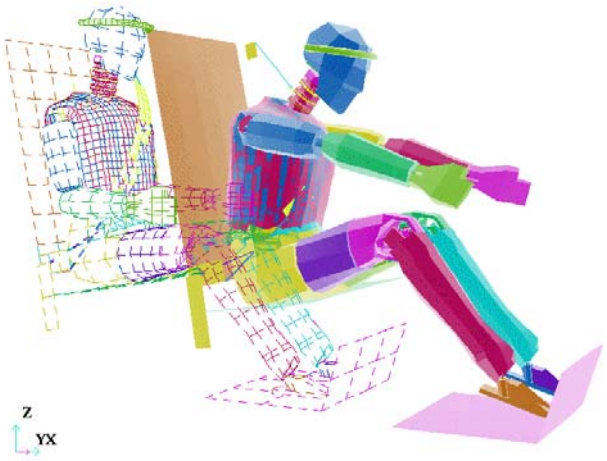

DYNAMIC FINITE ELEMENT METHODS

Lecture notes for SD2450
Biomechanics and Neuronics

By

Svein Kleiven, Peter Halldin and Dan Zenkert

Preface

Dynamic Finite Element Methods - Lecture Notes for SD2450, is a set of lecture notes following the taught lectures on dynamic finite element method (FEM), part in the course *Biomechanics and Neuronics.*

The second Chapter of this lecture notes is a compressed form of the Lecture notes in Matrix and Finite Element Methods, Paper 97-23 written by Dan Zenkert. This first part should only be seen as a fresh up on the Finite Element theory.

Hereafter, is the dynamic finite element method presented. The purpose of this part of the lecture notes is to give the reader a short introduction to this numerical technique. There are a few books dealing with non-linear finite element methods including the dynamic part, which more completely covers this topic [Bathe 1996 and Cook 1989].

The text is by no means complete, but could serve as an introductory text to the subject. There are numerous text books on the subject and there is still much research into FEM presently going on all over the world, and hence the amount of knowledge is continuously increasing.

Stockholm 2001-03-15

Svein Kleiven, Peter Halldin, and Dan Zenkert

Table of contents

Nomenclature

Here, the most important symbols used are defined. There are other symbols used for specific purposes in some parts of the text, and these are not given in the list below.

1, 2, 3	Element local coordinates [m or mm]
f	Polynomical function
h	Element length [m or mm]
k	Elements of stiffness matrix [various dimensions used]
n	Displacement in local coordinates [m or mm]
q	Distributed load [N/m or N/mm]
t	Time [s]
u, v, w	Element displacements in global *x, y* and *z*-directions [m or mm]
x, y, z	Global coordinates [m or mm]
w_i	Weight functions in Gauss integrations

A	Rod or beam cross-section area [m^2 or mm^2]
E	Elastic modulus [N/m^2 or N/mm^2]
F	Force component in global coordinates [N]
N	Force component in local coordinates [N]
L	Length of beam or rod [m or mm]
P	Point load or local load [N]

Greek letter

ε	Normal strain [-]
γ	Shear strain [-]
λ	Eigenvalue, buckling coefficient [-]
ν	Poisson ratio [-]
ω	Natural frequency [s^{-1}]
σ	Normal stress [N/m^2 or N/mm^2]
ρ	Density [kg/m^3]
τ	Shear stress [N/m^2 or N/mm^2]
ξ, η	Local element coordinates [-]
Π	Total potential energy [Nm or Nmm]

Operators

d/dx	Ordinary derivative with respect to e.g. *x*
∂/∂x	Partial derivative with respect to e.g. *x*
$\|\boldsymbol{x}\|$	Determinant of matrix $\boldsymbol{x}$
Δ	Laplace operator

Vectors or matrices (all given in bold face)

$\boldsymbol{u}$	Displacement vector (mainly used to rods) [m or mm]
$\boldsymbol{v}$	Virtual displacement vector [m or mm]
$\boldsymbol{w}$	Displacement vector (mainly used for beams) [m or mm]
$\boldsymbol{B}$	Vector of first order differentiated shape functions [-]
$\boldsymbol{E}$	Elasticity matrix (Hooke's law) [N/m^2 or N/mm^2]
$\boldsymbol{F}$	Force vector in global coordinatess [N]

$\boldsymbol{G}$	Vector of second order differentiated shape functions [-]
$\boldsymbol{J}$	Jacobian matrix [-]
$\boldsymbol{K_l}$	Element local stiffness matrix [various dimensions used]
$\boldsymbol{K^e}$	Element stiffness matrix
$\boldsymbol{K}$	Structure or element stiffness matrix
$\boldsymbol{M}$	Element or structure mass matrix
$\boldsymbol{N}$	Force vector in local coordinates [N] in analysis of trusses and rods
$\boldsymbol{N}$	Vector of shape functions [-] in FE
$\boldsymbol{T}$	Transformation matrix [-] in analysis of trusses and rods
$\boldsymbol{U}$	Vector of nodal diplacements

Indices

1, 2, 3	refers to local *1, 2* or *3*-direction
i, j	summation indices or tensor indices
x, y, z	refers to global *x, y* or *z*-direction

1 Numerical analysis

The Finite element method (FEM) is quite new as an engineering tool, being developed mainly during the 1960's and 1970's. During the past decades, development of small powerful personal computers and work stations has made FE-codes become a tool as common to many engineers as the pocket calculator. It has also become much more accessible through the easy-to-use interfaces provided by most commercial FE-codes. However, the engineering use of such codes still requires a thorough knowledge of both FEM and of the problem to be solved if an accurate result will be obtained.

The type of analysis is also very important and depends on the assumptions made, i.e., the idealisation. This requires insight into the physics of the problem. There are some sorts of loads applied to the structure; are these time dependant, do they change with deformation? Are deformations small or large? Are the material characteristics linear or non-linear in the regime in which analysis is performed? Are there areas of high stresses that require mesh refinements? Could there be contact between surfaces and if so, will friction have an effect? There are many such questions that should be addressed prior to modelling.

In fact, one should really need to know the structural response prior to making the analysis in order to be able to construct the model. Since this is not the case one must usually make estimations of the response, rely on experience or intuition. However, the result of the analysis often contain surprises! This may require a new idealisation of the problem, leading to a new model and more analyses. Estimations are required to approximately find load paths, deflections, etc. This is where a thorough theoretical knowledge is essential. These estimations will later also serve as a check when compared to the results of the analysis. Examples of such estimations could be; to find the maximum deflection of a plate leading to a decision on whether a linear analysis will suffice or a large deflection analysis may be required, or if stresses can be assumed to be within the linear elastic regime of the material.

Statically linear systems
For statically linear system, the response, e.g., deformations, are proportional to the load. In such cases, the structural stiffness matrix is constant and independent of the load amplitude. Most problems fall within this category. The finite element equations are then simply written as

$$\boldsymbol{F} = \boldsymbol{K}\boldsymbol{u}$$

Where $\boldsymbol{F}$ is the applied load, $\boldsymbol{K}$ is the stiffness matrix and $\boldsymbol{u}$ is the nodal deformation, see Section 2.

Statically non-linear systems
There are cases when the material characteristics, the geometry or the boundary conditions change with the deformation of the structure. In such cases, the stiffness matrix will depend on the deformation and the problem becomes non-linear. There are two main types of non-linearities; material and/or geometric non-linear problems. In the former case, the material response (the σ - ε curve for the material) is non-linear so that the modulus of the material depends on the strain. This implies that the stiffness matrix will also depend on the strain, which in turn depends on the deformation. Geometrically non-linear problems stem some different phenomena. These are;

(i) *Large strains* which affect global geometry, e.g. the cross-section of a beam decreases due to contraction.

(ii) *Large deformations* (and rotations). It is possible that large deformation may occur without large strains. This means that the compatibility equations (defining the relation between strains and deformations) no longer can be described by a constant matrix $\boldsymbol{B}$. A typical example of this is large deflection plate bending.

(iii) *Non-conservative loads*. As the deformations are allowed to develop the loading may in some problems follow the deformed shape, i.e., they change direction. A classical example of this is the so called Beck's column which is basically the same as the Euler type I column (clamped at one end and free at the other). In Beck's column the load is applied tangentially at the free end rather than as a constant vertical load (which direction is independent of the deformation – conservative) and this results in a different critical buckling load.

(iv) Large deformations can also change the boundary conditions. Such problems often occur when contact is considered. As the structure deform, the point of contact changes.

The finite element equation for non-linear problems may be written as

$$\boldsymbol{F}(\boldsymbol{u}) = \boldsymbol{K}(\boldsymbol{u})\boldsymbol{u}$$

Non-linear problems are usually solved by taking a series of linear steps. We write the equations in an incremental form $\boldsymbol{K}\delta\boldsymbol{u} = \delta\boldsymbol{F}$. Here, the stiffness matrix is a function of the displacement vector $\boldsymbol{u}$. In turn, the current $\boldsymbol{u}$ is the sum of the preceding $\delta\boldsymbol{u}$'s. The current $\boldsymbol{K}$ is called the *tangential stiffness matrix* and is used to compute the next step $\delta\boldsymbol{u}$. Then, a new update of $\boldsymbol{u}$ is computed, an update of $\boldsymbol{K}$ and another step is taken. The feature of geometrically non-linear problems is that the equilibrium equations must be written with respect to the deformed geometry. There are several different methods to perform this stepwise calculations and some are given by Cook (1981).

