

---

# **Computational Mechanics**

## **FEM Praktikum Programmierung**

### **Finite Element Analysis Program [FEAP]**

### **Einführung**

**Wintersemester 2020/2021**



**FEAP:** FE-Programm der *University of California, Berkeley*

- Stand-Alone-Programmsystem
- Lösung vielschichtiger Feldprobleme aus der Festkörpermechanik:
  - statische
  - transiente
  - dynamisch linear
  - dynamisch nichtlinear
- Temperaturfeldberechnungen und thermomechanisch gekoppelte Analysen
- Kontaktformulierungen zwischen Körpern mit inkompatiblen Netzen

**FEAPpv:** Frei zugängliches FE-Programm der *University of California, Berkeley*

- beschränkte Anzahl an Knoten
- beschränkte Anzahl an verfügbaren Elementen
- Eingeschränkter Befehlsumfang gegenüber FEAP

## Internetseite des Entwicklerteams unter Leitung von Prof. Robert Taylor, University of California, Berkeley

<http://www.ce.berkeley.edu/projects/feap/feappv/>

- Unter „Files to download“
- Windows Version 3.1 Revision b für 32 und 64 bit Betriebssysteme (15.04.2012)
- ausführbare Datei *feappv.exe* entpacken

## Internetseite FG Numerische Mechanik

Lehre  $\Rightarrow$  Computational Mechanics  $\Rightarrow$  Vorlesungsmaterial

- Windows Version 2.0 Revision a für 32 bit Betriebssysteme (07.04.2005)
- ausführbare Datei *feappv.exe* entpacken

Nutzer:                      Passwort: compmech

## Preprocessing (Modellaufbau)

- Kein grafischer Preprocessor vorhanden
- Modellaufbau erfolgt über Kommando- und Befehlsstruktur
- Problemformulierung im INPUT-File (ASCII-Datei)
  - Element
  - Knoten
  - Netz
  - Randbedingungen / Belastungen
  - Materialeigenschaften

```
!feap * * Übung ebene Scheibe * *
      0 0 0 2 2 4
      ! 0 0 0 2D-3D DOFs/Knoten Knotenanzahl/Element

global          ! Nur bei 2D
plane strain    ! stress

parameter
n1 = 2           ! Anzahl der Elemente
n2 = 2           ! Anzahl der Elemente
a = 40           ! Abmessung
b = 50           ! Abmessung
d = 1           ! Dicke
f = 1           ! Kraft F
E = 1           ! E-Modul
nu = 1/3.0       ! Querdehnzahl

block           ! Erzeugt einen Block mit kart-Koord
cart n1 n2
  1   0   0     ! Pkt. x-Koord. y-Koord.
  2   a   0     ! Pkt. x-Koord. y-Koord.
```

## Postprocessing (Ergebnisauswertung)

- visuell im Postprocessor mit grafischer Benutzeroberfläche
- EPS-Dateien (Bilddateien)
- Eingaben und Lösungsgrößen werden ins OUTPUT-File geschrieben
- gezieltes Herausschreiben von Größen wie Spannungen, Verschiebungen etc. in ASCII-Datensätze



### Einfacher Solver für nichttransiente statische Probleme

- Eingabe des Solvers in INPUT-File
- Verwendung von sogenannten Batch-Kommandos

```
batch  
  tangent  
  form  
  solve  
end
```

#### tangent

- Bildung der Steifigkeitsmatrix

**K**

#### form

- Aufstellen der rechten Seite

**P**

#### solve

- Lösen des Gleichungssystems

**Ku = P**

## Erzeugen einer Eingabedatei für FEAP (*Preprocessor*)

Anlegen einer Eingabedatei als unformatierte Textdatei (Windows Editor):

z.B. Ibeispiel1.txt → Dateiname *muss* mit großem „I“ beginnen

Eingabedatei enthält Informationen über:

- Geometrie, Knoten, Elemente
- Materialparameter
- Verformungsrandbedingungen, Lasteinwirkung
- Gleichungslöser

## Starten des Programms FEAP (*Hauptprozess*)

- Aufruf der ausführbaren Datei → Angabe Eingabedatei *mit* Endung
- FE-System wird gelöst

