

Isogeometric Analysis 2011  
Austin, Texas – January 13-15, 2011

# Overview on Use of *FEAP*

[www.ce.berkeley.edu/~rlt/presentations/](http://www.ce.berkeley.edu/~rlt/presentations/)

**Robert L. Taylor**

Department of Civil & Environmental Engineering  
University of California, Berkeley

12 January 2011

# Solving Problems using FEAP

- This lecture presents:
  - Summary of capabilities of *FEAP*.
  - Input data file structure.
  - Problem solution modes.
  - Graphics capability.
  - Use with T-splines meshing.

## Background Reading

- **Additional information available in:**

- On line manuals for *FEAP* at

- [www.ce.berkeley.edu/feap](http://www.ce.berkeley.edu/feap)

- Manuals available: Installation, User, Contact, Parallel, Example, Programmer).

- Books on FEM: O.C. Zienkiewicz and R.L. Taylor, *The Finite Element Method*, 6<sup>th</sup> edition, Elsevier Butterworth-Heinemann, Oxford, 2005.

- See last chapter in Basis and Solids volumes.

- *FEAP* distributed under license by UC.

- Small version *FEAPpv* available free: [www.berkeley.edu/feap/feappv](http://www.berkeley.edu/feap/feappv).

# FEAP Program Capabilities

- *FEAP*: Finite Element Analys Program.
- Solves problems formulated by a *finite element method*.
- Problems can be:
  - Linear or non-linear.
  - Static, quasi-static or transient.
  - Coupled (multi-physics) homogeneous or partitioned.
  - Parallel (using PETSc)
  - Multi-scale  $FE^2$  form.
- Solution in *batch* or *interactive* mode.
- Output: Print, graphics (screen or PostScript), time history files.

## **FEAP Program Capabilities**

- Finite element library includes:
  - Small and finite deformation analysis of solids.
  - Thermal analysis of solids.
  - Small and large displacement frame (bending, shear & axial deformation).
  - Small and large displacement membrane.
  - Small displacement shell.
- Elements can be used in 1-d, 2-d & 3-d analyses.

## FEAP Program Capabilities

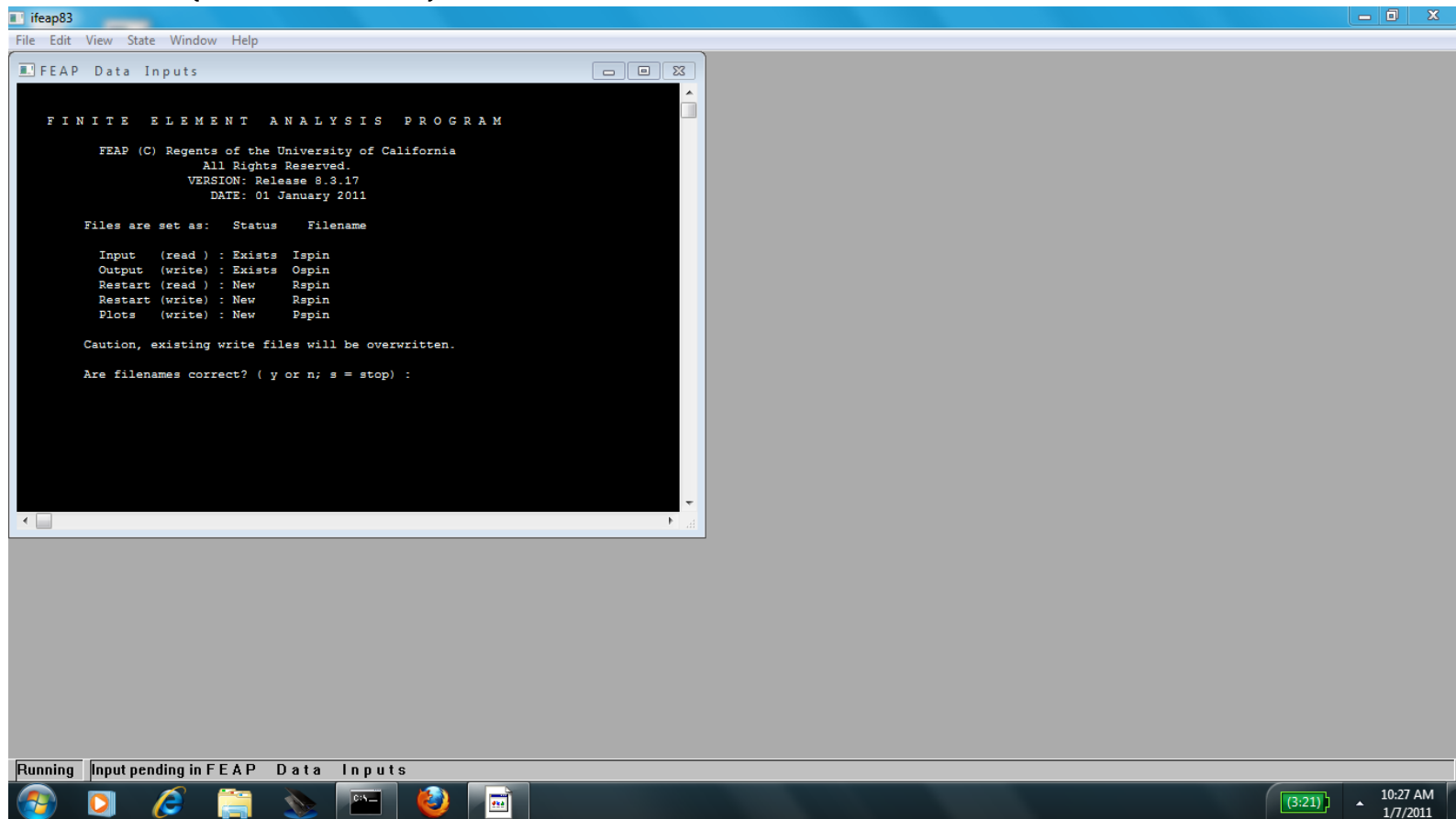
- Small deformation elements use material library containing:
  - Elastic (Isotropic & orthotropic with thermal expansion)
  - Visco-elastic (isotropic deviatoric only; complex moduli)
  - Elasto-plastic –  $J_2$  with isotropic and kinematic hardening
  - Generalized plasticity –  $J_2$  model.
  - RVE - interface (multi-scale use)
  - User model interface.

## FEAP Program Capabilities

- Finite deformation elements use material library containing:
  - Elastic (Neohookean, Mooney-Rivlin, St.Venant-Kirchhoff, Ogden, Fung, Arruda-Boyce, Yeoh)
  - Visco-elastic (isotropic deviatoric with damage)
  - Elasto-plastic –  $J_2$  with isotropic and kinematic hardening
  - RVE - interface (multi-scale use)
  - User model interface.