Dynamic problems

Dynamic analyses must be performed when the mass inertia forces become large, i.e. of the same order of magnitude as the applied static loads. The mass inertia forces are proportional to the acceleration of the nodal points. For cyclic loads, this acceleration is proportional to the square of the loading frequency. The main consideration is now the frequency of the loading, which must be compared with the resonance or eigen frequency of the structure. A light and stiff structure has high eigen-frequencies whereas a heavy compliant system has low eigen-frequencies. A rule-of-thumb is that if the loading frequency is lower than 1/5 of the lowest structural eigen-frequency one may treat the load as static. This will imply that the load and the structural response are acting in-phase.

There are two types of dynamic loads; *stationary* or *transient dynamic loads*. In the latter case one normally use *modal analysis* to determine the eigen-frequencies of the system. For structural safety these system eigen-frequencies should not coincide with the frequency(-ies) of the loading.

A *transient* loading could be a short pulse loading (impact) or an initial value problem (a suddenly released constraint for example). Crash simulation of car bodies, object impact or shock loading are examples where transient analysis is required. The solution can be obtained by numerical integration of the equations of motion. This can be done by so called *implicit* or

explicit time integration, where the latter is more commonly used, see Section 3. There are special FE-codes for this purpose, e.g. ABAQUS Explicit and LS-DYNA 3D.

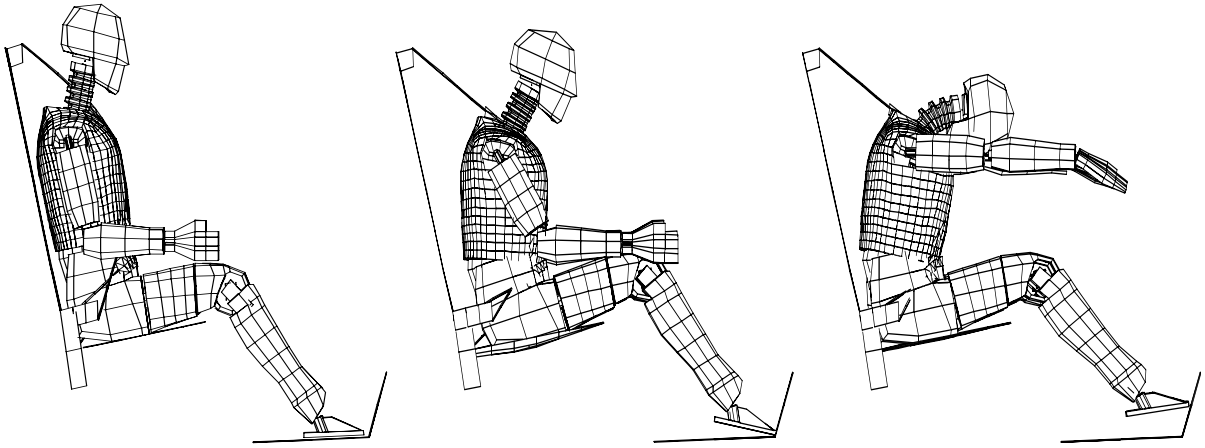

Figure 1 Example of transient FEM analysis; simulation of car crash with a belted dummy.

Section 2 gives a short background to the linear static FEM. This section is just to fresh up your static FE knowledge from earlier courses. Section 3 deals with the basic theories behind the dynamic FEM and finally Section 4 will give examples on important tips & trix when working with dynamic FE programs.

2 The static finite element formulation

This Section covers the most basic parts from the Lecture Notes 4E1100/01 Lightweight Structures.

The notes are quite theoretical emphasising the derivation of the governing partial differential equations (PDE) used for finite element (FE) formulations, i.e. the so called weak form of the equation. Derivations are mainly performed using the method of virtual work but also energy methods are presented.

FE is a method formulated for computerised calculations and is not supposed to be done by means of pen and paper. Thus, even the smallest hand-made example becomes overly complicated to solve by FE approximation. However, through the automated formulation, very large problems with complicated boundaries and load cases can easily be solved numerically.

2.1 Governing equation

Let's study the one-dimensional elasticity problem, as schematically illustrated in Fig.2.1 from a general viewpoint. It is a rod (or bar) with varying cross-section, i.e. $A = A(x)$, subjected to an edge load F and a body force $q(x)$. In the most general case, even the material properties are allowed to vary, i.e. $E = E(x)$. In the following, only a one-dimensional problem is studied, and hence the local and global coordinates are assumed the same. Thus, $\boldsymbol{F} = \boldsymbol{N}$ and $\boldsymbol{u} = \boldsymbol{n}$ and the notation for forces and displacements are denoted $\boldsymbol{F}$ and $\boldsymbol{u}$. A further reason for this is that $\boldsymbol{N}$ is later used to denote shape function vectors.

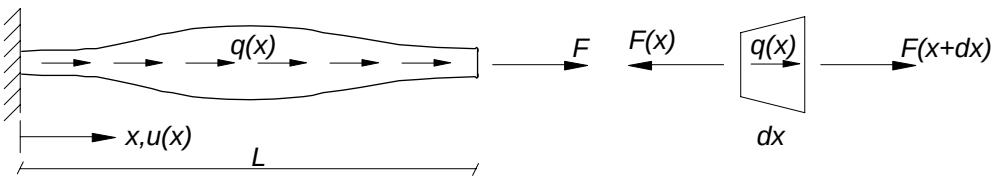

Figure 2.1 Schematic of one-dimensional elasticity problem

Note here that the "pressure" or body force denoted q is actually a line load with dimension [N/m] or [N/mm]. The problem is assessed using the following fundamental elasticity relations:

From equilibrium $\quad F(x+dx) + q(x)dx - F(x) = 0 \rightarrow q(x) = -\dfrac{dF(x)}{dx}$

Stress definition $\quad \sigma(x) = \dfrac{F(x)}{A(x)}$

Constitutive relation $\quad \sigma(x) = E(x)\varepsilon(x)$

Strain definition $\quad \varepsilon(x) = \dfrac{du(x)}{dx}$

These combined give

$$\frac{d}{dx}\left(F(x)\right) = \frac{d}{dx}\left(\sigma(x)A(x)\right) = \frac{d}{dx}\left(E(x)A(x)\frac{du(x)}{dx}\right) = -q(x) \tag{2.1}$$

which is a second order differential equation in the unknown displacement *u*. This is the so called *strong formulation* of the governing equation and any displacement assumption used for the solution of this problem must be at least twice differentiable in order to approximate at least a constant load *q*. *u* is the said to require C^1 *continuity*.

Boundary conditions: $u(0) = u^*$ Kinematic or essential B.C.
$F(L) = F_0$ Natural B.C.

Any displacement assumption (*ansatz*) must satisfy the kinematic B.C. and there should exist a second derivative.

2.2 Virtual work method

Now, multiply the governing differential equation with an arbitrary scalar function or a so called *weight function v(x)*

$$v(x)\left[\frac{d}{dx}\left(E(x)A(x)\frac{du(x)}{dx}\right) + q(x)\right] = 0$$

and then integrate over the entire domain

$$\int_0^L v(x)\left[\frac{d}{dx}\left(E(x)A(x)\frac{du(x)}{dx}\right) + q(x)\right]dx = 0$$

Equality must then also hold. Next, integrate by parts to obtain

$$\left[EA\frac{du}{dx}v\right]_0^L - \int_0^L EA\frac{du}{dx}\frac{dv}{dx}dx + \int_0^L qvdx = 0$$

$$\int_0^L EA\frac{du}{dx}\frac{dv}{dx}dx = \int_0^L qvdx + F(L)v(L) - F(0)v(0) \tag{2.2}$$

This is the so called *weak formulation* of the governing differential equation. This equation can also be derived from the principle of the *minimum of the potential energy*, which will be done below. *v(x)* can be interpreted as a virtual displacement that satisfy the kinematic boundary conditions, i.e. an *ansatz*. As now seen, for any displacement assumption only a first derivative must exist and hence simpler functions may be used. This is why the formulation is described as *weak*. As seen, the used displacement functions (assumed displacements) must have first derivatives which may be integrated. Thus, the derivatives *du/dx* and *dv/dx* are not necessarily continuous, however, *u* and *v* must be continuous. Hence, only C^0 continuity is required, which means that only displacement must be continuous. This means that displacements only need to be continuous over element boundaries, i.e., at nodes. The function *v* is called a *virtual displacement* and *dv/dx* a *virtual strain*. Each term in eq.(2.2) represents a virtual work - by internal stresses, imposed tractions (*q*) and boundary forces, from left to right, respectively.