## Grafische Oberfläche (*Postprocessor*)

- Ergebnisdarstellung am Bildschirm oder als EPS-Grafik

# FEAP-Dateien

Dateiname	Inhalt
<i>IName</i>	Datenbasis, Einstellungen, Geometrie, Material, Elemente,....
<i>OName</i>	Ergebnisse, Zusammenstellung der Eingabedaten, Ausgabe der Elementspannungen und Verzerrungen, Knotenverschiebungen, etc.
<i>RName, SName</i>	Restartdateien
<i>PNamea.str</i>	ASCII-Datei mit Spannungswerten
<i>PNamea.dis</i>	ASCII-Datei mit Verschiebungswerten
<i>FEAPAAAAA.eps</i>	EPS-Bild vom Grafikplot

## Struktureller Aufbau der INPUT-File

- Unterteilung der Datei in zwei Hauptabschnitte:
  1. Definition des Randwertproblems mit Befehlsblöcken
  2. Blöcke mit „Batch-Kommandos“
- Befehlsblock kann über mehrere Zeilen definiert werden

```
coord» »  
  1 1    0    0»  
  2 0   100    0
```

```
feap * * Beispiel gevouteter Stab  
0 0 0 2D-3D DOFs/Knoten Knotenanzahl/Element
```

```
*  
... Definition des Randwertproblems  
end» » » »
```

```
batch» » » »  
... Batch-Kommando  
end
```

```
inter» Interactive: Grafischer Postprocessor  
stop|
```

- Trennung einer Anweisung innerhalb einer Zeile eines Befehlsblocks mit
  - Komma
  - Leerzeichen
  - Tabulator

**Wichtig: Nach jedem Befehlsblock muss stets eine Leerzeile folgen!**



# Aufbau INPUT-File (2)

1. Kopfzeile
2. Information zur globalen Problemgröße
3. Parameterdefinition (optional)
4. Definition des Randwertproblems (RWP)
5. Solvereinstellungen
6. Steuerung des Postprozesses
7. Interaktiver Modus

```
feap * * Beispiel gevouteter Stab
0 0 0 2 2 2
```

```
parameter
E = 3*10^4
A = 10
```

```
coord > > > >
1 1 0 0
2 0 100 0
```

```
elem > > >
1 0 1 1 2 >
```

```
mate 1 > >
truss
elastic isotropic E
cross section A
```

RWP

```
eboun, set > > >
1 0 1 1 > >
1 100 0 1
```

```
eforce > > >
1 100 20 0 > >
```

```
end > > > >
```

```
batch !solver > >
tangent
form
solve
end
```

```
batch !postprocess > >
disp,all
stre,all
plot cont 1
end
```

```
inter > > > >
stop
```

# Aufbau INPUT-File (3)

## Kopfzeile

- Schlüsselwort: Feap
- Problembezeichnung: Testbeispiel

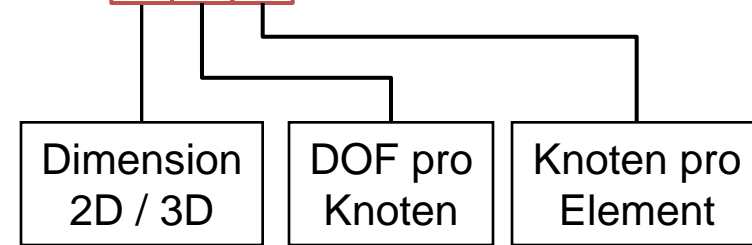
```
feap Testbeispiel
```

## Information zur globalen Problemgröße

- Die ersten drei Einträge werden von FEAP berechnet – Nullen eingetragen!
  1. Gesamtknotenanzahl
  2. Anzahl Elemente
  3. Anzahl Materialien
- Die letzten drei Einträge müssen angegeben werden

```
feap Testbeispiel
```

```
0 0 0 2 2 2
```



DOF: Degrees of Freedom /  
Freiheitsgrade

## Parameterdefinition (optional)

- Definition von Parametern ohne Variablendeklaration
- Rechenoperationen mit Parametern möglich

```
parameter
```

```
a=20
```

```
b=10
```

```
c=a+b*a*b
```

## Definition des Randwertproblems (RWP)

- RWP wird zwischen dem *Kopf* und *end* definiert
- Definition der
  - Geometrie / Knoten
  - Elemente
  - Randbedingungen
  - Belastungen
  - Elementtyp (Feldgleichungen)
  - Materialdefinition