# FEAP Program Capabilities

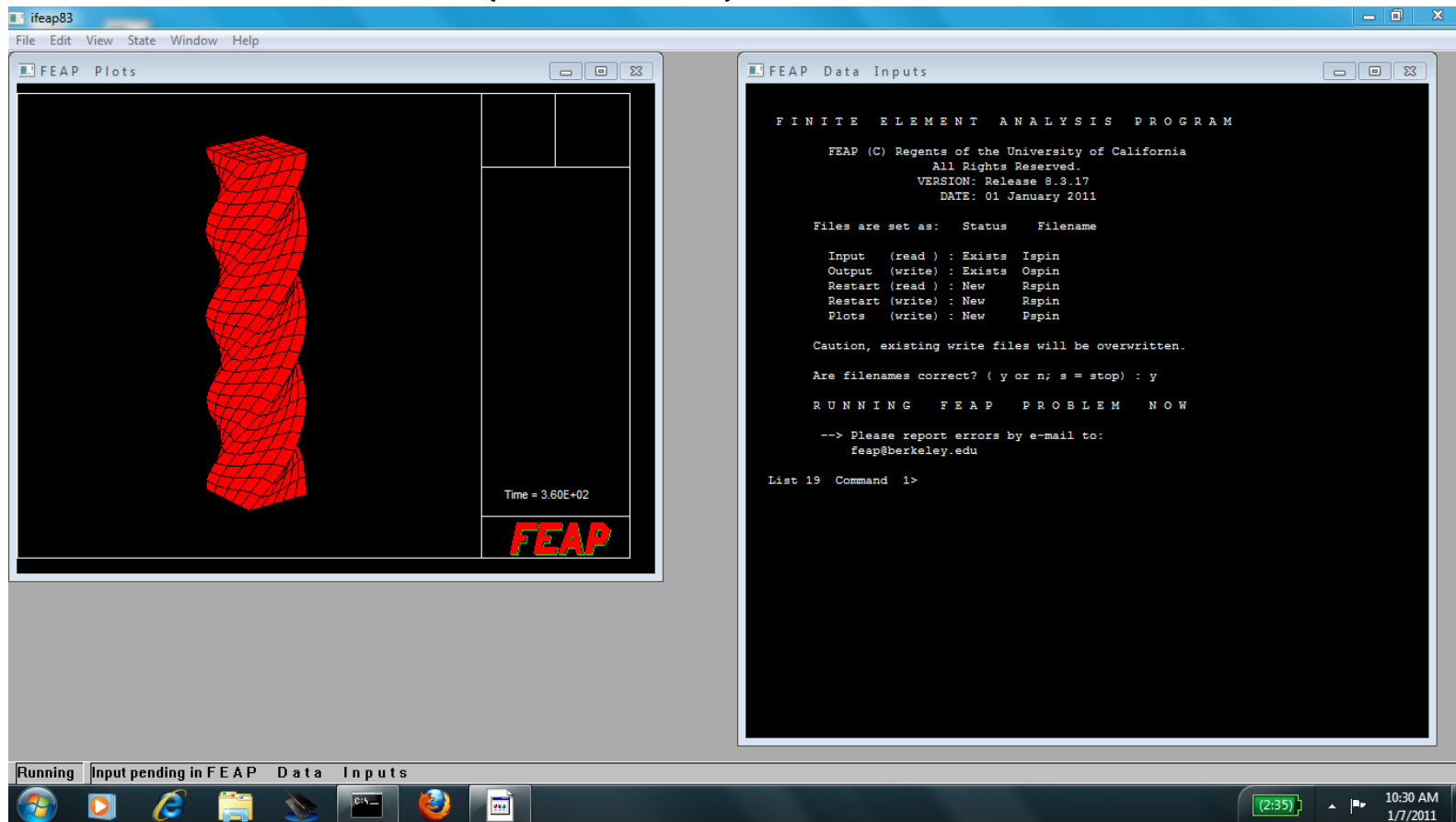
- Startup (Windows):





# FEAP Program Capabilities

- After plot initiation (Windows):



# FEAP Program Capabilities

- Startup & Graphics (Unix/Mac):

```
Terminal — feap — 80x24
r1t:~/feap/ver83/nurbfep/inputs/spin $ feap

  F I N I T E   E L E M E N T   A N A L Y S I S   P R O G R A M

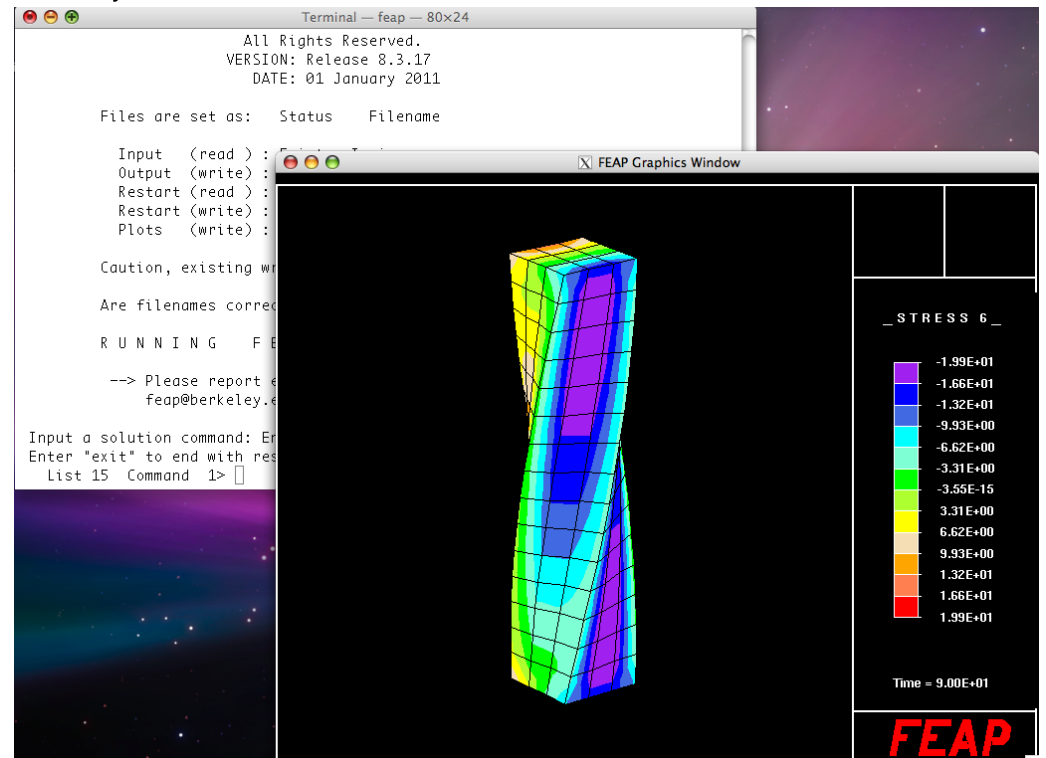
    FEAP (C) Regents of the University of California
      All Rights Reserved.
    VERSION: Release 8.3.17
      DATE: 01 January 2011

Files are set as:  Status   Filename

Input  (read) : Exists  Ispin
Output (write) : Exists  Ospin
Restart (read) : New     Rspin
Restart (write) : New     Rspin
Plots  (write) : New     Pspin

Caution, existing write files will be overwritten.

Are filenames correct? ( y or n; s = stop) : █
```



## FEAP Input Data Files

- Description of mesh by *input data file(s)*.
- Basic structure of file:
  - Control data
  - Mesh inputs (nodes, elements, material data, loads, etc.)
    - \* Nodes: 1:NUMNP; Elements 1:NUMEL; Materials 1:NUMMAT
  - Mesh manipulation (e.g., merge parts, link nodes, etc.)
  - Contact interface description (none with NURBS or T-splines).
  - Solution using *command language* statements.

# FEAP Input Data Files

- Control data:

```
FEAP * * title for output file  
NUMNP NUMEL NUMMAT NDIM NDOF NEN
```

where

- FEAP     – Indicates start of problem.
  - NUMNP    – Number of nodes
  - NUMEL    – Number of elements
  - NUMMAT   – Number of material sets
  - NDIM     – Spatial dimension of mesh
  - NDOF     – Maximum number d.o.f/node
  - NEN       – Maximum number nodes/element
- Often, NUMNP, NUMEL, NUMMAT can be input as zero (0).  
For T-spline solution: all can be omitted.

## FEAP Input Data Files

- Example: Mesh for a *patch test*.

```
FEAP * * Patch test
  0 0 0  2 2 4
```

```
COORdinateS
```

```
 1  0  0.0  0.0
 2  0 10.0  0.0
 3  0 10.0 10.0
 4  0  0.0 10.0
 5  0  2.0  2.0
 6  0  7.5  3.0
 7  0  6.5  7.0
 8  0  2.5  6.5
```

```
ELEMeNtS
```

```
 1  1  1  1  2  6  5
 4  0  1  4  1  5  8
 5  0  1  5  6  7  8
```

```
BOUNDary
```

```
 1  0  1  1
 4  0  1  0
```

```
FORCe
```

```
 2  0 100.0 0.0
 3  0 100.0 0.0
```

```
MATeRial 1
```

```
  solid
```

```
    elastic isotropic 200e9 0.3
```

```
END
```