2.3 The Ritz method

Assume displacement field *u(x)*. The total elastic energy in the system can be written

$$\Pi = \int_V \int_0^{\varepsilon} \sigma d\varepsilon dV - \int_0^L qu dx - F(L)u(L) + F(0)u(0) \tag{2.3}$$

but for a linear elastic material

$\int_0^{\varepsilon} \sigma d\varepsilon = \frac{1}{2}\sigma\varepsilon$ and by using $\sigma = E\varepsilon$ and that $\varepsilon(x) = \frac{du}{dx}$ this can be rewritten to

$$\Pi = \frac{1}{2}\int_0^L EA\frac{du}{dx}\frac{du}{dx}dx - \int_0^L qu dx - F(L)u(L) + F(0)u(0)$$

The potential energy is functional in *u(x)*. We can now apply the theorem of the *minimum of the potential energy*, which states that *the potential energy of an elastic system has a stationary value for all small displacement when the system is in equilibrium, and the equilibrium is stable if the stationary value is a minimum*. Applied here this means that applying a small displacement δu to the displacement u so that it becomes $u + \delta u$. Thus,

$$\Pi(u+\delta u) = \Pi(u) + \frac{1}{2}\int_0^L EA\frac{d\delta u}{dx}\frac{du}{dx}dx + \frac{1}{2}\int_0^L EA\frac{du}{dx}\frac{d\delta u}{dx}dx$$
$$+ \frac{1}{2}\int_0^L EA\frac{d\delta u}{dx}\frac{d\delta u}{dx}dx - \int_0^L q\delta u dx - F(L)\delta u(L) + F(0)\delta u(0)$$

and now $\delta\Pi = \Pi(u + \delta u) - \Pi(u)$ must be zero leading to

$$\int_0^L EA\frac{d\delta u}{dx}\frac{du}{dx}dx = \int_0^L q\delta u dx + F(L)\delta u(L) - F(0)\delta u(0)$$

but δu can just as well be interpreted as a small displacement. As seen, the equation is exactly the same as eq.(2.2) derived by other means.

2.4 The finite element method

For most boundary value problems the geometry and loading are so complex that it is impossible to find a displacement assumption that satisfies the governing differential equation and all the boundary conditions.

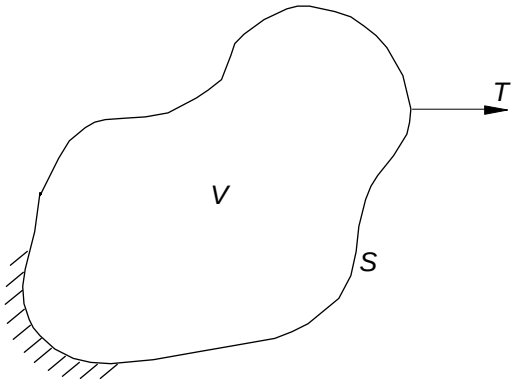

Figure 2.2 Domain (*V*) of boundary value problem. Boundary *S* and surface tractions *T*.

Instead one can possibly divide the domain V into subdomains V_h and make simple assumptions for each one and make sure continuity exists over subdomain boundaries. With this approach one misses the exact solution but we hope to find a set of u_h within each V_h which together approximates the exact solution. In the rod case we must use shape functions which are piecewise linear (can be differentiated ones). This can be done by the approximation

$$u(x) = \sum_{i=1}^{n} u_i N_i(x) \tag{2.4}$$

where N_i is a piecewise linear function that can be differentiated (at least) ones and u_i are nodal values of u. N_i are called the *element shape functions*. An example of a shape function is shown in Fig.2.3.

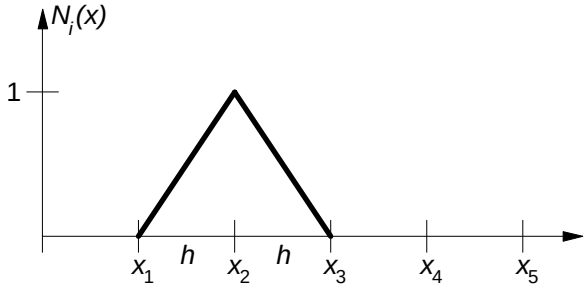

Figure 2.3 Example of shape function

The interval $x \in (h,2h)$ is called *finite element*

Each point $x = 0$, *h, 2h,* etc. is called *node*

Now superimpose several $N_i(x)$ so that eq.(2.4) is satisfied for some choice of coefficients u_i.

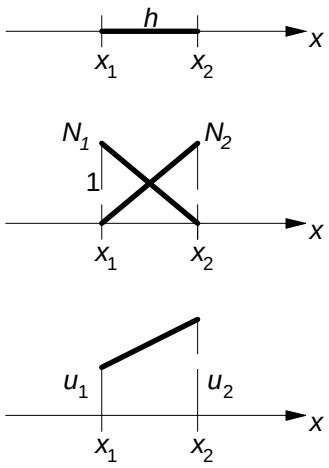

Figure 2.4 Example of shape functions

Linear element: $h = x_2 - x_1$

$$N_1(x) = \frac{x_2 - x}{h}$$

$$N_2(x) = \frac{x - x_1}{h}$$

Gives the deformation in the element as

$$u(x) = \sum_{i=1}^{2} u_i N_i(x)$$

$$= \frac{x_2 - x}{h} u_1 + \frac{x - x_1}{h} u_2$$

i.e. $u(x_1) = u_1$ and $u(x_2) = u_2$

The derivatives are then also

$$\frac{dN_1}{dx} = -\frac{1}{h}, \quad \frac{dN_2}{dx} = \frac{1}{h} \text{ and then } \frac{du}{dx} = \sum_{i=1}^{2} u_i \frac{dN_i}{dx} = \frac{-u_1}{h} + \frac{u_2}{h}$$

Hence, over the entire interval we get something as schematically illustrated in Fig.2.5, provided the constraints $u(0) = u(L) = 0$.

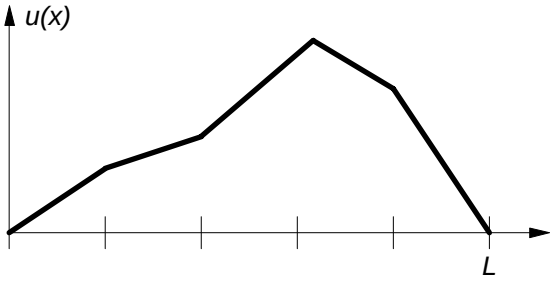

Figure 2.5 Example of shape function

If we now go back again to the rod problem and make the finite element discretisation

$$u(x) = \sum_{i=1}^{n} u_i N_i(x) \text{ and } v(x) = \sum_{j=1}^{n} v_j N_j(x)$$

where u is the assumed displacement field, and v an assumed virtual displacement. Introducing these relations into the weak formulation of the governing differential equation, Eq.(2.2) yields

$$\int_0^L EA \sum_{i=1}^{n} \sum_{j=1}^{n} u_i v_j N_i' N_j' dx = \int_0^L q \sum_{j=1}^{n} v_j N_j dx + F(L) \sum_{j=1}^{n} v_j N_j(L) - F(0) \sum_{j=1}^{n} v_j N_j(0)$$

but since displacements u_i and v_j are discrete nodal values and thus not functions of x, we can shift the order of the summation and the integration to

$$\sum_{j=1}^{n}\left\{\int_0^L EA\sum_{i=1}^{n}u_i N_i' N_j' dx\right\}\cdot v_j = \sum_{j=1}^{n}\left\{\int_0^L qN_j dx + F(L)N_j(L) - F(0)N_j(0)\right\}\cdot v_j$$

which must be valid for an arbitrary function v or vector v_j . To further highlight the result, rewrite to

$$\sum_{i=1}^{n}\int_0^L EAN_i' N_j' dx \cdot u_i = \int_0^L qN_j dx + F(L)N_j(L) - F(0)N_j(0) \text{ for every } j \tag{2.5}$$

Thus, for every j in Eq.(2.5) the equation gives on row in the equation system (j are rows and i columns in this notation). This is equal to writing

$$\sum_{i=1}^{n} k_{ji}u_i = F_j \text{ or in matrix form } \boldsymbol{F} = \boldsymbol{Ku} \tag{2.6}$$

i.e., n equations in n unknowns. The components of the stiffness matrix $\boldsymbol{K}$ are thus

$$k_{ij} = \int_0^L EAN_i' N_j' dx \tag{2.7}$$

the components of the load vector $\boldsymbol{F}$ are

$$F_j = \int_0^L qN_j dx + F(L)N_j(L) - F(0)N_j(0) \tag{2.8}$$

and $\boldsymbol{u}$ is a vector of nodal displacements.