## Solvereinstellungen

- Batch-Kommando nach *end* des RWP

## Steuerung des Postprozesses

- Batch-Kommando nach *end* des Solvers
- Ausgabe in ASCII-Datei sowie in EPS-Grafikdatei

```
coord> > > >
  1 1    0    0>
  2 0  100    0

elem> > >
  1 0 1 1 2> >

mate 1> >
  truss
    elastic isotropic E
    cross section A

eboun, set> > >
  1    0 1 1> > >
  1 100 0 1>

eforce> > >
  1 100 20 0> > >

end> > > >
```

```
batch      !solver> >
  tangent
  form
  solve
end
```

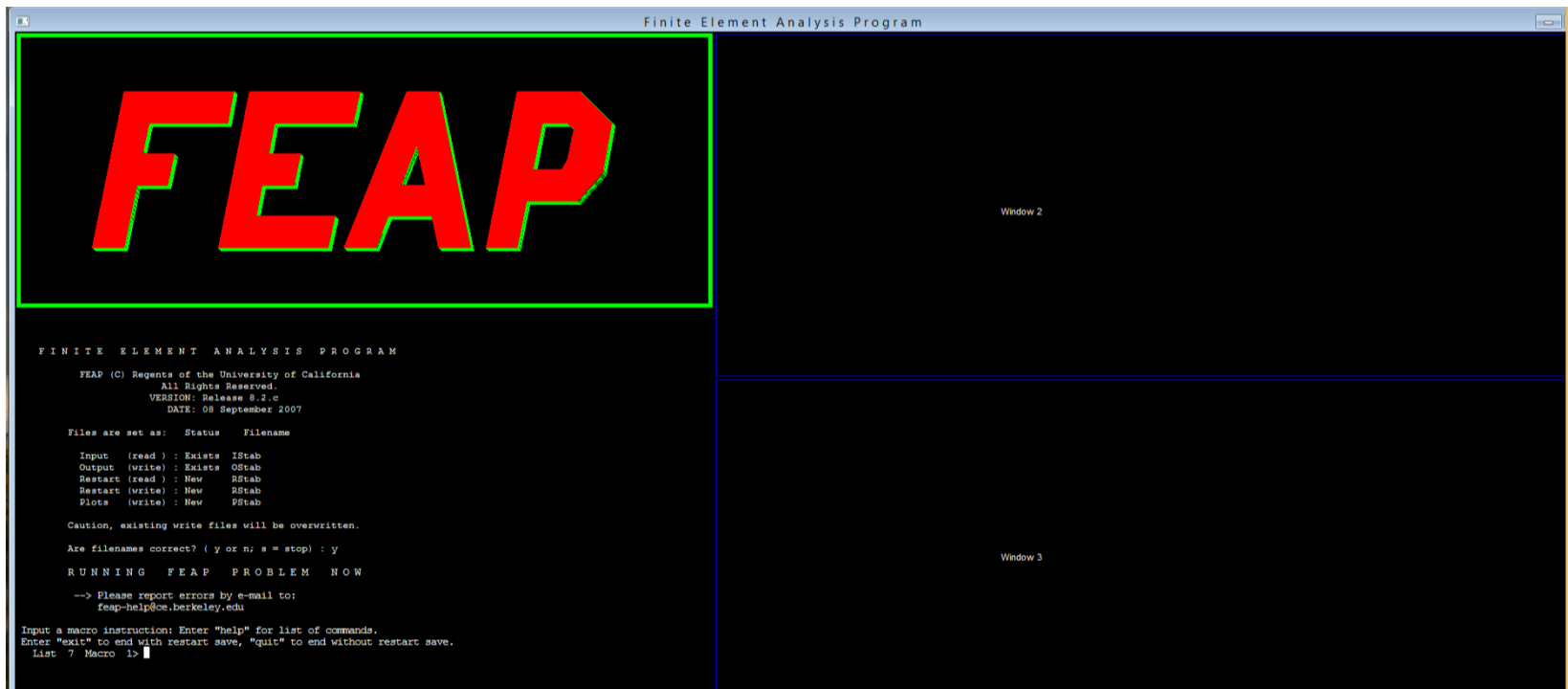
```
batch      !postprocess>
  disp,all
  stre,all
  plot cont 1
end
```

# Aufbau INPUT-File (5)

## Interactive

- Der Befehl `inter` aktiviert den interaktiven Modus zur Ergebnisauswertung
- grafische Benutzeroberfläche des Postprocessors erscheint nach Berechnungsende im FEAP-Fenster