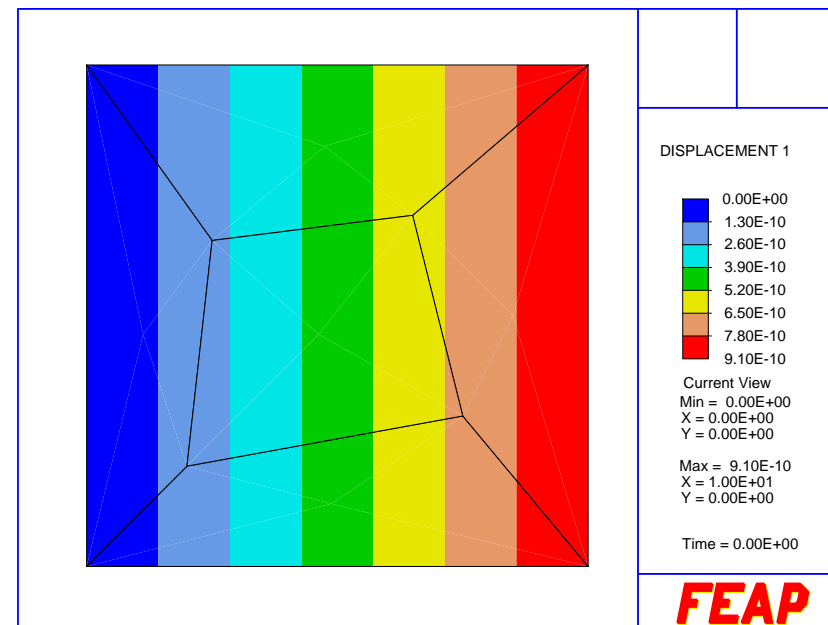
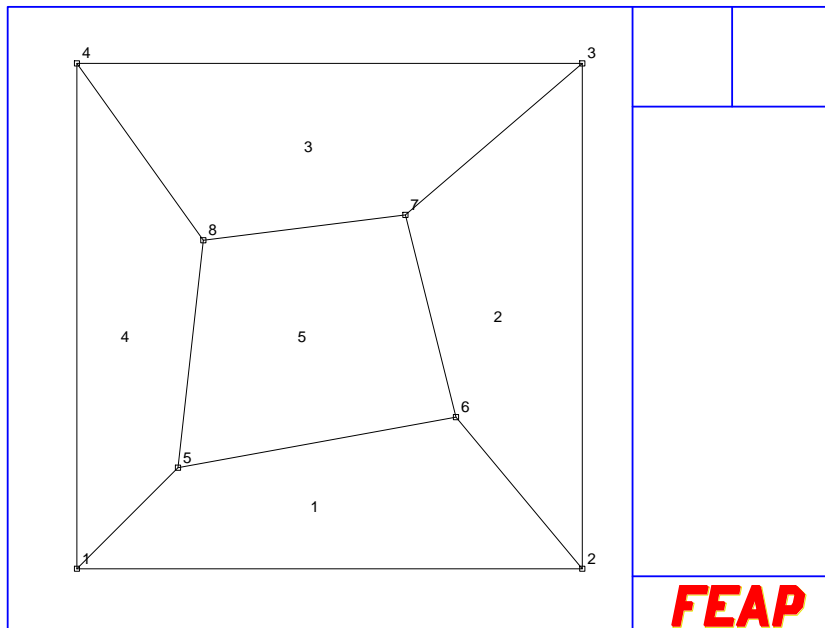
```
INTERactive
```

```
STOP
```

- *FEAP* counts: 8 nodes, 5 elements & 1 material.
- Uses **first 4-characters** of text.

# FEAP Input Data Files

- Above input statements give the mesh and after solution the plot may be added.



- Some basic solution commands described later.

# FEAP Input Data Files

- Mesh data can be split into parts using **INCLude** option.
- Example: Patch test.

```
FEAP * * Patch test
      0 0 0  2 2 4

INCLude Ipmesh

MATERial 1
      solid
      elastic isotropic 200e9 0.3

END
INTERactive
STOP
```

File Ipmesh contains

COORDinates

```
1  0  0.0  0.0
2  0 10.0  0.0
.... etc. ....
```

FORCe

```
2  0  100.0  0.0
3  0  100.0  0.0
```

etc.

- Similar form used later to interface T-spline file.

## FEAP Input Data Files

- *FEAP* commands interpret 4 characters.
- *FEAP* numerical data given by:
  - Numerical: 1, 1.0, -5.3e-3
  - Parameters (limited to 2-characters):  
aa, z1, e0
  - Expressions:  $x_0 + r \cdot \sin(30)$ ,  $e_1 \cdot b \cdot h^3 / 12$   
Operations: +, -, \*, /, ^  
Functions: sin, abs, exp, atan, ...
- Parameters set by commands:

```
PARAMeter
  x0 = 5.56
  pi = acos(-1.0)
  ! End with blank line.
```

Note: Data following ! not used by *FEAP*.



## FEAP Input Data Files

- All data read by parser.
- *FEAP* counts number of nodes, elements and material sets.  
Can be zero on control statement.
- Standard data record structure:
  - Unformatted input: Field widths limited to 15 characters
  - No more than 16 data items per record.
  - Data items separated by: comma (,); space ( ); equal (=).
  - Blank characters ignored except in expressions.  
Note: `x + r*sind(3)` would be read as 3 fields.
  - Data after ! ignored (comment)

## FEAP Input Data Files

- Nodal coordinate and element connection

- Coordinates by node number (COORD):

NUMBER N\_GEN X\_1 X\_2 X\_3

\* NUMBER = node number

\* N\_GEN = increment to next node

\* X\_i = value (i = 1,NDIM)

- Elements by number (ELEM):

NUMBER N\_GEN N\_MAT IX\_1 .. IX\_NEL

\* NUMBER = element number

\* N\_GEN = increment to nodes

\* N\_MAT = material set identifier

\* IX\_i = node number (i = 1,NEL; NEL ≤ NEN)

## FEAP Input Data Files

- Nodal data specified by:

- Node number

FORCe

5 0 5.0 -2.3

sets node 5 force:  $F_1 = 5$ ,  $F_2 = -2.3$ .

- Edge coordinate value

EB0Undary

2 5.0 1 0

sets BC code:  $ID_1 = 1$  and  $ID_2 = 0$  for nodes with  $x_2 = 5$ .

Note: A zero has unknown solution value.

- Coordinate value

CFORce

node 5.0 0.0 5.0 -2.3

sets force at node closest to  $x_1 = 5$ ,  $x_2 = 0$  to  $F_1 = 5$  and  $F_2 = -2.3$ .

# FEAP Input Data Files

- Material data sets
  - Specifies element type: SOLId, PLATe, SHELl, FRAME, TRUSs, GAP, POINt, THERmal, USER.
  - Defines associated element group.
  - Defines degree-of-freedom assignments.
  - Describes constitutive model property values.
  - Defines finite element formulation (e.g., displacement, mixed, small or finite deformation, etc.).
  - Defines other element properties (e.g., quadrature order, body loading, etc.).
  - Specify NURBS or T-spline interpolation and quadratures.

## FEAP Input Data Files

- Material set: Form for linear elastic T-spline solid element

```
MATeRial ma
  SOLId
    ELAStic ISOTropic E nu
    T-SPline interp  q1 q2 q3  ! (or NURBS)
```

- Use standard displacement model to describe elements where  $N\_MAT = MA$ .
- Elastic properties are *isotropic* with *parameters* set by E (Young's modulus) and *nu* (Poisson's ratio).
- Specifies T-spline interpolation with  $q1, q2, q3$  quadrature.
- Small deformation element.

# FEAP Input Data Files

- Finite deformation solids

- Set explicitly:

```
MATeRial ma
  SOLId
    ELAStic ISOTropic E nu
    T-Spline quadratrue q q q
  FINIte
```

- Set implicitly by model

```
MATeRial ma
  SOLId
    ELAStic NEOHook      E nu
    T-Spline quadratrue q q q
```

- Small deformation set by SMALL.

# NURBS Isogeometric Modeling

## NURBS Analysis procedure in FEAP

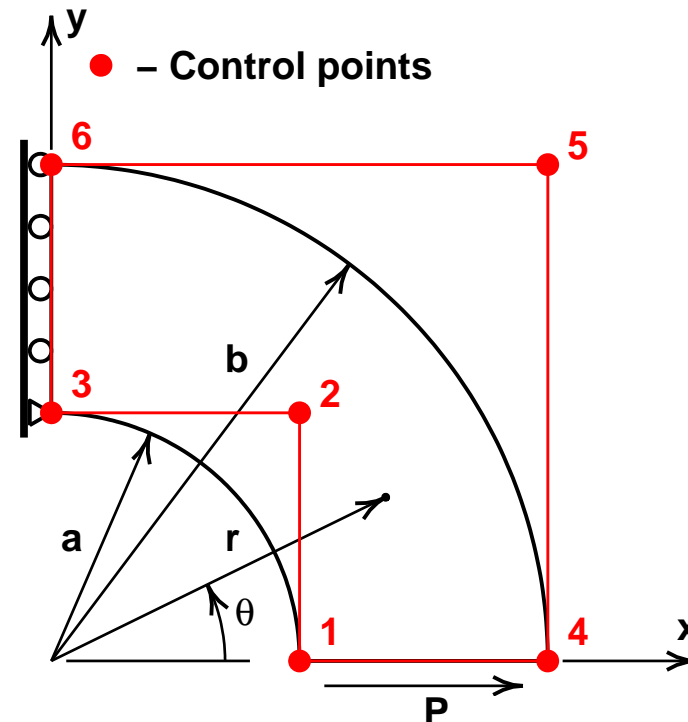
- Define coarse set of control points, knots, 1-d knot-point list, side-patch description:
- Example: Curved beam – specification of NURBS and knots