Now, the expression in eq.(2.8) implies that the number of stiffness matrix components to be evaluated are $n \times n$, where n are the number of nodes. Going back to Fig.2.2, we used a simple linear shape function for which N_j only differs from zero in those elements containing node i. This means that components in the stiffness matrix k_{ij} will only differ from zero if the element contains nodal points i and j. Furthermore, this also implies that the stiffness matrix becomes a banded matrix with non-zero entries clustered along the diagonal, provided the node numbering is kept in some ordered manner.

If the shape function chosen has such properties, eq.(2.2) can be rewritten as

$$\int_0^L EA\frac{du}{dx}\frac{dv}{dx}dx = \sum_{elem}\int_{x_i}^{x_{i+1}} EA\frac{du}{dx}\frac{dv}{dx}dx = \sum_{elem}\int_0^h EA\frac{du}{dx}\frac{dv}{dx}dx$$

i.e., instead of integration over the entire structure, one can integrate over each element and eventually sum up all individual components - stiffness matrix assembly.

2.5 Stiffness matrix assembly

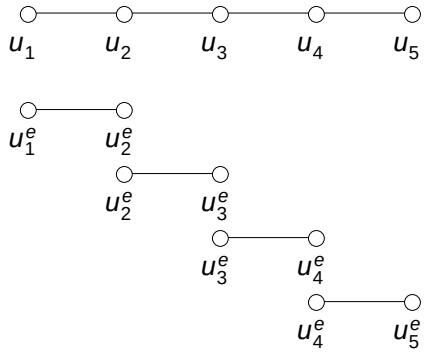

Figure 2.6 Assembly of rod elements

The element stiffness matrices are now assembled as

$$\begin{bmatrix} x & x & 0 & 0 & 0 \\ x & x & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & x & x & 0 & 0 \\ 0 & x & x & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & x & x & 0 \\ 0 & 0 & x & x & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & x & x \end{bmatrix} = \begin{bmatrix} x & x & 0 & 0 & 0 \\ x & x & x & 0 & 0 \\ 0 & x & x & x & 0 \\ 0 & 0 & x & x & x \\ 0 & 0 & 0 & x & x \end{bmatrix}$$

and for the load vector

$$\begin{pmatrix} x \\ x \\ 0 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ x \\ x \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ x \\ x \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 0 \\ x \\ x \end{pmatrix} = \begin{pmatrix} x \\ x \\ x \\ x \\ x \end{pmatrix}$$

which finally gives the system of equations to be solved as

$$\boldsymbol{Ku} = \boldsymbol{F} \tag{2.9}$$

The solution to this equation system is unique if rigid body motion (translation and rotation) is prevented by appropriate boundary conditions. We can further also deduct that displacements naturally are continuous over element boundaries since they are connected in the nodes. However, the strains (and hence also stresses) are constant within the elements, since

$$\varepsilon_x = \frac{du}{dx} = \frac{du}{d\xi}\frac{d\xi}{dx} = \left(-\frac{u_1}{2} + \frac{u_2}{2}\right)\frac{2}{h} = \frac{1}{h}(u_2 - u_1)$$

and hence not continuous over the element boundaries. The internal load is obtained through

$$P(x) = EA\varepsilon(x)$$

which then also must be constant within each element. Recall that the body force q has been transformed to nodal loads!

2.6 Numerical integration

The type of integrals appearing in finite element formulations

$$I = \int_{-1}^{1} f(\xi) d\xi$$

are poorly suited for automated computations. Instead, approximate the integral I with

$$I = \int_{-1}^{1} f(\xi) d\xi \approx \sum_{i=1}^{n} w_i f(\xi_i) \tag{2.10}$$

where w_i are *weight functions*. The number of evaluation points (Gauss points) depends on the order of the function f. Assume that f is a linear polynomial, as shown in Fig.2.7.

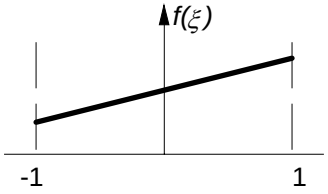

Figure 2.7 Linear function in the ξ system.

$I = 2f(0)$ gives exact integration. In general, assume

$$f(\xi) = \alpha_1 + \alpha_2 \xi + \alpha_3 \xi^2 + \alpha_4 \xi^3 + \ldots\ldots + \alpha_{2n} \xi^{2n-1}$$

Then

$$I = \int_{-1}^{1} f(\xi) d\xi = 2\alpha_1 + \frac{2}{3}\alpha_3 + \ldots\ldots + \frac{2}{2n-1}\alpha_{2n-1}$$

but as also defined

$$I = \sum_{i=1}^{n} w_i f(\xi_i) = \alpha_1 \sum_{i=1}^{n} w_i + \alpha_2 \sum_{i=1}^{n} w_i \xi_i + \alpha_3 \sum_{i=1}^{n} w_i \xi_i^2 + \ldots\ldots + \alpha_{2n} \sum_{i=1}^{n} w_i \xi_i^{2n-1}$$

If we now compare these, we find that

$$\sum_{i=1}^{n} w_i = 2,\ \sum_{i=1}^{n} w_i \xi_i = 0,\ \sum_{i=1}^{n} w_i \xi_i^2 = \frac{2}{3},\ \sum_{i=1}^{n} w_i \xi_i^{2n-2} = \frac{2}{2n-1} \text{ and } \sum_{i=1}^{n} w_i \xi_i^{2n-1} = 0 \tag{2.11}$$

which thus provide $2n$ conditions; n for positions ξ_i and n for weights w_i for the $2n$ unknown α's in the polynomial. Hence, for a polynomial of degree $p = 2n-1$ it is sufficient to use n points for exact integration. Or, exact integration is obtained if the number of integration points n is larger than, or equal to

$$n \geq \frac{p+1}{2} \tag{2.12}$$

For example, a third degree polynomial ($p = 3$) it should be sufficient to use $n = 2$ integration points. Thus, from eq.(10)

$$w_1 + w_2 = 2,\ \xi_1 w_1 + \xi_2 w_2 = 0,\ \xi_1^2 w_1 + \xi_2^2 w_2 = \frac{2}{3} \text{ and } \xi_1^3 w_1 + \xi_2^3 w_2 = 0$$

which provide us with four equations in four unknowns which can be solved directly. However, from symmetry we get that $w_1 = w_2$ and that $\xi_1 = -\xi_2$, then it is easily deducted that

$w_1 = w_2 = 1$ and $\xi_1 = -\frac{1}{\sqrt{3}} = -\xi_2$

and thus

$$\int_{-1}^{1} \sum_{i=1}^{4} \alpha_i \xi^{i-1} d\xi = 1 \times f\left(-\frac{1}{\sqrt{3}}\right) + 1 \times f\left(\frac{1}{\sqrt{3}}\right)$$

Similarly, for $p = 5$ it is required to use three integration points (eq.(2.11)), which results in

$$w_1 = w_3 = \frac{5}{9},\ w_2 = \frac{8}{9} \text{ and } \xi_1 = -\xi_3 = -\sqrt{\frac{3}{5}},\ \xi_2 = 0$$

$I_{\text{exact}} = w_1 f(\xi_1) + w_2 f(\xi_2) + w_3 f(\xi_3)$

Some more results are given in Table 2.1.

Gauss points	ξ_i	W_i
$n = 1$	0.00	2.00
2	±0.577350	1.00
3	0.00	0.888888
	±0.774596	0.555555
4	±0.339981	0.652145
	±0.861136	0.347854
5	0.00	0.568888
	±0.538469	0.478628
	±0.906179	0.236926

Table 1 Positions and weights for Gauss integration.

3 The Dynamic Finite Element Method

Direct time integration methods for dynamic analysis of structures, using the finite element method are described.

3.1 Introduction

If the frequency of excitation applied to a structure is less than roughly one-fifth of the structure's lowest natural frequency of vibration, then the effects of inertia can be neglected and the problem is *quasistatic.* The *mass matrix*, is written as [**m**] for an element and [**M**] for a structure, accounts for inertia and is a discrete representation of mass in a structure. Problems of structural dynamics can be divided into two broad classifications. In one, we ask for *natural frequencies of vibration* and the corresponding mode shapes. Usually we want to compare the natural frequencies of the structure with frequencies of excitation. It is usually desirable to assure that these frequencies are well separated. In the other classification, we ask how a structure moves with time under prescribed loads and/or motions of its supports; that is, we ask for *time-history* analysis. Two popular methods of time-history analysis are *direct integration methods* and *modal methods.* In this compendium we only consider direct integration methods.

Finite Element Equations

Let us now derive the finite element equations, described in chapter 2. But now we consider the response of the general three-dimensional body shown in Fig. 3.1.