`inter`

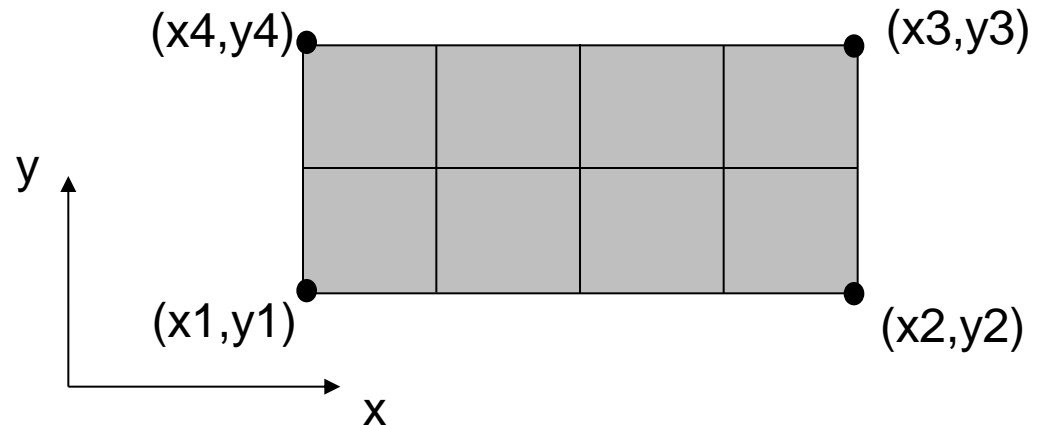


# Auswahl FEAP Befehle für INPUT-File (1)

- Erzeugung und Vernetzung von Flächen

BLOCK

```
cart, n1, n2  
1, x1, y1  
2, x2, y2  
3, x3, y3  
4, x4, y4
```



- Verschiebungsrandbedingungen am Knoten

BOUNDary

```
Knt-Nr, Inkrem, DOF-1, DOF-2
```

- Verschiebung vorgeben (vorher Verschiebungsrandbed. mit „boun“ definieren!)

DISPlacement

```
Knt-Nr, Inkrem, u1, u2
```

- Knotenkraft

FORCe

```
Knt-Nr, Inkrem, F1, F2
```

## Auswahl FEAP Befehle für INPUT-File (2)

- Material und Elementtyp
  - Dehnstab:

```
MATERial, Nr  
truss  
elastic isotropic E-Modul  
cross section A
```



- Solidelement (2D):

```
MATERial, Nr  
Solid  
elastic isotropic E-Modul nu  
thick section t
```



- 2-D EVZ oder ESZ einstellen

```
global  
plane strain ! oder stress
```

## Auswahl FEAP Befehle für INPUT-File (3)

- Randbedingung entlang einer Kante definieren

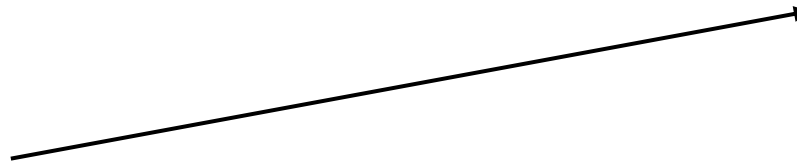
EBOUndary

Koord , Wert, DOF-1, DOF-2

z.B.

EBOUndary

1,0,1,1



- Kräfte für alle Knoten entlang einer Kante definieren

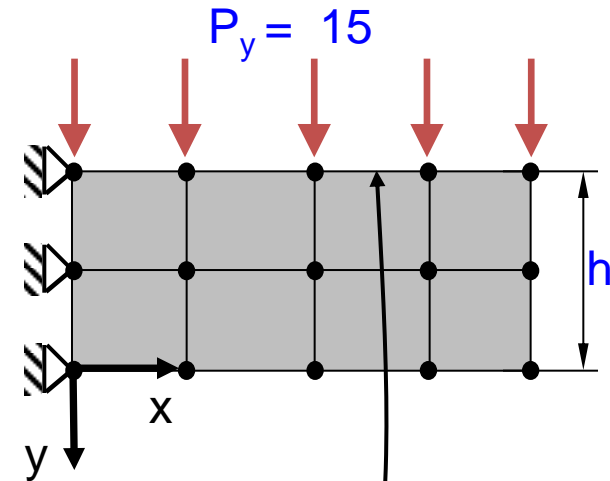
EFORce

Koord , Wert, F1, F2

z. B.

eforce

2,-h,0,15



- Knoten definieren

COORDinate

Knt-Nr., Inkrem., x-Koord., y-Koord.