NURBs

1	0	5.0	0.0	1.00
2	0	5.0	5.0	$\sqrt{2}/2$
3	0	0.0	5.0	1.00
4	0	10.0	0.0	1.00
5	0	10.0	10.0	$\sqrt{2}/2$
6	0	0.0	10.0	1.00

KNOTS

knot	1	0.00	0.00	1.00	1.00		
knot	2	0.00	0.00	0.00	1.00	1.00	1.00



# NURBS Isogeometric Modeling

- Example: Continued – Specification of sides and patch

NSIDes

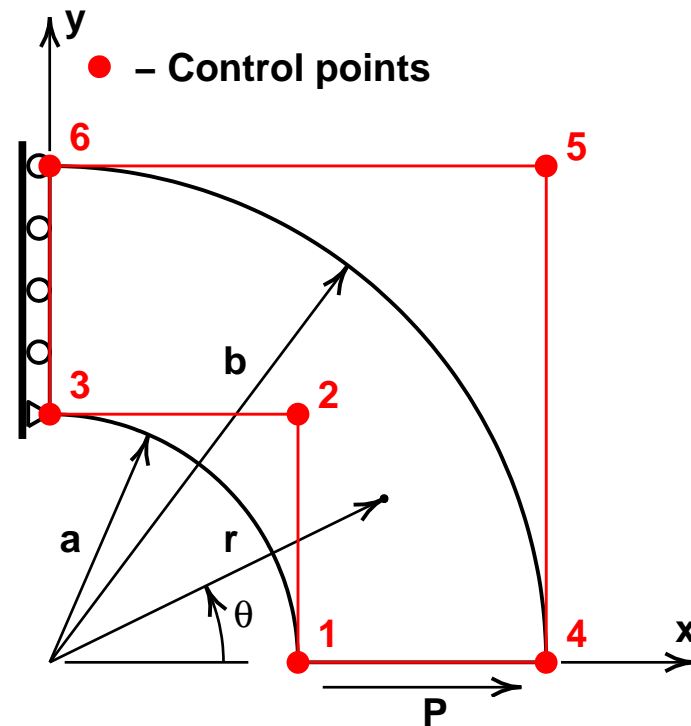
side 1 0 2 1 2 3

side 2 0 2 4 5 6

side 3 0 1 1 4

NBL0ck

block 2 1 3

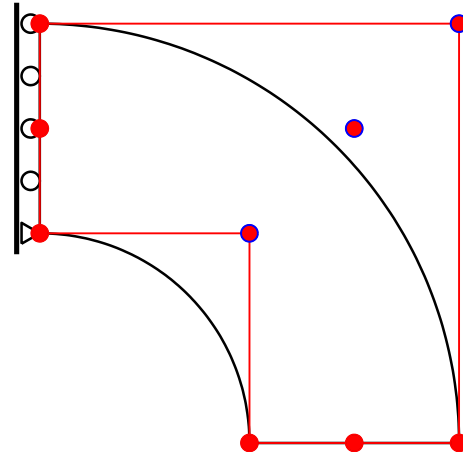


- Need to add material properties, loading and boundary conditions. Use standard *FEAP* commands for most.

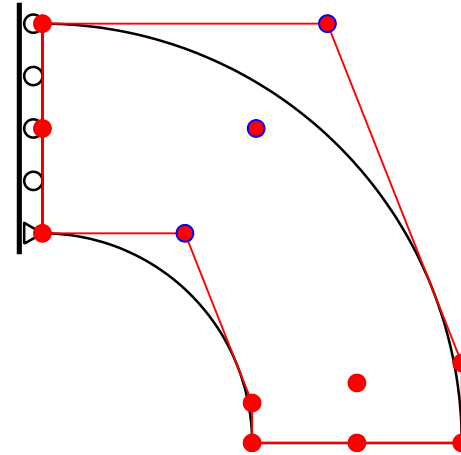


# NURBS Isogeometric Modeling

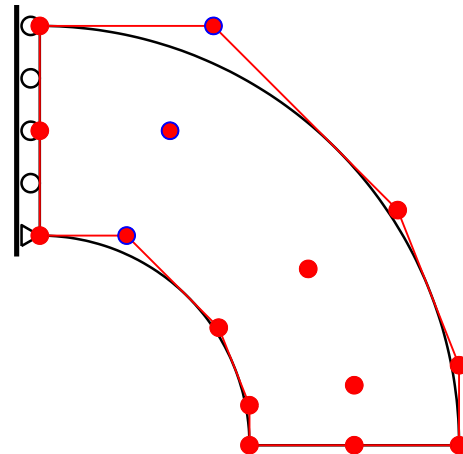
$k$ -refinement in circumferential direction



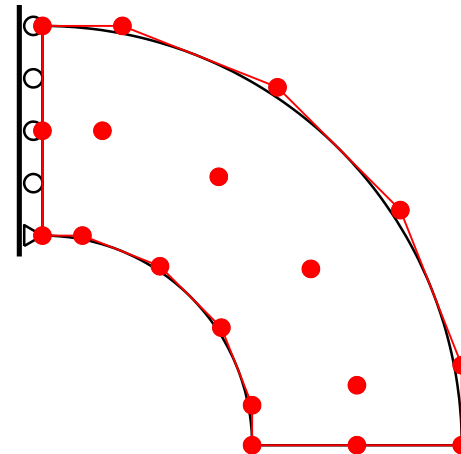
(a) Elevate radial knot



(b) Insert knot 1



(c) Insert knot 2



(d) Insert knot 3

## NURBS Isogeometric Modeling

- Elevate NURBS block 1, direction 3, 2 orders.

```
include Ispini
batch
  elevate init
  elevate block 1 3 2
  elevate end
end
```

- Knot insertions for NURBS block 1 in direction 3.

```
parameter
  kk = 0

loop,4
  parameter
    kk = kk + 0.25

    include Ispini
    batch
      insert init
      insert block 1 3 kk 1  ! Last entry is number of times
      insert end
    end
  next
```

# NURBS Isogeometric Modeling

## *k*-refinement in FEAP (Cont.)

- Each elevation or insertion creates a flat mesh NURB\_mesh
- Can be used to describe problems recursively in input file using  
INCLUDE NURB\_mesh

- Use loops (in input file) to perform repeated insertions

```
LOOP,9
```

```
PARAMeter
```

```
    d = d + 0.1
```

```
INCLUDE NURB_mesh
```

```
BATCH
```

```
    INSERT INITIALize
```

```
    INSERT KNOT 1 d 1
```

```
    INSERT END
```

```
END
```

