial strains, 674
- element, 674
 - nodal point, 674
 - strain fields, 676
- ULH formulation, 677
- Initial stresses, 677
- element, 678
 - nodal point, 678
 - restrictions, 679
- Initial temperature gradients, 673
- Initial temperatures, 673
- Initial thermal strains, 674
- Initial velocities, 673
- Intial conditions
- initial velocities, 673
- IOPTIM parameter, 685
- Irradiation creep material model, 291, 302
- Iso-beam elements, 84
- element output, 94, 96
 - formulations, 94
 - general 3-D, 84
 - linear, 91
 - mass matrices, 96
 - material models, 94
 - nonlinear, 94
 - numerical integration, 89, 95
 - plane strain 2-D, 84
 - plane stress 2-D, 84
 - shear deformations, 89
 - warping effects, 88
- Isothermal plasticity material models, 265
- Isotropic hardening, 266
- Isotropic hyperelastic effects, 329
- Iterative solvers, 544
- J**
- J-integral, 629, 635
- K**
- Kinematic hardening, 266
- Kirchhoff stresses, 245
- L**
- Lanczos iteration method, 538
- Lankford coefficients, 280
- Large displacement formulation, 9, 65, 94, 99, 159, 252, 282, 286, 309, 370
- Large displacement/large strain formulation, 21, 43, 242, 243, 267, 273, 278, 285, 292, 328, 376, 391, 394
- Large displacement/small strain formulation, 21, 43, 241, 267, 273, 278, 284, 292, 376, 389, 391, 394
- Large strain analysis
- ULH formulation, 246
- Latent heat variables, 708
- LDC method, 487, 553, 564
- constant increment of external work method, 554
 - constant spherical arc length constraint method, 554
 - load vector multiplier, 555
- Line contour method, 629, 631
- definition of line contours, 633
 - output, 635
- Line contour variables, 703
- Line search, 550
- Linear buckling analysis
- secant formulation, 531
- Linear dynamic analysis, 570
- Linear formulation, 9, 22, 43, 99, 118, 159, 252

- Linear static analysis, 541, 586, 587, 588
Linearized buckling analysis, 531
 classical formulation, 531
 error bound, 532
 restart analysis, 533
Load penetration, 497
Load step response type, 749
Load step results, 748
Load/displacement incrementation, 546
Loading
 centrifugal, 504, 505
 concentrated, 490
 distributed, 494
 Electromagnetic, 515
 initial, 519
 mass-proportional, 504, 507
 pipe internal pressures, 514
 prescribed temperature gradients, 512
 prescribed temperatures, 512
 pressure, 493
 user-supplied, 519
Location-independent variables, 702
Logarithmic strains, 244
LUBBY2 creep law, 300
 variable coefficients, 302
- M**
- Mapping files, 512, 688
Mass
 inertial properties, 681
Mass matrices
 moment-curvature beam elements, 78
Mass matrices for
 2-D elements, 24
 iso-beam elements, 96
 pipe elements, 166
 plate/shell elements, 105
shell elements, 132, 133
truss elements, 10
Mass matrix, 584
Mass scaling, 589
Mass-proportional loads, 504, 507
 potential-based fluid elements, 220
Material models, 241
 Arruda-Boyce, 336
 Cam-clay, 380
 concrete, 430
 concrete, 309
 creep, 290
 creep variable, 301
 creep-variable, 290
 curve description, 370
 curve fitting, 340
 Drucker-Prager, 376
 elastic-isotropic, 251, 253
 elastic-orthotropic, 251, 253
 fabric, 389
 gasket, 399
 geotechnical, 370
 Gurson, 284
 hyper-foam, 337
 Ilyushin, 282
 irradiation creep, 291, 302
 isothermal plasticity, 265
 isotropic hyperelastic effects, 329
 Mohr-Coulomb, 385
 Mooney-Rivlin, 331
 Mroz-bilinear, 272
 Mullins effect, 358
 multilinear-plastic-creep, 290, 299
 multilinear-plastic-creep-variable, 290, 301
 nonlinear elastic, 262
 Ogden, 334
 orthotropic effect, 362
 plastic, 428
 plastic creep, 290, 299
 plastic-bilinear, 266