- Element über die Knoten definieren

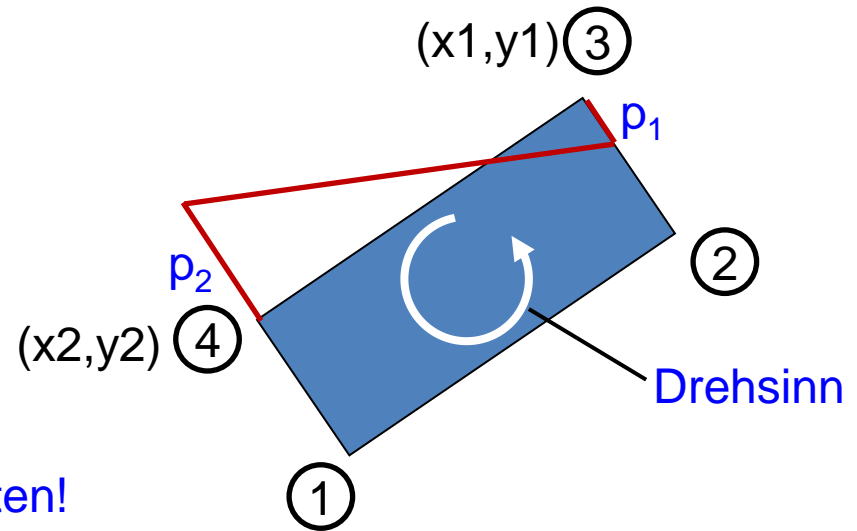
ELEMent

Elem-Nr, Inkrem., Mat.-Nr, 1.-Knt, i.-Knt

## Auswahl FEAP Befehle für INPUT-File (4)

- Streckenlasten

```
CSURface
normal, data
linear
1, x-Koord, y-Koord, p1
2, x-Koord, y-Koord, p2
```



Wichtig: Drehsinn der Elemente beachten!

- Lösung und Ausgabesteuerung in OUTPUT-File (OName)

```
BATCh
  TANGent      ! Struktursteifigkeitsmatrix aufstellen
  FORM         ! rechte Seite bilden
  SOLVe        ! Gleichungssystem lösen
  DISP,all     ! alle Knotenverschiebung in O-Datei ausgeben
  STRE,all     ! alle Spannungen in O-Datei ausgeben
  REAC,all     ! alle Reaktionskräfte in O-Datei ausgeben
END
```



## 1. Lineare Probleme [2, S. 182] (s. Vorlesung „Einführung in die FEM“)

Gls:  $\mathbf{K} \mathbf{u} = \mathbf{P}$   $\longrightarrow$  statisch, linear und **nicht** transient

- Befehlsstruktur

```
BATCh
```

```
  TANGent    ! Steifigkeitsmatrix bilden
```

```
  FORM       ! rechte Seite berechnen (Lastvektor)
```

```
  SOLVe      ! Gleichungssystem lösen
```

```
END
```

- Kurzschreibweise

```
BATCh
```

```
  TANGent,,1
```

```
END
```

### 2. Nichtlineare Probleme [2, S. 183 ff] (z. B. nichtlineares Materialmodell)

Verschiebungsinkrement  $\Delta \mathbf{u}$  wird iterativ mit NEWTON-Verfahren berechnet

➤ Lösung eines Nullstellenproblems in Form einer Residuengleichung

$$\mathbf{K} \Delta \mathbf{u}^i = \mathbf{R}(\mathbf{u}^i) \quad \text{mit} \quad \mathbf{K} = -\frac{\partial \mathbf{R}}{\partial \mathbf{u}} \quad \mathbf{u}^{i+1} = \mathbf{u}^i + \Delta \mathbf{u}^i$$

#### • Befehlsstruktur

BATCh

LOOP iter 10 ! 10 iterationen im NEWTON-Verfahren

TANGent ! Tangente berechnen

FORM ! Residuum berechnen

SOLVE ! Gl's lösen

NEXT iter

END

➤ Abbruchkriterium für das NEWTON-Verfahren

• Energienorm

$$E^i = \Delta \mathbf{u}^i \cdot \mathbf{R}^i$$

• Toleranz

$$E^i < tol E^0$$

• Befehl zur Einstellung der Toleranz

TOL (default 1.0 E-12)