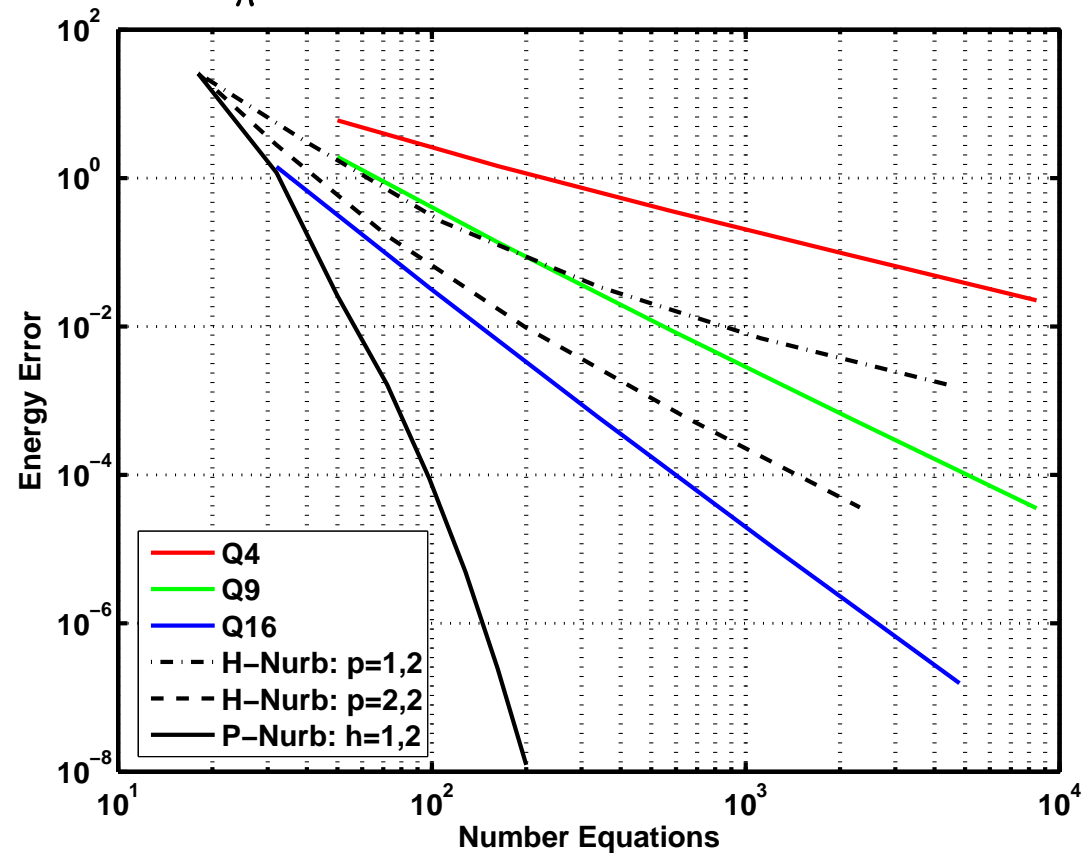
```
NEXT
```

inserts knot 1 nine times at intervals of 0.1 units.

# NURBS Isogeometric Modeling

- Curved beam under end shear results, exact energy

$$E_{ex} = \frac{1}{\pi} [\log 2 - 06] = 0.02964966844238$$



Curved beam subjected to end shear. Energy error.

# NURBS Isogeometric Modeling

## Confined tension strip

- Consider rectangular  $8 \times 8$  with unit circular hole.
- Lateral boundaries restrained and strip stretched by uniform displacement
- Material given by modified Neo-Hookean model

$$W = \frac{1}{4}K \left( J^2 - 1 - 2 \log J \right) + \frac{1}{2}G \left( J^{-2/3} \mathbf{b} : \mathbf{1} - 3 \right)$$

where  $\mathbf{b} = \mathbf{F} \mathbf{F}^T$  and properties set to:

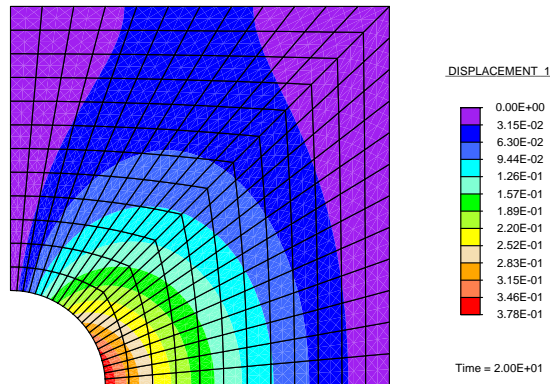
$$K = 1.6 \times 10^8 \quad \text{and} \quad G = 3.2 \times 10^4$$

Results in a nearly incompressible behavior.

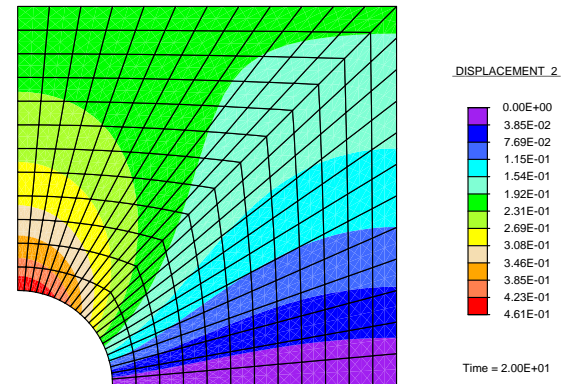
- Model by Q9/P1 standard mixed elements and  $\bar{F}$  NURBS

# NURBS Isogeometric Modeling

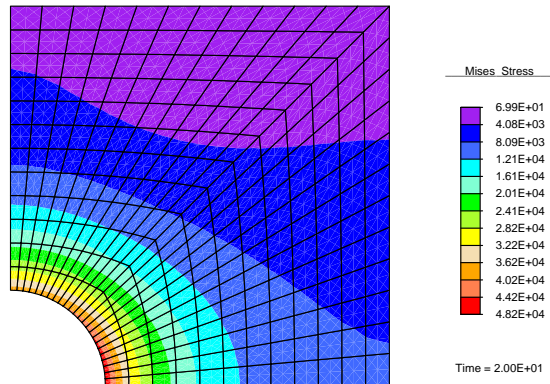
## Confined tension strip: Q2/P1 Elements



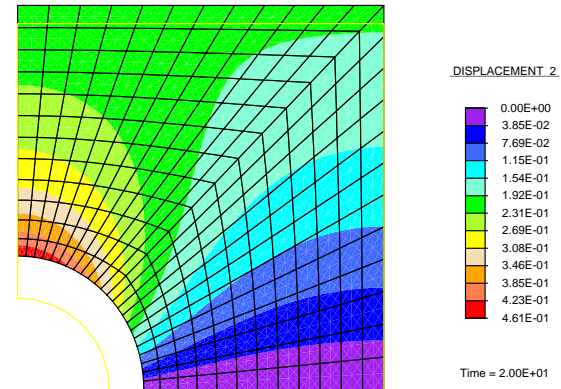
(a) u-displacement



(b) v-displacement



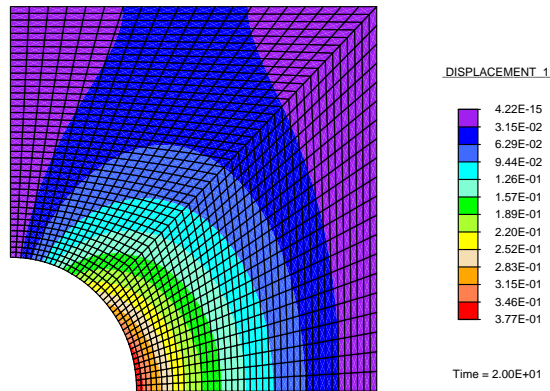
(c) Mises stress



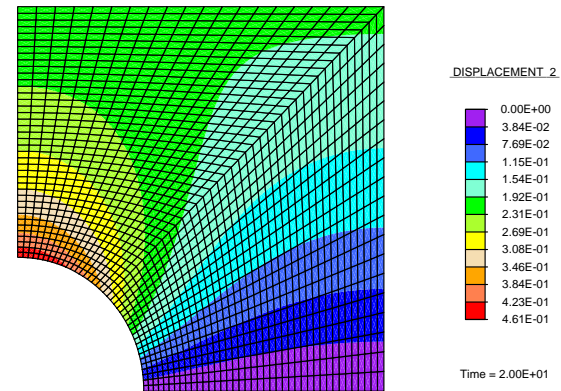
(d) Deformed shape

# NURBS Isogeometric Modeling

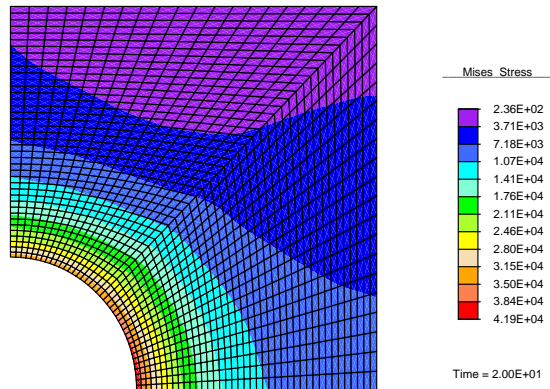
Confined tension strip: Q2/Q1/Q1  $u - p - \theta$  NURBS