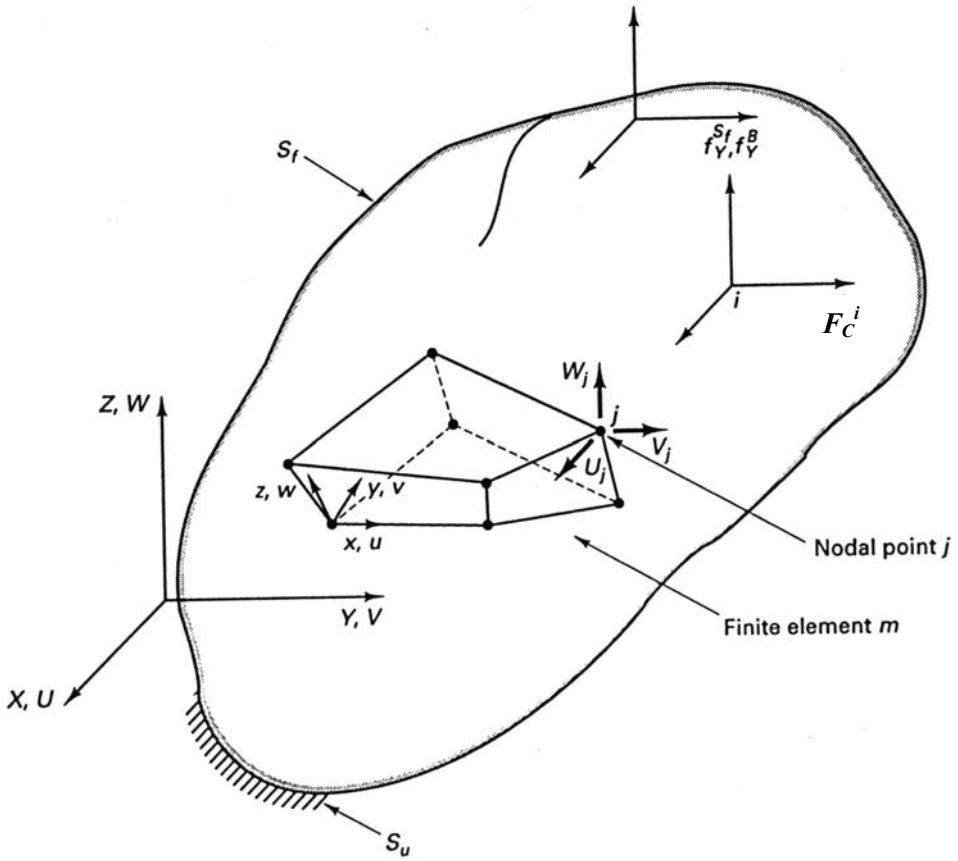

Figure 3.1 – General three dimensional body with an 8-node three-dimensional element.

In the finite element analysis we approximate the body in Fig. 3.1 as an assemblage of discrete finite elements interconnected at nodal points on the element boundaries. The displacements measured in a local coordinate system *x, y, z* within each element are assumed to be a function of the displacement at the *j* finite element nodal points. Therefore, for element *m* we have:

$$\boldsymbol{u^{(m)}(x,y,z) = N^{(m)}(x,y,z)\hat{U}} \qquad \textbf{or shorter} \qquad \boldsymbol{u^{(m)} = N^{(m)}\hat{U}} \tag{3.1}$$

Where $N^{(m)}$ is the displacement interpolation matrix, the superscript *m* denotes element *m,* and $\hat{U}$ is a vector of the three global displacement components U_i, V_i, W_i at all nodal points, including those at the support of the element assemblage; i.e., $\hat{U}$ is a vector of dimension 3*N,*

$$\boldsymbol{\hat{U}^T = [U_1\ V_1\ W_1\ U_2\ V_2\ W_2\ \ldots U_N\ V_N\ W_N]} \tag{3.2}$$

Or more generally:

$$\boldsymbol{\hat{U}^T = [U_1\ U_2\ \ldots .\ U_n\ ]} \tag{3.3}$$

Where it is understood that U_i may correspond to a displacement in any direction *X, Y, or Z,* and may also signify a rotation when we consider beams, plates, or shells. Since $\hat{U}$ includes the displacements (and rotations) at the supports of the element assemblage, we need to impose the known values of $\hat{U}$ prior to solving for the unknown nodal point displacements. Figure 3.1 shows a typical finite element of the assemblage. This element has eight nodal points, one at each of its corners, and can be thought of as a "brick" element. We should imagine that the complete body is represented as an assemblage of such brick elements put together so that there are no gaps between the element domains.With the assumptions on the displacements in (3.1) we can now evaluate the corresponding element strains,

$$\boldsymbol{\varepsilon^{(m)}(x,y,z) = \partial u^{(m)}(x,y,z) = B^{(m)}(x,y,z)\hat{U}} \tag{3.4}$$

where

$$\boldsymbol{B^{(m)}(x,y,z) = \partial N^{(m)}(x,y,z)}$$

or shorter

$$\boldsymbol{\varepsilon^{(m)} = B^{(m)}\hat{U}}$$

where *B* is the strain-displacement matrix; the rows of *B* are obtained by appropriately differentiating and combining rows of the matrix *N.* The differential operator matrix ∂ is 6 by 3 for three-dimensional problems, and 3 by 2 for two-dimensional problems. In two dimensions:

$$\begin{Bmatrix} \varepsilon_x \\ \varepsilon_y \\ \gamma_{xy} \end{Bmatrix} = \begin{bmatrix} \dfrac{\partial}{\partial x} & 0 \\ 0 & \dfrac{\partial}{\partial y} \\ \dfrac{\partial}{\partial y} & \dfrac{\partial}{\partial x} \end{bmatrix} \begin{Bmatrix} u \\ v \end{Bmatrix} = \partial \begin{Bmatrix} u \\ v \end{Bmatrix} \tag{3.5}$$

The stresses in a finite element are related to the element strains and the element initial stresses using:

$$\boldsymbol{\sigma}^{(m)} = \boldsymbol{C}^{(m)}\boldsymbol{\varepsilon}^{(m)} + \boldsymbol{\sigma}^{I(m)} \tag{3.6}$$

where $C^{(m)}$ is the elasticity matrix of element m and $\sigma^{I(m)}$ are the given element initial stresses.

The purpose of defining the element displacements and strains in terms of the complete array of finite element assemblage nodal point displacements makes it easy to formulate *the principle of virtual displacements* which is the basis of the displacement-based finite element method. This principle states that the equilibrium of the body in Fig.3.1 requires that for any compatible virtual displacement imposed on the body in its state of equilibrium, the total internal virtual work is equal to the total external virtual work:

Internal virtual work *External virtual work*

$$\int_V \bar{\boldsymbol{\varepsilon}}^T \boldsymbol{\sigma}\, dV = \int_V \bar{\boldsymbol{U}}^T \boldsymbol{f}^B dV + \int_{S_f} \bar{\boldsymbol{U}}^{S_f\,T} \boldsymbol{f}^{S_f} dS + \sum_i \bar{\boldsymbol{U}}^{iT} \boldsymbol{F}_C^i \tag{3.7}$$

Where $\bar{\boldsymbol{\varepsilon}}^{\mathbf{T}}$ are the virtual strains corresponding to virtual displacements, $\bar{\boldsymbol{U}}^{\mathbf{T}}$, and $\boldsymbol{\sigma}$ is the stresses in equilibrium with applied loads. We will se that by the use of (3.1) and (3.4) in the principle of virtual displacements will automatically lead to an effective assemblage process of all element matrices into the governing structure matrices. This assemblage process is referred to as *the direct stiffness method.* First we rewrite (3.7) as a sum of integration's over the volume and areas of all finite elements:

$$\sum_m \int_{V^{(m)}} \bar{\boldsymbol{\varepsilon}}^{(m)T} \boldsymbol{\sigma}^{(m)}\, dV^{(m)} = \sum_m \int_{V^{(m)}} \bar{\boldsymbol{U}}^{(m)T} \boldsymbol{f}^{B(m)} dV^{(m)} + \sum_m \int_{S_1^{(m)},\ldots S_q^{(m)}} \bar{\boldsymbol{U}}^{S(m)T} \boldsymbol{f}^{S(m)} dS^{(m)} + \sum_i \bar{\boldsymbol{U}}^{iT} \boldsymbol{F}_C^i \tag{3.8}$$

Where m=1,2,…k, where k=number of elements, and $S_1^{(m)},\ldots,S_q^{(m)}$ denotes the element surfaces that are part of the body surface S.