- plastic-creep-variable, 290, 301
 plastic-multilinear material, 266
 plastic-orthotropic, 278
 Ramberg-Osgood, 434
 rubber and foam, 328
 Shape Memory Alloy, 403
 SMA, 403
 temperature dependence, 368
 thermal strain effect, 366
 thermo-elastic, 286
 thermo-elasto-plasticity and creep, 290
 thermo-isotropic, 286
 thermo-orthotropic, 286
 thermo-plastic, 290
 thermoplasticity and creep, 428
 user-supplied, 409
 viscoelastic, 390, 433
 viscoelastic effects, 348
 viscoplastic, 422
- Material models for
 2-D solid elements, 21
 3-D solid elements, 42
 axisymmetric shell elements, 94
 iso-beam elements, 94
 pipe elements, 158
 plate-shell elements, 99
 shell elements, 115
 truss elements, 9
- Materially-nonlinear-only
 formulation, 9, 22, 43, 95, 99, 118, 159, 268, 273, 278, 282, 284, 287, 292, 309, 370, 376, 381, 385, 391, 394
- Matrices for
 3-D solid elements, 47
 axisymmetric shell elements, 96
 beam elements, 83
- Mean-square response, 612
 interpretation of, 615
- Memory allocation, 542
 in-core solution, 685
 out-of-core solution, 685
 Mesh evaluation, 752
 Mesh glueing, 528
 Mesh surface variables, 703
 Miscellaneous variables, 729
 Mixed interpolation formulation, 21, 42
 Mixed-interpolated finite elements, 18, 38
 Mixed-interpolation formulation, 269, 273, 292, 332, 395, 410
 Modal damping, 573
 Modal mass, 593
 Modal participation factors
 properties of, 599
 Modal stresses, 539
 Mode shape response type, 749
 Mode shape results, 748
 Mode superposition, 570, 572, 575, 577
 potential-based fluid elements, 205
 Modeling of gaps, 263, 272
 Modified Newton iterations, 546, 548, 562
 line searches, 548
 Mohr-Coulomb material model, 385
 Moment and force release option, 56
 Moment-curvature beam elements, 70
 axial behavior, 72
 cyclic behavior, 77
 elastic-plastic material model, 74
 element output, 79
 flexural behavior, 72
 mass matrices, 78
 nonlinear elastic model, 73
 numerical integration, 73
 section results, 79
 shear deformations, 73
 stress resultants, 79
 torsional behavior, 73

- warping effects, 73
- Moments of inertia
- inertial properties, 681
- Mooney-Rivlin material model, 331
- 3-D analysis, 332
 - axisymmetric analysis, 332
 - plane strain analysis, 332
 - plane stress analysis, 331
 - selection of material constants, 333
- Mroz hardening, 272
- Mroz-bilinear material model, 272
- Mullins effect, 358
- Multigrid solver, 546
- Multilayer shell elements, 127
- specification of layer thicknesses, 128
 - transverse shear stresses, 127
- Multilinear-plastic-creep material model, 290, 299
- Multilinear-plastic-creep-variable material model, 290, 301
- N**
- Newmark method, 571, 572, 575
- Node discrete variables, 703
- Node field variables, 702
- Node release-only technique, 644
- Node shift/release technique, 644
- Node-node contact, 444
- Nominal strains, 245
- Nonconvergence, 559, 560
- Nonlinear dynamic analysis, 574
- Nonlinear elastic material model, 262
- Nonlinear static analysis, 546
- selection of incremental solution method, 561
- Non-positive definite stiffness matrix, 543
- Numerical integration for
- 2-D solid elements, 22
 - 3-D solid elements, 44
- axisymmetric shell elements, 89, 95
- beam elements, 68
- iso-beam elements, 89, 95
- pipe elements, 162
- plate-shell elements, 103
- shell elements, 124
- truss elements, 9
- O**
- O.R.N.L. rules for cyclic loading conditions, 303
- Ogden material model, 334
- 3-D analysis, 334
 - axisymmetric analysis, 334
 - plane strain analysis, 334
 - plane stress analysis, 334
 - selection of material constants, 335
- Operating systems supported by ADINA, 2
- Ovalization degrees of freedom, 152, 156
- P**
- Parallel processing, 684
- rigid target contact algorithm, 684
 - scheduler types, 684
 - sparse solver, 684
- Periodic symmetry, 660
- Pipe elements, 151
- element output, 167
 - formulations, 159
 - mass matrices, 166
 - material models, 158
 - numerical integration, 162
 - ovalization degrees of freedom, 152
 - pipe-beam, 151
 - pipe-shell, 151
 - recommendations for use, 168
 - warping degrees of freedom, 152
- Pipe internal pressures, 153, 159, 514