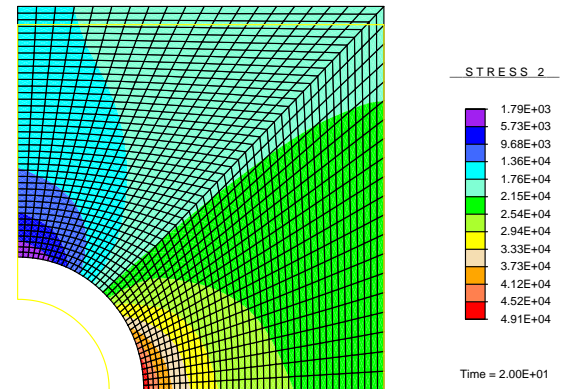
(a) u-displacement



(b) v-displacement



(c) Mises stress

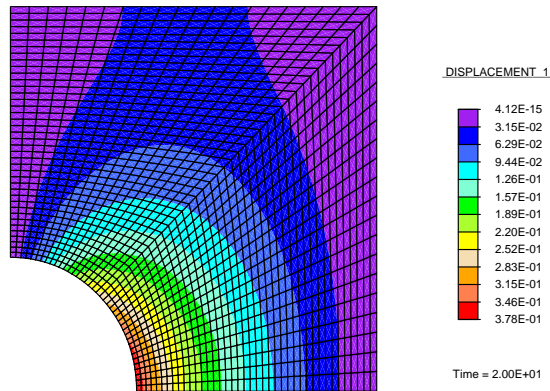


(d) Deformed shape

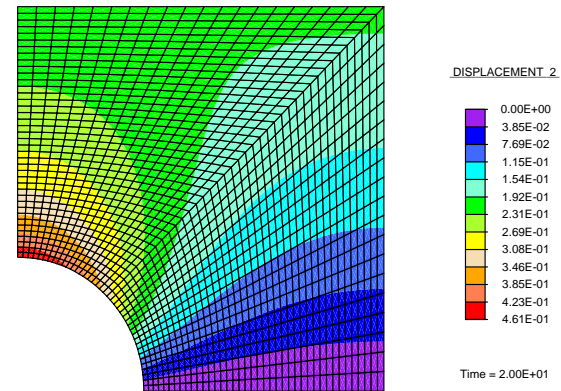


# NURBS Isogeometric Modeling

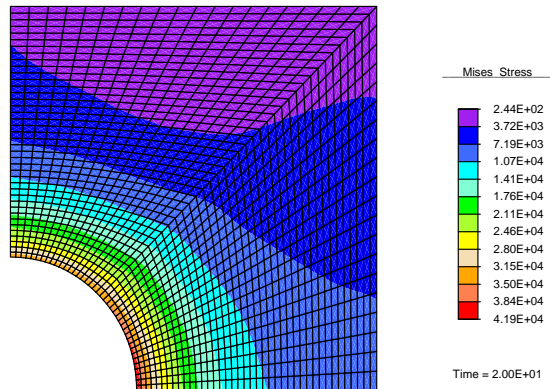
Confined tension strip: Q2/Q1/P1  $u - p - \theta$  NURBS



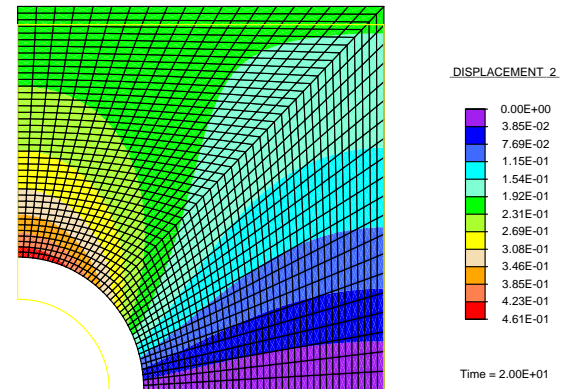
(a) u-displacement



(b) v-displacement



(c) Mises stress



(d) Deformed shape



# NURBS Isogeometric Modeling

## Elastic-Plastic Necking

- Plane strain strip under uniform extension
- Finite strain elastic-plastic model with  $J_2$  yield function.
- Material properties are:  
 $K = 164.206$  ;  $G = 80.1938$  ;  $\sigma_0 = 0.45$  ;  $\sigma_\infty = 0.715$  ;  $h = 0.12924$  ;

Uniaxial yield stress given by

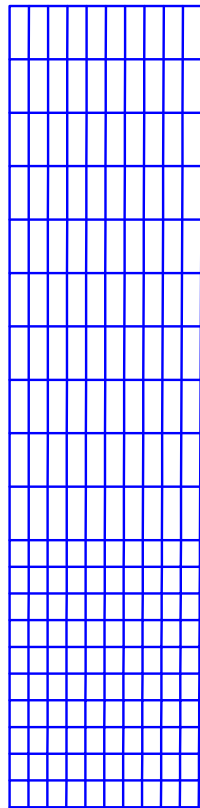
$$\sigma_y = \sigma_\infty + (\sigma_0 - \sigma_\infty) \exp(-\beta e_p) + \sqrt{\frac{2}{3}} h e_p$$

where  $e_p$  effective plastic strain and  $h$  isotropic hardening.

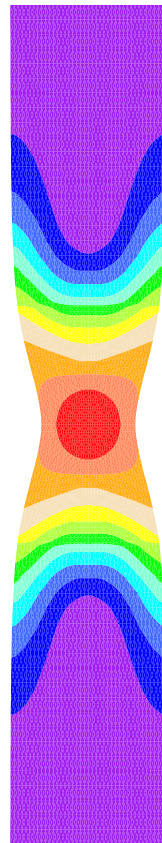
- Geometry: Length is 53.334 and width is 12.826. Symmetry used for one quadrant model (reduce center to 0.982 of width).
- Use quadratic NURBS with 10 knots in width and 20 in length. Analysis also performed with Q1/P0 & Q2/P1 elements.