Verschiedene Methoden zur zeitlichen Diskretisierung mittels **FEAP** Kommandostruktur einstellbar

➤ Reduzierter Befehlsumfang in **FEAPpv**

Definition der zeitlichen Diskretisierungsmethoden in `BATCH`-Umgebung

- implizite, zeitliche Diskretisierung
- explizite, zeitliche Diskretisierung

## 1. Quasistatische, transiente Probleme [2, s. 188 ff]

$$\mathbf{R}(t) = \mathbf{P}(t) - \mathbf{P}_{\text{int}}(\mathbf{u}(t)) = \mathbf{0} \quad \text{mit} \quad \mathbf{P}_{\text{int}} = \int_{\Omega} \mathbf{B}^T \boldsymbol{\sigma} dV$$

- Befehlsstruktur

BATCh

```
DT,,0.01          ! setzt Zeitschrittweite auf 0.01
LOOP time 20      ! 20 Zeitschritte durchführen
    TIME          ! aktuellen Zeitschritt um DT erhöhen
    LOOP iter 10 ! 10 Iterationen im NEWTON-Verfahren
        TANGent   ! Tangente berechnen
        FORM      ! Residuum berechnen
        SOLVe     ! Gls lösen
    NEXT iter
NEXT time
```

END

## 2. Nichtlineare, transiente Probleme [2, S. 190ff]

$$\mathbf{M} \ddot{\mathbf{u}}(t) + \mathbf{C} \dot{\mathbf{u}}(t) + \mathbf{P}_{\text{int}}(\mathbf{u}(t)) = \mathbf{P}(t)$$

z. B. implizite, zeitliche Diskretisierung und Verwendung des NEWTON-Verfahrens

- Befehlsstruktur

BATCh

```
DT,,0.01          ! setzt Zeitschrittweite auf 0.01
INITial (DISPlacements and/or RATEs)! Optional
TRANsient <xx>    ! tansiente Berechnung mit Methode <xx>
FORM ACCEleration !Optional: F(0),u(0) oder v(0)≠0
LOOP time 20      ! 20 Zeitschritte durchführen
    TIME          ! aktuellen Zeitschritt um DT erhöhen
    LOOP iter 10! 10 Iterationen im NEWTON-Verfahren
        TANGent    ! Tangente berechnen
        FORM       ! Residuum berechnen
        SOLVe      ! Gl's lösen
    NEXT iter
NEXT time
END
```

Beispiel NEWMARK-Verfahren: Für <xx> NEWMark eintragen

## Beispiel: Explizite, zeitliche Diskretisierung [2, S. 193]

- Befehlsstruktur

BATCh

DT,,0.01 ! setzt Zeitschrittweite auf 0.01

TRANsient EXPLicit! expliziter transiente Zeitdiskr.

INITial (DISPlacements and/or RATEs)! **Optional**

MASS LUMP ! diagonale Massenmatrix aufstellen

FORM ACCEleration! Anfangsbeschleunigung

LOOP time 20 ! 20 Zeitschritte durchführen

TIME EXPLicit! kritischen Zeitschritt kontrollieren

FORM EXPLicit

NEXT time

END

Beispiel: Verbesserte explizite, zeitliche Diskretisierung [2, S. 193 ff]

➤ Für den Fall, dass das Residuum anfänglich null ist.

- Befehlsstruktur

```
BATCh
```

```
DT,,0.01          ! setzt Zeitschrittweite auf 0.01
```

```
TRANsient EXPLicit! expliziter transiente Zeitdiskr.
```

```
INITial (DISPlacements and/or RATEs)! Optional
```

```
MASS LUMP          ! diagonale Massenmatrix aufstellen
```

```
FORM ACCEleration! Anfangsbeschleunigung
```

```
LOOP time 20       ! 20 Zeitschritte durchführen
```

```
    LOOP,,1        ! eine Iteration durchführen
```

```
        TIME EXPLicit! kritischen Zeitschritt kontrollieren
```

```
        FORM EXPLicit
```

```
    NEXT
```

```
NEXT time
```

```
END
```

## 3. Lineare, transiente Probleme

$$\mathbf{M} \ddot{\mathbf{u}}(t) + \mathbf{C} \dot{\mathbf{u}}(t) + \mathbf{K} \mathbf{u}(t) = \mathbf{P}(t)$$