- Plastic strains, 295
 Plastic-bilinear material model, 266
 Plastic-creep material model, 290, 299
 Plastic-creep-variable material model, 290, 301
 Plastic-multilinear material model, 266
 Plastic-orthotropic material model, 278
 Lankford coefficients, 280
 Plate/shell elements, 99
 mass matrices, 105
 Plate-shell elements
 elasto-plastic material model, 104
 element output, 106
 formulations, 99
 material models, 99
 numerical integration, 103
 shear deformations, 102
 stiffness matrix for, 100
 Pore pressures, 394
 Porous media formulation, 394
 Porous structures, 394
 Positive definite stiffness matrix, 541
 Post-collapse response, 553, 564
 Postprocessing, 695
 Potential-based fluid elements, 184
 3-D, 185
 axisymmetric, 185
 choice of formulation, 209
 direct time integration, 238
 element output, 239
 free surfaces, 186
 ground motions, 200
 infinite elements, 186
 initial pressure in, 236
 mass-proportional loads, 220
 planar, 185
 pressure loads, 220
 static analysis, 237
 Power creep law, 299
 Power spectral density, 603, 612
 Prescribed displacements, 510
 Prescribed load variables, 725
 Prescribed temperature gradients, 512
 Prescribed temperatures, 512
 Pressure loads, 493
 deformation-dependent, 495
 load penetration, 497
 plate/shell, 504
 potential-based fluid elements, 220
 shell, 504
 three-dimensional, 496
 two-dimensional, 496
 Products of inertia
 inertial properties, 681
- ## Q
- Quarter-point midside nodes, 650
 Quasi-static amplitude, 608
- ## R
- Radiation field variables, 703
 Ramberg-Osgood material model, 434
 Random response type, 617, 750
 Random response types, 612
 Random vibration analysis, 612
 ground motion direction, 616
 initial calculations, 616
 mean-square response, 612
 modal damping ratios, 616
 potential-based fluid elements, 207, 238
 power spectral density, 612, 616
 Rate-dependent plasticity, 271
 Rayleigh damping, 573, 575, 585
 Reaction variables, 718
 Reactions, 663
 Rebar elements, 12
 Recommendations for use of shell elements, 147