# NURBS Isogeometric Modeling

## Elastic-Plastic Necking Example

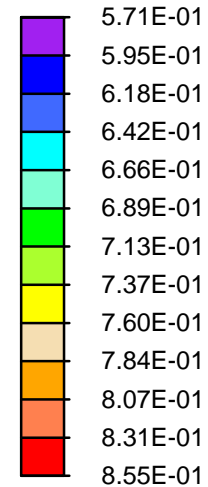


(a) Mesh



(b) Mises stress at full extension

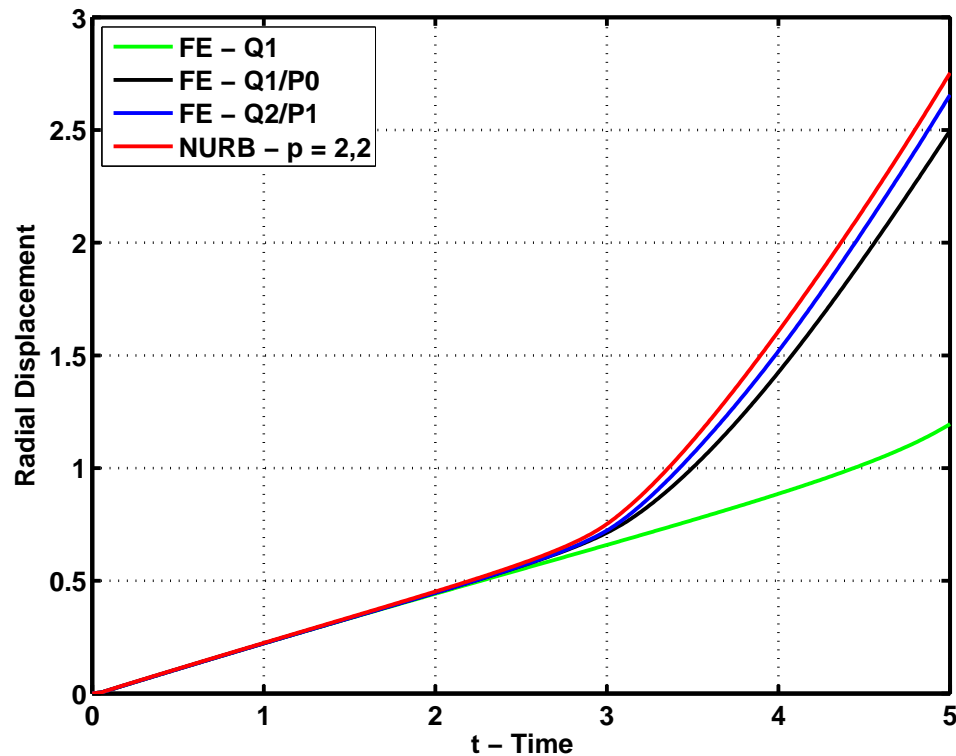
Mises Stress



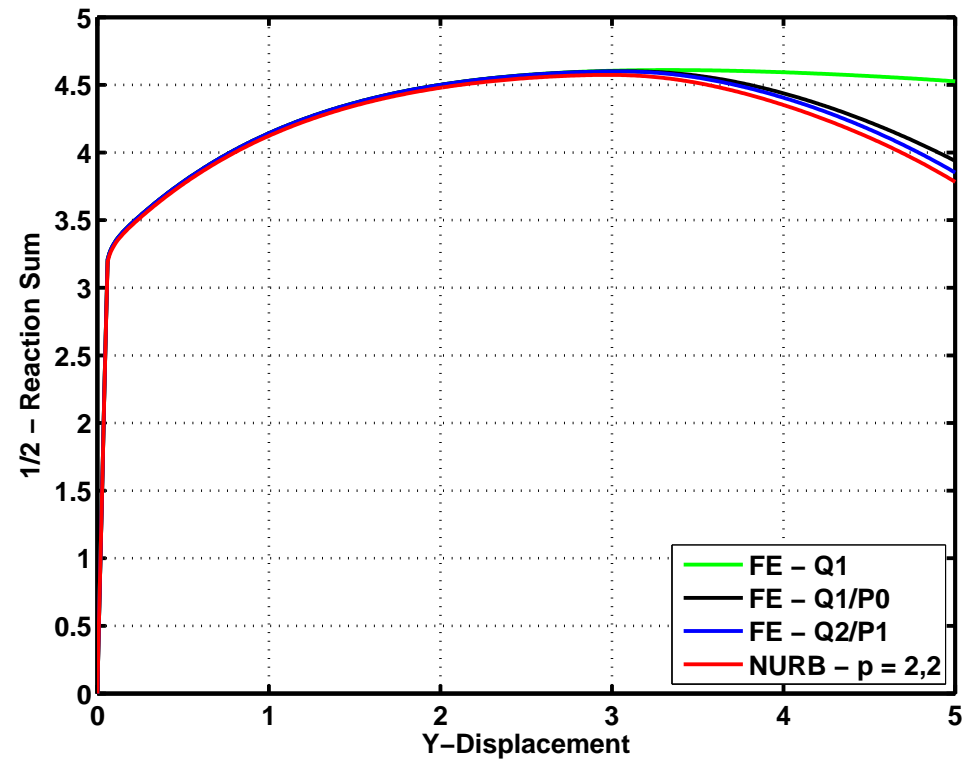
Time = 5.00E+00

# NURBS Isogeometric Modeling

## Elastic-Plastic Necking Example



(c) Necking displacement



(d) Load-displacement

## FEAP Command Language Solution

- After FE mesh described, users responsible for solution steps.
- May be given in **BATCh** or **INTERactive** mode.
- Example of batch solution:

```
BATCh
  TANGent          ! Form  "K"          Tangent
  FORM            ! Form  "R = F - P(u)" Residual
  SOLVe           ! Solve "K du = R"
END
```

or

```
BATCh
  TANGent,,1       ! Means same as above
END
```

- Interactive solution with:  
INTERactive
- Can give more than one batch and/or interactive set.

## FEAP Command Language Solution

- LOOP-NEXT can be used in solution mode also.
- Useful for iterative or time stepping solutions.
- Time stepping for linear transient solution:

```
TRANSient      ! Activate Newmark
DT,,dt         ! Set time step
LOOP,,nt       ! DO 'nt' steps
  TIME         ! Advance time
  TANGent,,1   ! Solve step
  ....        ! Outputs/plots
NEXT           ! End of loop
...
```

- Iterative solution for non-linear problem

```
LOOP,,ni       ! DO 'ni' iterations
  TANGent,,1   ! Solve iterate
NEXT           ! End of loop
...
```

## FEAP Command Language Solution

- Transient non-linear solution combines both:

```
TRANSient
DT,,dt
LOOP,,nt
  TIME
  LOOP,,ni
    TANGent,,1
  NEXT
  ....
NEXT
...
```

- Note: All command language statements have form:

```
TEXT TEXT Value Value Value
```

- Example: Solution step with line search:

```
TANGent,LINE_search,1
```

- *FEAP* can have unsymmetric tangent by:

```
UTANGent,,1
```

## FEAP Command Language Solution

- Using *FEAP* in interactive mode.
- Permits easy view of solution at each command step.
- Graphical form using plot options.  
Give only PLOT then individual commands.
- Look at properties of an element matrix.
  - Use: EPRInt to print last element array.
  - Use: EIGEn,<VECTor>,kk to print eigen-pairs for element 'kk'.
- Use: HELP to see commands (MANUa1,,3 to get all).
- Use: SHOW,<option> to see size, dictionary, individual arrays.
- Use: HIST to see previous commands used; also to re-execute previous command sets.

HIST,,3,12 ! Redo commands 3 to 12

## FEAP Command Language Solution

- Example of EIGE from 8-node brick:

```
List 16  Command 1> eige,,5
*Command 1 * eige          v: 5.00      0.00      0.00
                                t=      8.44      0.27
Eigenvalues for STIFFNESS of element 5
```

Matrix: Eigenvalue

row/col	1	2	3	4	5	6
1	6.174E+01	2.472E+01	2.465E+01	2.389E+01	2.323E+01	2.290E+01
2	2.179E+01	2.051E+01	1.891E+01	1.550E+01	1.218E+01	1.195E+01
3	1.165E+01	7.279E+00	6.691E+00	6.134E+00	4.060E+00	3.971E+00

- Note: No zero eigenvalues. Due to geometric stiffness!



## FEAP Graphical Features

- Use: PLOT to enter graphics mode.  
(Exit back to command language using 'e').

- Use: HELP to see list of commands

- Plot contours:

```
CONTour 1    ! Display contour of DOF 1
STREss  1    ! Display stress component 1
```

- USE: POSTscript to output file for later use

```
POST        ! Start PostScript file
MESH
OUTLine
POST        ! End PostScript file
```

Produces set of files: FeapAAAA.eps, etc.

## FEAP Graphical Features

- In command language mode can also produce files of time history data

```
BATCh
  TPL0t,,int    ! Output each 'int' steps
END
DISPl  node    dof
STREss element component
SHOW                ! List to output file
```

## Problem Solving with FEAP

- Running multiple problems:
- Prepare normal *FEAP* input files: Iprobx  
(Here x denotes a problem number)
- Prepare master input file with data:

```
INCLude Iprob1
INCLude Iprob2
...
INCLude Iprobn
STOP
```
- Any STOP commands in Iprobx will be ignored.
- File names must be given in case sensitive mode.

# Problem Solving with FEAP

- Example: Problem with mesh refinement

Master file

```
PARAMeter
  n = 2

INCLude Isquare
PARAMeter
  n = 4

INCLude Isquare
...
PARAMeter
  n = 64

INCLude Isquare
STOP
```

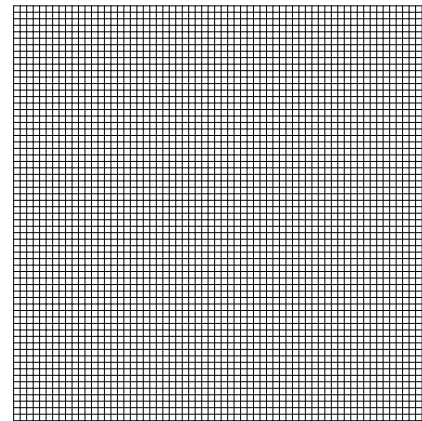
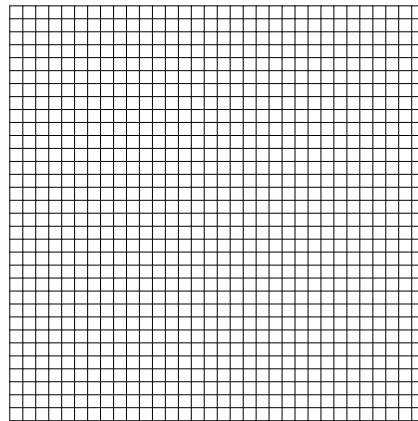
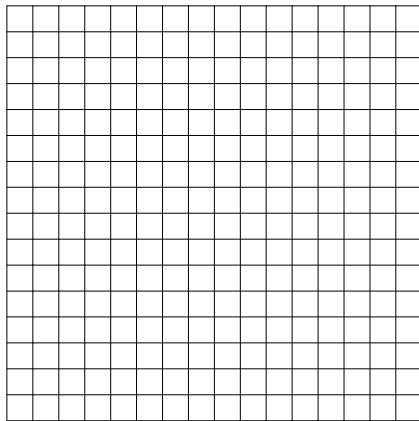
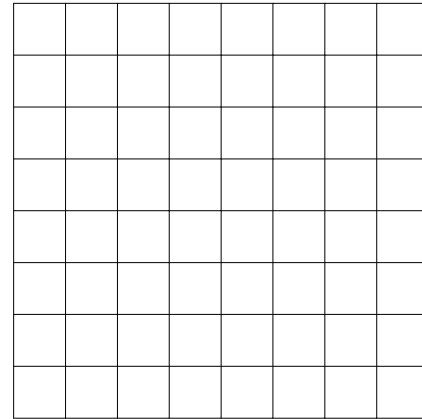
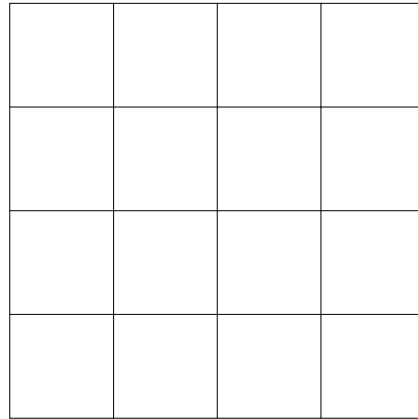
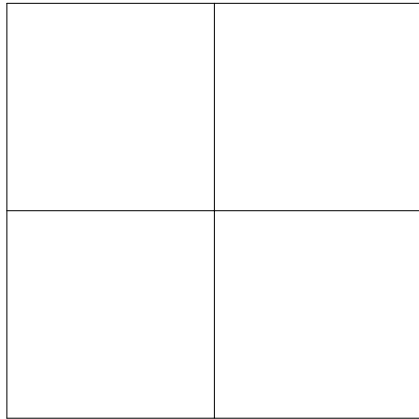
Problem file Isquare

```
FEAP * * Mesh for square
0 0 0 2 2 4
BLOCK
  CART n n
    1 0 0
    2 5 0
    3 5 5
    4 0 5

...
END
BATCh
  TANG,,1
  DISP,,1
END
```