If we now substitute (3.1), (3.4) and (3.6), for the virtual displacements, stress and strains, into (3.8), we obtain

$$\bar{\hat{\boldsymbol{U}}}^T \left[\sum_m \int_{V^{(m)}} \boldsymbol{B}^{(m)T} \boldsymbol{C}^{(m)} \boldsymbol{B}^{(m)}\, dV^{(m)} \right] \bar{\hat{\boldsymbol{U}}} =$$

$$\bar{\hat{\boldsymbol{U}}}^T \left[\sum_m \int_{V^{(m)}} \boldsymbol{N}^{(m)T} \boldsymbol{f}^{B(m)} dV^{(m)} + \sum_m \int_{S_1^{(m)},\ldots S_q^{(m)}} \boldsymbol{N}^{S(m)T} \boldsymbol{f}^{S(m)} dS^{(m)} - \sum_m \int_{V^{(m)}} B^{(m)T} \sigma^{I(m)} dV^{(m)} + \boldsymbol{F}_C \right] \tag{3.9}$$

To obtain from (3.9) the equations for the unknown nodal point displacements, we apply the principle of virtual displacements n times by imposing unit virtual displacements in turn for all components of $\bar{\hat{\boldsymbol{U}}}$. In the first application $\bar{\hat{\boldsymbol{U}}} = [\boldsymbol{1,0,0,\ldots,0}]$, in the second application

$\overline{\hat{U}} = [0,1,0,...,0]$ etc., until the *n*th component, so that for an assemblage of elements we obtain equilibrium by:

$$KU=F \tag{3.9}$$

where the matrix **K** is the stiffness matrix of the element assemblage,

$$K = \sum_m \int_{V^{(m)}} B^{(m)T} C^{(m)} B^{(m)} dV^{(m)} \quad = \sum K^{(m)} \tag{3.10}$$

The load vector **F** is given by $\mathbf{F= F_B + F_S - F_I + F_C}$, and the effect of the element body forces is included by:

$$F_B = \sum_m \int_{V^{(m)}} N^{(m)T} f^{B(m)} dV^{(m)} \tag{3.11}$$

the effect of element surface forces,

$$F_S = \sum_m \int_{S_1^{(m)},...S_q^{(m)}} N^{S(m)^T} f^{S(m)} dS^{(m)} \tag{3.12}$$

the effect of element initial stresses,

$$F_I = \sum_m \int_{V^{(m)}} B^{(m)T} \sigma^{I(m)} dV^{(m)} \tag{3.13}$$

and the nodal concentrated loads $\mathbf{F}_c$. We should note that the *i*th component in $\mathbf{F}_c$ is the concentrated nodal force that corresponds to the *i*th displacement component in $\hat{\mathbf{U}}$. We also note that the summation of the element volume integrals in (3.10) expresses the direct addition of the element stiffness matrices $\mathbf{K}^{(m)}$ to obtain the stiffness matrix of the total assemblage. In the same way, the assemblage force vectors are obtained. The process of assembling the element matrices by this direct addition is called the *direct stiffness method.* Equation (3.9) is a statement of the static equilibrium of the element assemblage. In these considerations, the applied forces may vary in time, in which the displacements also vary with time and (3.9) is a statement of equilibrium for any specific point in time.

Assuming that the element velocities and accelerations are approximated in the same way as the element displacements in (3.1), and by including element inertia as contributions to the body forces, we obtain:

$$F_B = \sum_m \int_{V^{(m)}} N^{(m)T} [f^{B(m)} - \rho^{(m)} N^{(m)} \ddot{U}] dV^{(m)} \tag{3.14}$$

The equilibrium equations are, in this case,

$$M\ddot{U} + KU = F \tag{3.15}$$

where the matrix **M** is the mass matrix of the structure,

$$\boldsymbol{M} = \sum_{m} \int_{V^{(m)}} \rho^{(m)} \boldsymbol{N}^{(m)T} \boldsymbol{N}^{(m)} \boldsymbol{dV}^{(m)} \tag{3.16}$$

3.2 Time-History analysis. Direct Integration Methods

In direct integration, the equations in (3.15) are integrated using a step-by-step procedure. The term "direct" meaning that prior to the numerical integration, no transformation of the equations into different forms is carried out. In direct integration methods or step-by-step methods, a finite difference approximation is used to replace the time derivatives appearing in equation (3.15) i.e. the velocities and accelerations are approximated by differences of displacements at various instants of time.

Difference methods for direct integration can be categorised as *explicit* or *implicit.* Explicit methods have the form

$$\boldsymbol{U}_{t+\Delta t} = \boldsymbol{f}(\boldsymbol{U}_t, \dot{\boldsymbol{U}}_t, \ddot{\boldsymbol{U}}_t, \boldsymbol{U}_{t-\Delta t}, ...) \tag{3.17}$$

hence $U_{t+\Delta t}$ is permitted to be determined in terms of completely historical information consisting of displacements and time derivatives of displacements at time *t* and before. Implicit methods have the form

$$\boldsymbol{U}_{t+\Delta t} = \boldsymbol{f}(\dot{\boldsymbol{U}}_{t+\Delta t}, \ddot{\boldsymbol{U}}_{t+\Delta t}, \boldsymbol{U}_t, ...) \tag{3.18}$$

And hence computation of $U_{t+\Delta t}$ requires knowledge of the time derivatives of $U_{t+\Delta t}$, which are unknown. Explicit and implicit methods have markedly different properties. This has important practical implications.

3.3 The Central-Difference Method

A popular method, which is characteristic of explicit methods in general, is the *central-difference method*. It approximates the velocities and accelerations by

$$\ddot{\boldsymbol{U}}_t = \frac{\boldsymbol{1}}{\boldsymbol{\Delta t^2}} (\boldsymbol{U}_{t+\Delta t} - \boldsymbol{2U}_t + \boldsymbol{U}_{t-\Delta t}) \tag{3.19}$$

$$\dot{\boldsymbol{U}}_t = \frac{\boldsymbol{1}}{\boldsymbol{2\Delta t}} (\boldsymbol{U}_{t+\Delta t} - \boldsymbol{U}_{t-\Delta t}) \tag{3.20}$$

These equations are obtained by expanding $U_{t+\Delta t}$ and $U_{t-\Delta t}$ in Taylor series about time *t*

$$\boldsymbol{U}_{t+\Delta t} = \boldsymbol{U}_t + \Delta t\dot{\boldsymbol{U}}_t + \frac{\Delta t^2}{2}\ddot{\boldsymbol{U}}_t + \frac{\Delta t^3}{6}\dddot{\boldsymbol{U}}_t + \cdots \tag{3.21}$$

$$\boldsymbol{U}_{t-\Delta t} = \boldsymbol{U}_t - \Delta t\dot{\boldsymbol{U}}_t + \frac{\Delta t^2}{2}\ddot{\boldsymbol{U}}_t - \frac{\Delta t^3}{6}\dddot{\boldsymbol{U}}_t + \cdots \tag{3.22}$$

Subtracting (3.22) from (3.21) yields (3.19) while adding them yields (3.20). In both cases, terms containing Δt^2 and higher powers are omitted. Hence, the central difference formulas are said to be *second-order accurate*. In other words, the error is of $O(\Delta t^2)$, which implies that halving the time step should approximately quarter the error.

The displacement solution for time $t+\Delta t$ is obtained by considering the equation of motion (3.15) at time *t*, i.e.,

$$\boldsymbol{M}\ddot{\boldsymbol{U}}_t + \boldsymbol{K}\boldsymbol{U}_t = \boldsymbol{F}_t \tag{3.23}$$

Substituting (3.19) and (3.20) into (3.23) gives

$$\left[\frac{1}{\Delta t^2}\boldsymbol{M}\right]\boldsymbol{U}_{t+\Delta t} = \boldsymbol{F}_t - \left[\boldsymbol{K} - \frac{2}{\Delta t^2}\boldsymbol{M}\right]\boldsymbol{U}_t - \left[\frac{1}{\Delta t^2}\boldsymbol{M}\right]\boldsymbol{U}_{t-\Delta t} \tag{3.24}$$

From which we can solve for $U_{t+\Delta t.}$ It should be noted that the solution of $U_{t+\Delta t}$ is thus based on using the equilibrium conditions at time *t*, i.e., $U_{t+\Delta t}$ is calculated using (3.23). For this reason the integration procedure is called an *explicit integration method.* Such integration schemes do not require a factorisation of the stiffness matrix in the step-by-step solution.

It can be shown that equation (3.24) is *conditionally stable* and require Δt such that

$$\Delta t \leq 2/\omega_{max} \tag{3.25}$$

Where ω_{max} is the highest natural frequency of $\det(\mathbf{K}-\omega^2\mathbf{M})=0$. If this is not satisfied, computations will be *unstable*.