- Remeshing, 688
Residual response type, 749
Residual results, 749
Response spectrum analysis, 591
 absolute sum method, 594
 algebraic sum method, 595
 CQC method, 596
 double sum method, 595
 DSC method, 595
 ground motion direction, 601
 initial calculations, 599
 loading response spectra, 601
 modal combinations, 594
 modal damping ratios, 601
 modal-combinations, 601
 potential-based fluid elements, 207,
 238
 residual responses, 592
 residual terms, 596, 601
 spatial combinations, 601
 SRSS method, 594
 static correction terms, 596
 static load case, 599
 ten percent method, 594
Response spectrum response type, 749
Response spectrum analysis
 static load case, 602
Response type
 response-combination, 601
Response types
 envelope, 751
 harmonic, 749
 load step, 749
 mode shape, 749
 random, 612, 617, 750
 residual, 749
 response spectrum, 749
 response-combination, 750
 response-spectrum, 601
Response-combination response type,
 601, 750
Response-spectrum response type,
 601
Restart option, 681
 frequency analysis, 683
Result calculation within the AUI, 695
 combination transformations, 696
 examples, 696
 resultant transformations, 696
 resultants after combinations, 698
 resultants before combinations, 698
 spatial transformations, 696
 time transformations, 696
Result locations
 contact segments, 740
 draw bead segments, 740
 element section, 739
 element/layer, 735
 line contours, 741
 location-independent, 735
 mesh integration points, 741
 node, 735
 radiosity segments, 741
 virtual shifts, 741
Results
 load step, 748
 mode shape, 748
 residual, 749
Resuses (of substructures), 656
Retained nodes, 656
Rigid body modes, 535
Rigid links, 526
Rigid target, 684
Rigid target method, 449
Ring element, 7
Rubber and foam material models,
 328
Rupture conditions, 271, 277, 281,
 293, 296, 304

S

SDOF system response, 617
 Section field variables, 703
 Segment method, 448
 Selection of contact algorithm, 450
 Shape Memory Alloy, 403
 Shape Memory Alloy material model,
 403
 Shape memory effect, 403
 Shell elements, 107
 16-node, 149
 4-node, 147
 9-node, 149
 basic assumptions in, 107
 composite, 127
 director vectors, 108
 displacement-dependent pressure
 loading, 118
 element output, 140
 failure criteria, 133
 locking, 149
 mass matrices, 132, 133
 material models, 115
 MITC16, 149
 MITC4, 147
 MITC9, 149
 multilayer, 127
 nodal forces, 146
 nodal point degrees of freedom, 118
 numerical integration, 124
 recommendations for use, 147
 section results, 143
 shear deformations, 113
 stress resultants, 143
 thick, 147
 thin, 147
 transition, 122
 Shell midsurfaces
 plotting results onto, 748
 Shell stiffeners, 89
 Shell-shell intersections, 124

Shell-solid intersections, 124
 Shifting procedure, 539
 Six degrees of freedom node, 120
 Skew systems, 485
 SMA, 403
 SMA material model, 403
 Small displacement formulation, 9,
 65, 94, 99, 159, 252, 282, 286, 309,
 370, 381, 385
 Small displacement/small strain
 formulation, 21, 43, 241, 267, 273,
 278, 284, 292, 376, 391, 394
 Smoothing, 746
 restrictions, 747
 Solution of $KU=R$, 541
 active column solver, 542
 iterative solvers, 544
 multigrid solver, 546
 sparse solver, 542
 Sparse matrix solver, 542
 Sparse solver, 684
 in-core, 543
 memory allocation, 542
 out-of-core, 543
 parallel processing, 684
 Spring/damper/mass elements, 169,
 573
 linear single-degree-of-freedom,
 171
 linear two-degree-of-freedom, 172
 nonlinear, 173
 Spring-tied elements, 176
 Static analysis, 541, 586, 587, 588
 potential-based fluid elements, 201,
 237
 Stiffness stabilization, 547, 691
 Strain energy densities, 680
 Strain hardening, 303
 Strain measures, 241, 244
 Almansi strains, 244
 engineering strains, 241, 244