# Problem Solving with FEAP

- Produces sequence of meshes:



## Problem Solving with FEAP

- To use this feature mesh must
  - Either not have explicit definition of a node number or master file must provide all node numbers as parameters.
  - Let *FEAP* do counting for number of elements and nodes.
- Parameters may describe any data (including node and element numbers). For example, range of Poisson ratios may be parameters.
- Run one coarse mesh in INTERactive mode to establish the command language statements needed.
- Test the file on two coarse meshes (or two parameters) to ensure all works as *planned*.
- A NOPRint in mesh & batch execution file reduces output file size.

## Programming Elements for FEAP in Fortran

- User elements all have single subprogram to interface

```
SUBROUTINE ELMTnn(D,UL,XL,IX,TL,S,R,NDF,NDM,NST,ISW)
```

where nn ranges from 01 to 50.

- In addition information is passed using COMMON statements.

```
INTEGER          NUMNP,NUMEL,NUMMAT,NEN,NEQ,IPR
```

```
COMMON /CDATA/  NUMNP,NUMEL,NUMMAT,NEN,NEQ,IPR
```

Example: NEN is used to dimension some arrays.

- Best to use an include statement, e.g.:

```
INCLUDE 'cdata.h'
```

N.B. Windows case insensitive for file name, UNIX is not!

## Programming Elements for FEAP in Fortran

- Main control switch (ISW) values:

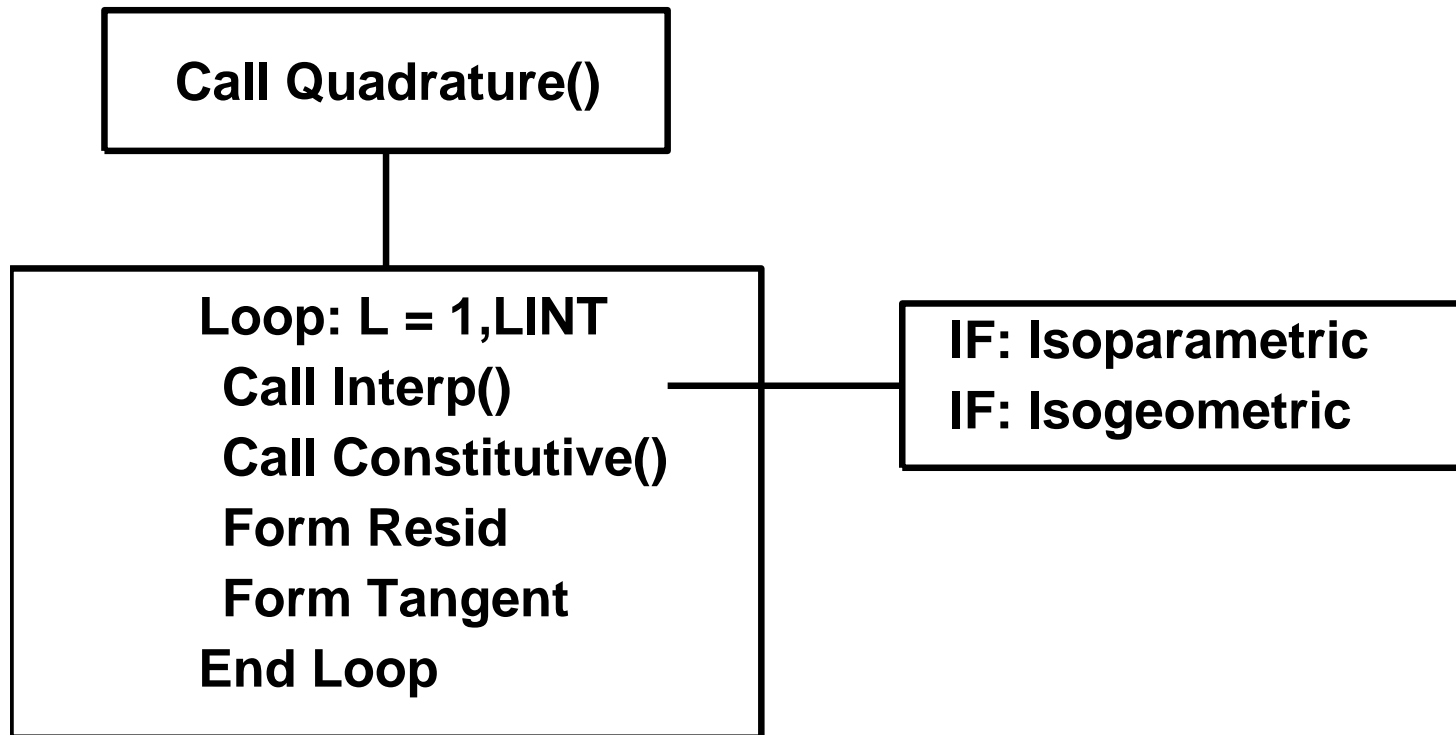
ISW	Description	Command
0	Describe element	SHOW,ELEM,nn
1	Input D(*)	MATE ma
3	Compute stiffness and residual	TANG or UTAN
4	Output values	STRE
5	Compute mass	MASS
6	Compute residual	FORM
8	Project values	PLOT,STRE

- Currently options go to 26 (see Programmer Manual).



# Programming Elements for FEAP in Fortran

- Typical displacement formulation element module



- Use **quadrature** and **interpolation** modules.

## Programming for FEAP in Fortran

- Other user module options exist to modify *FEAP*. Examples:
  - UMESH subprograms for user input data. Permits interfacing to mesh generation programs (e.g., T-Splines & GiD from CIMNE in Barcelona, Spain).
  - UMACR subprograms for user command language statements.
  - UMODEL subprograms for user stress-strain equations.
- UMESH2 used to interface T-spline refinement file to *FEAP*.

## T-spline Solutions with FEAP

- Control records:

```
FEAP * * Problem Title
      0  0  0  0  0  0
```

No number of nodes, etc. needed (sometime NDF needed).

- Material model records:

```
MATerial number
  SOLId or MEMBrane only
    ELASTIC .....
      T-SPline ,, q1 q2 q3
        .....
          ! End of material
```

where q1, q2, q3 are number of quadrature points.

## Input of T-spline File

- To input the T-spline mesh the statements are:

```
T-SPline
  PLOT INTervals = nint
  MATERial number = ma
  FILE = xxxxxxxx.ext  <or xxxxxxxx.txt>
    ! End of T-spline inputs
```

- The PLOT and MATE records are optional.  
If given they **must precede** the FILE record.

## Input of T-spline File

- The plot interval value divides each T-Element into `nint` subintervals in each direction for graphics display.
- Default value for `nint` is 1. Permitted range is 1 to 7.
- The material statement assigns all elements in the T-spline file the material number `ma` (Default is 1).
- Multiple sets of T-SPline files may be included.