Lösung mit Zeitschrittverfahren (NEWMARK) - weitere Möglichkeiten in [2, S. 196 ff]

- Befehlsstruktur

BATCh

```
DT,,0.01          ! setzt Zeitschrittweite auf 0.01
INITial (DISPlacements and/or RATEs)! Optional
TRANsient NEWMark! tansiente Berechnung mit Newmark
FORM ACCEleration !Optional: F(0),u(0) oder v(0)≠0
LOOP time 20      ! 20 Zeitschritte durchführen
    TIME          ! aktuellen Zeitschritt um DT erhöhen
    TANGent       ! Tangente berechnen
    FORM          ! Residuum berechnen
    SOLVe         ! GlS lösen
```

NEXT time

END

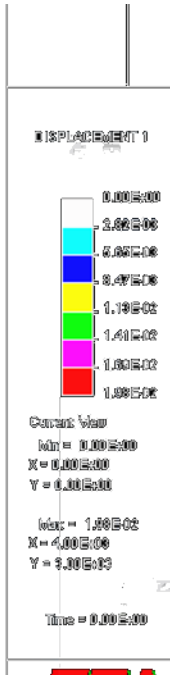
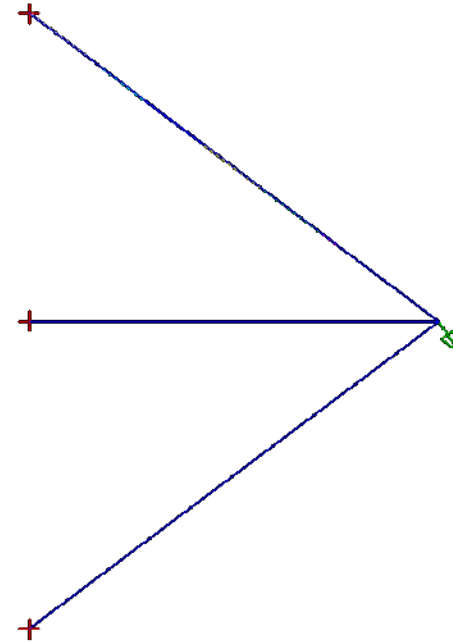


## Graphische Darstellung von Ergebnissen im Postprocessor

- Mit dem Befehl `plot` können
  - Knoten
  - Elemente
  - Randbedingungen etc.visualisiert werden
- Anwendung in grafischer Benutzeroberfläche (FEAP-Fenster)

Beispiel:

```
> plot
      > boun      : Randbedingungen
      > stre,1    : Spannung  $\sigma_{11}$ 
      > mesh      : Netz
```



- Alternativ: Batch-Befehle im INPUT-File

### Erstellen von Ausdrucken am Bildschirm und in EPS-Dateien über INPUT-File

batch

plot cont	→	contour: farbige Verläufe von Verschiebungen, Spannungen
plot disp	→	Knotenverschiebungen
plot stre	→	Spannungen
plot load	→	Belastung
plot boun	→	Verschiebungsrandbedingungen
plot mesh	→	FE-Netz
plot reac	→	Lagerreaktionskräfte
plot node	→	Knotennummerierung
plot elem	→	Elementnummerierung
plot axis	→	Koordinatenachsen
plot defo	→	Verformte Darstellung
plot clear	→	Bereinigen des Ausgabefensters

end

batch

plot postscript	→	öffnet *.eps Datei
plot mesh	→	erzeugt grafische Ausgabe in *.eps
plot postscript	→	schließt *.eps

end

[1] FEAP Anleitung des Fachgebiets in Deutsch

Internetseite Fachgebiet:

Lehre → Computational Mechanics → Vorlesungsmaterial

[2] R. L. TAYLOR: FEAP – A Finite Element Analysis Program, User Manual, vers. 8.4, Dez., 2013

<http://www.ce.berkeley.edu/projects/feap/manual84.pdf>

[3] R. L. TAYLOR: FEAP – A Finite Element Analysis Program, Theory Manual, vers. 8.4, May, 2013

<http://www.ce.berkeley.edu/projects/feap/theory84.pdf>

[4] R. L. TAYLOR: FEAP – A Finite Element Analysis Program, Example Manual, vers. 8.4, May, 2013

<http://www.ce.berkeley.edu/projects/feap/example84.pdf>