3.4 Implicit Direct Integration Methods

Most of the useful implicit methods are *unconditionally stable* and have no restriction on the time step size other than as required for accuracy. A popular implicit method is called the *Newmark Method.* This method is given by

$$\dot{\boldsymbol{U}}_{t+\Delta t} = \dot{\boldsymbol{U}}_t + \Delta t\left[(1-\gamma)\ddot{\boldsymbol{U}}_t + \gamma\ddot{\boldsymbol{U}}_{t+\Delta t}\right] \tag{3.26}$$

$$\boldsymbol{U}_{t+\Delta t} = \boldsymbol{U}_t + \Delta t\dot{\boldsymbol{U}}_t + \frac{\Delta t^2}{2}\left[(1-2\beta)\ddot{\boldsymbol{U}}_t + 2\beta\ddot{\boldsymbol{U}}_{t+\Delta t}\right] \tag{3.27}$$

Where β and γ are chosen to control stability and accuracy. Newmark was originally proposed as an unconditionally stable scheme (called the constant-average-acceleration method), in which case γ=1/2 and β=1/4. In addition to (3.25) and (3.26), for solution of the displacements, velocities, and accelerations at time $t+\Delta t$, the equilibrium equations (3.15) at time $t+\Delta t$ are also considered:

$$\boldsymbol{M}\ddot{\boldsymbol{U}}_{t+\Delta t} + \boldsymbol{K}\boldsymbol{U}_{t+\Delta t} = \boldsymbol{F}_{t+\Delta t} \tag{3.28}$$

Because the equilibrium at time $t+\Delta t$ is considered, this method is said to be *implicit.*

3.5 Example of an Explicit time integration scheme

The equation (3.23) is solved in the time domain by explict integration using the central difference method. The velocities, $\dot{\boldsymbol{U}}_{t+\Delta t/2}$, are obtained by expanding $\dot{\boldsymbol{U}}_{t+\Delta t/2}$ and $\dot{\boldsymbol{U}}_{t-\Delta t/2}$ in Taylor series around time, *t,* and taking the difference. The displacements, $\boldsymbol{U}_{t+\Delta t}$, are obtained directly from (3.21), using the velocity at $t+\Delta t/2$ to improve the accuracy. For explicit programs the equation of motion is evaluated at the present time, *t*.

$$\ddot{\boldsymbol{U}}_t = \boldsymbol{M}^{-1}\left(\boldsymbol{F}_t - \boldsymbol{K}\boldsymbol{U}_t\right) \tag{3.29}$$

$$\dot{\boldsymbol{U}}_{t+\Delta t/2} = \dot{\boldsymbol{U}}_{t-\Delta t/2} + \ddot{\boldsymbol{U}}_t \Delta t \tag{3.30}$$

$$\boldsymbol{U}_{t+\Delta t} = \boldsymbol{U}_t + \dot{\boldsymbol{U}}_{t+\Delta t/2}\ \Delta \boldsymbol{t} \tag{3.31}$$

$$\boldsymbol{\sigma}_{t+\Delta t} = \boldsymbol{\sigma}_t + \dot{\boldsymbol{\sigma}}_{t+\Delta t/2}\boldsymbol{\Delta t} \tag{3.32}$$

where the stress rate $\dot{\boldsymbol{\sigma}}$ is achieved from the constitutive law, and Δt is the time step size. In the stress rate update, rigid body rotations must be accounted for. This is necessary to avoid inducing stresses due to rigid body motions.

The geometry is updated by adding the total displacements to the initial geometry

$$\boldsymbol{X}_{t+\Delta t} = \boldsymbol{X}_0 + \boldsymbol{U}_{t+\Delta t} \tag{3.31}$$

4 Specific dynamic finite element matters

Explicit dynamic FE codes have become quite stable computational tools used to solve complex problems. It is possible to use the code as a black box from which results are computed. However, it is important to have some theoretical knowledge to be able to discretisize and compute the dynamic FE problem and also critically to analyse and results.

Most of the difficulties arise from contact problems or the fact that just one integration point is used in the elements to reduce computer time.

4.1 Time step

In both implicit and explicit time integration, the problem is divided into a number of time steps. In implicit codes the time step is around 1 millisecond (1E-3 s) and in explicit codes the time step is around 1 microsecond (1E-6 s). The short time step is the reason for the large CPU time that is required for the dynamic finite element codes like DYNA3D and ABAQUS-Explicit.

The duration of the time step for each element is dependent on the stiffness (E), the characteristic length (L) and the density of each element. The time step is equal to the time it takes for the sound to pass the smallest element (or element side) in the model, (Eq. 4.1).

$$\Delta t < L / c \tag{4.1}$$

where L is the characteristic length in an element and $c^2 = E / \rho$.

The characteristic length is defined as possible:
- solid element. L=Volume/largest face area
- shell element. L=min(L1,L2) where L1=area/smallest diagonal, L2=smallest element side
- beam. L= length of the smallest beam element.

4.2 The finite element mesh

In order to reduce the computational time it is important to have a conform mesh, i.e. all elements should have approximately the same size. See example in Figure 4.1.

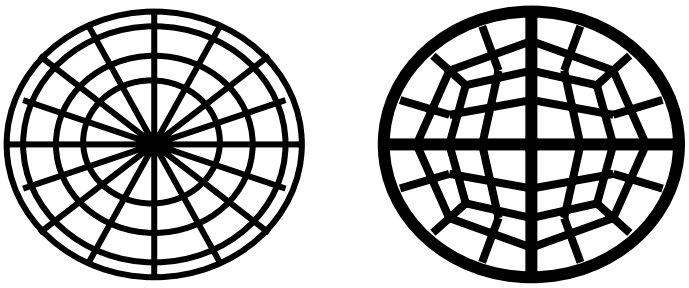

Figure 4.1 To the left: BAD degenerated elements with small time step. To the right: GOOD conform mesh.

4.3 Hourglassing

As mentioned above dynamic FE codes often use one integration point to save computational time. The biggest disadvantage to one-point integration is the need to control the zero energy modes when they arise, called hourglassing modes, see Figure 4.2. In static finite element codes like ANSYS at least 4/8 integration points are used for shell/solid elements. One integration point cannot store information for all nodes. The hourglass mode is the biggest reason for instabilities in dynamic finite element calculations.

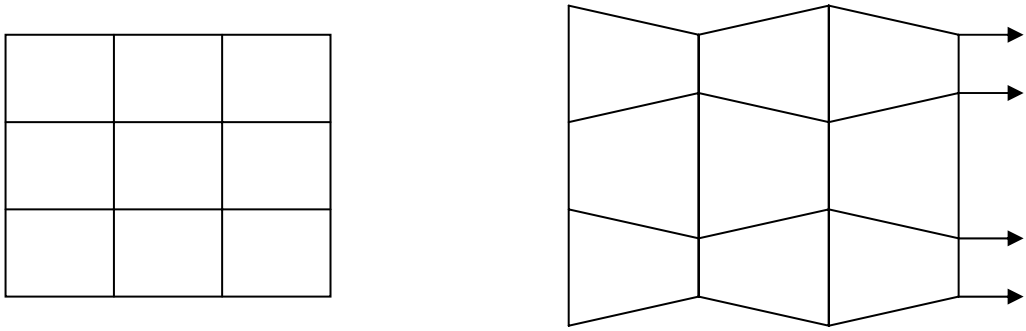

Figure 4.2 Typical membrane hourglass deformation.

An additional internal force is required to maintain the deformation stability of the element. Normally in dynamic finite element programs a force is set orthogonal to the first principal strains. It could look as in Figure 4.3.

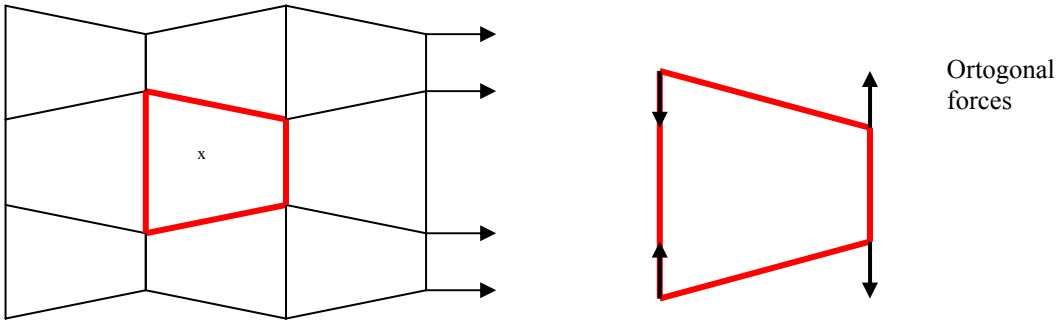

Figure 4.3. Ortogonal forces to stabilise the hourglass deformation.

The orthogonal forces used to control the hourglass modes should be small. If they are to large, energy will disipate from internal energy to hourglass energy. As a rule-of-thumb the quota between the hourglass energy and the internal energy should not exceed 10%.

In three dimensions the hourglass modes often result in negative volumes for solid elements, see example in Figure 4.4.