-
- Green-Lagrange strains, 244
 logarithmic strains, 244
 stretches, 244
 Strain variables, 711
 Strain-rate effects
 2-D elements, 271
 3-D elements, 271
 shell elements, 271
 truss elements, 271
 Stress measures, 241, 245
 2nd Piola-Kirchhoff stress, 245
 Cauchy stress, 245
 Cauchy stresses, 241
 engineering stress, 245
 engineering stresses, 241
 Kirchhoff stress, 245
 Stress variables, 708
 Stretch tensor
 left, 249
 Stretches, 244
 Structural vibration, 572
 Sturm sequence check, 538
 Subspace iteration method, 536
 Sturm sequence check, 538
 Substructures, 656
 Substructuring, 655
 condensed nodes, 656
 linear analysis, 655
 nonlinear analysis, 657
 restrictions, 658
 retained nodes, 656
 Superelastic effect, 403
 Supported computers and operating systems, 2
 Suppressing contact oscillations, 473
 Swelling analysis, 394
 Symmetry condition (for pipe elements), 157
- T**
- Temperature files, 512
 Temperature variables, 707
 Temperature-dependent material, 368
 Thermal strains, 287, 295
 Thermo-elastic material models, 286
 Thermo-elasto-plasticity and creep material models, 290
 Thermo-isotropic material model, 286
 Thermo-orthotropic material model, 286
 Thermo-plastic material model, 290
 Thermoplasticity and creep material model, 428
 Thermorheologically simple material, 329, 368
 Thickness variables, 728
 Tied contact, 444
 Time functions, 486, 488
 constant, 488
 short circuit, 488
 sinusoidal, 488
 Time hardening, 304
 Time offset, 305
 Time step management, 588
 Time variables, 728
 TL formulation, 22, 43, 252, 268, 273, 278, 287, 292, 309, 328, 370, 376, 381, 385, 391, 394, 410
 Traction variables, 717
 Trapezoidal rule, 571, 576
 TRS material, 329, 368
 True strains, 245
 Truss elements, 7
 element output, 11
 formulations, 9
 mass matrices, 10
 material models, 9
 numerical integration, 9
 rebar, 12
 recommendations for use, 11
-

U

UL formulation, 65, 252, 268, 282, 287, 292
 ULH formulation, 43, 242, 243, 268, 273, 292, 376, 381, 385, 391, 394, 410
 ULJ formulation, 243, 268, 278
 Undrained analysis, 394
 User-defined variables, 733
 User-supplied creep coefficients, 301
 User-supplied elements, 177
 User-supplied fracture mechanics, 653
 User-supplied friction models, 464
 User-supplied loads, 519
 User-supplied material model, 409

V

Variables, 702
 acceleration, 706
 contact, 702
 coordinate, 703, 726
 displacement, 705
 draw bead, 703
 eigenvector, 724
 electric field, 720
 element local node, 703
 element local node field, 703
 element/layer field, 702
 failure criterion, 722
 fluid, 707
 flux, 718
 force, 714
 fracture, 720
 frequency/mode, 729
 interpolation within elements, 745
 latent heat, 708
 line contour, 703
 location-independent, 702
 mesh surface, 703

miscellaneous, 729
 node discrete, 703
 node field, 702
 prescribed load, 724, 725
 radiation field, 703
 reaction, 718
 section field, 703
 smoothing, 746
 strain, 711
 stress, 708
 temperature, 707
 thickness, 728
 time, 728
 traction, 717
 User-defined, 733
 velocity, 705
 virtual shift, 703

Velocity variables, 705

Virtual crack extension method, 630, 635

description of virtual material shifts, 640
 dynamic correction, 639
 output, 642
 pressure correction, 638
 thermal correction, 638

Virtual material shifts, 640
 spatially fixed, 647
 spatially moving, 647
 virtual vectors, 642

Virtual shift variables, 703

Viscoelastic effects, 348

Viscoelastic material model, 390, 433
 Prony series, 393

Viscoplastic material model, 422

Volume

inertial properties, 681

von Karman ovalization modes, 154

von Mises yield condition, 68

W

Warping degrees of freedom, [152](#), [156](#)

Wave propagation, [571](#)

Wilson theta method, [571](#), [575](#)

Z

Zero-slope-of-pipe-skin, [157](#)