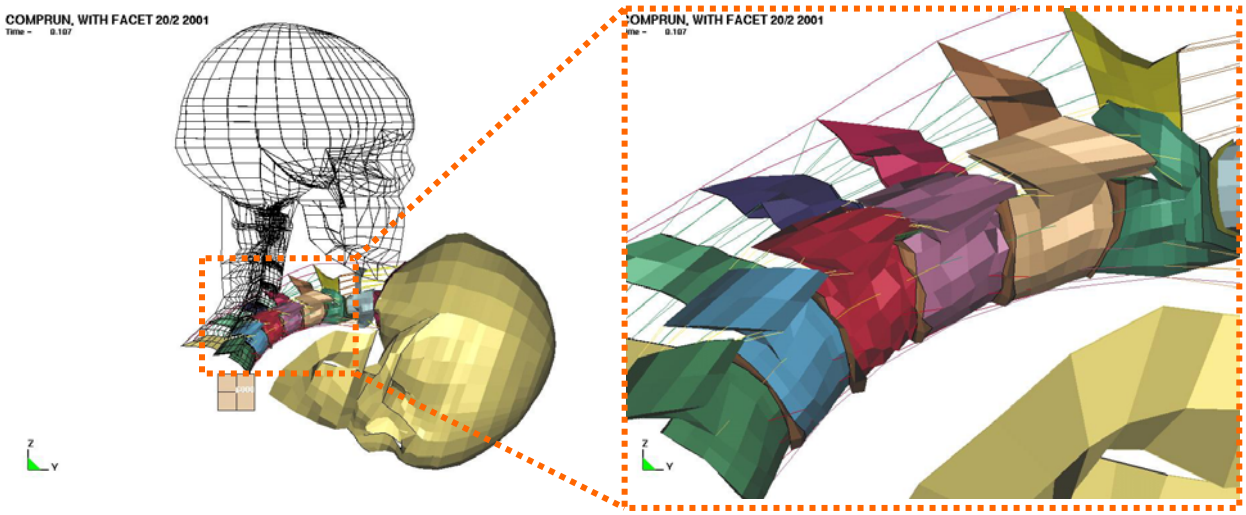

Figure 4.4 Example on hourglassing in a finite element model of the neck.

4.4 Contacts

In many technical problems there are contacts between parts that need to be defined to simulate the correct boundary conditions. For example in car crash simulations contacts are an important ingredience to describe an impact between two bodies. Other problems that include contacts are draw-bed simulations used to analyse metal sheet forming.

In static FE codes, contacs are normally treated by the use of so called Gap-elements, see Figure 4.5. The Gap-element can be likened to a non-linear spring that is attached to nodes belonging to the impacting bodies. The nodes need to lie close to each other and in pairs for the best solution.

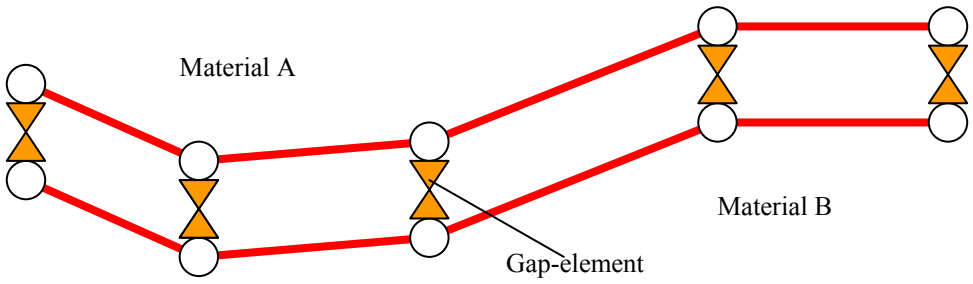

Figure 4.5 Schematic figure of gap elements used in static FE codes.

Dynamic FE codes do have more stable contact algorithms and contact problems are solved more efficiently than in static analysis. The major reason for this is that the problem is divided in such small time steps that the contacts separating different materials in the analysis have time to react. Another reason is that contact often is defined between nodes and surfaces resulting in less restrictions of the mesh geometry.

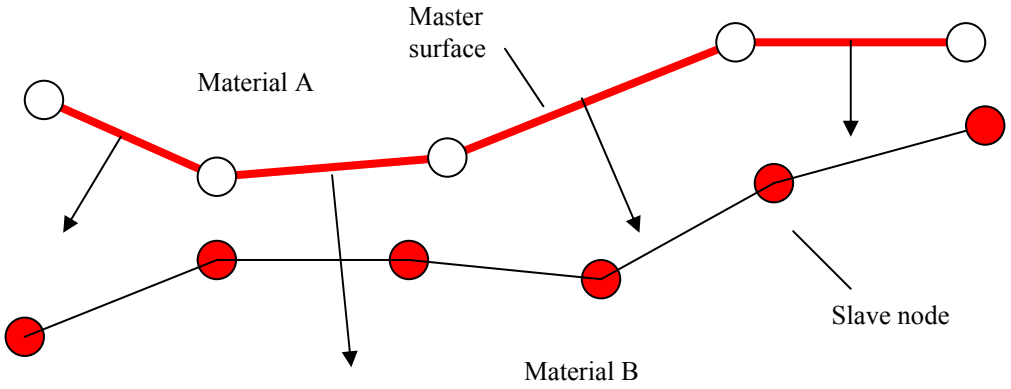

Figure 4.6 Schematic figure of a contact surface.

There are a number of different contact methods used in dynamic FE codes.

- *Penalty method*
- *Kinematic constraint method*
- *Distributed parameter method*

In all methods one side of the interface is designated as the slave side and the other is designated as the master side, see Figure 4.6. The slave nodes are constrained to slide on the master surface after impact and must remain on the master surface until a tensile force develops between the node and the surface.

The *penalty method* is the most common method used in dynamic FE codes. The contact is activated as soon as a slave-node penetrates a master-segment. The *penalty method* consists of springs placed between all slave nodes and the master segment. The force in these springs are calculated with the expressions bellow.

$$\boldsymbol{f}_s = -l\boldsymbol{k}_i\boldsymbol{n}_i \tag{4.1}$$

where $\boldsymbol{f}_s$ is the force applied to the slave node, $\boldsymbol{l}$ is the amount of penetration, $\boldsymbol{k_i}$ is the stiffness factor given in Eq. 4.2 and $\boldsymbol{n_i}$ is the normal vector to the master segment.

$$\boldsymbol{k}_i = \frac{\boldsymbol{f}_{si}\boldsymbol{K}_i\boldsymbol{A}_i^2}{\boldsymbol{V}_i} \tag{4.2}$$

Where $\boldsymbol{f_{si}}$ is a scale factor for the interface stiffness, $\boldsymbol{K_i}$ is the Bulk modulus for the master segment and $\boldsymbol{A_i}$ and $\boldsymbol{V_i}$ is the master element area and volume, respectively.

The contact is activated as soon as a slave node penetrates a master surface. Normally, the contact surface is not the same as the element surface. The element surface is projected in the surface normal direction by a thickness defined as half the shell thickness or approximately 5%-10% of the solid element characteristic length. This could look like Figure 4.7. The definition of the contact thickness can result in slave nodes passing the master surface without being detected. This can happen if the master surface is too convex (see gaps in the Figure).

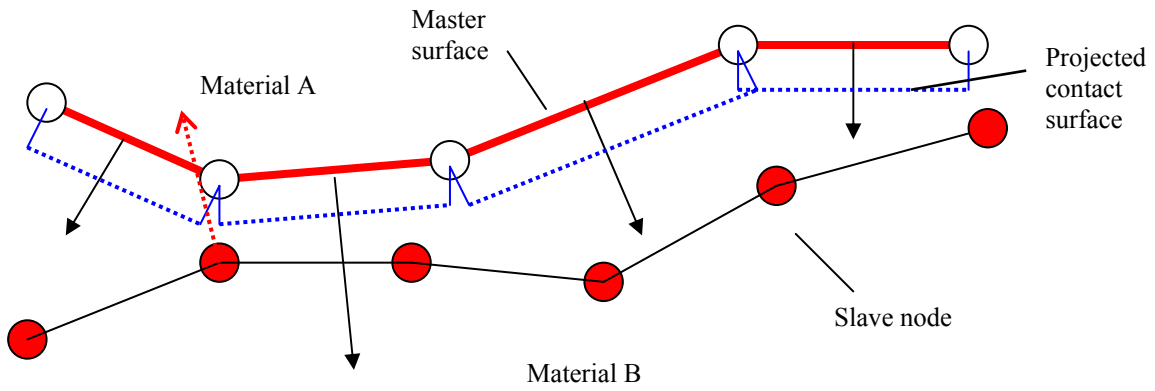

Figure 4.7 Schematic figure showing the master segment thickness defined in the normal direction. The dotted arrow shows the possibility for the slave node to penetrate the master surface without detection.

5 Bibliography

There are a large number of books written on the subject of finite element theory, analysis and computation. Some gives a good overview while others are more specialised. New books on the subject are published frequently and there are vast sources of information on this particular topic. Below are four references which were mainly used to compile the present lecture notes. The first one (Szabó & Babuška) is very theoretical and mathematical. The book by Ottosen & Petersson is quite comprehensive in the basic derivations of various finite element formulations for structural and heat transfer problems. Cook deals mainly with finite elements structural mechanics problems and is as such a very comprehensive volume.

Szabó B. and Babuška I., *The Finite Element Method*, John Wiley & Sons, New York, 1991.

Ottosen N. and Petersson H., *Introduction to the Finite Element Method*, Prentice Hall, New York, 1992.

Cook R.D., *Concepts and Applications of Finite Element Analysis*, 2:nd edition, John Wiley & Sons, New York, 1981.

Bathe KJ, Finite Element Procedures, Prinice-Hall, New Jersey, 1996.