EXAMPLES FOR ELASTO-PLASTIC MATERIAL BEHAVIOR

ECI280

Professor: Boris Jeremić

PhD student: Yuan Feng

02/09/2017

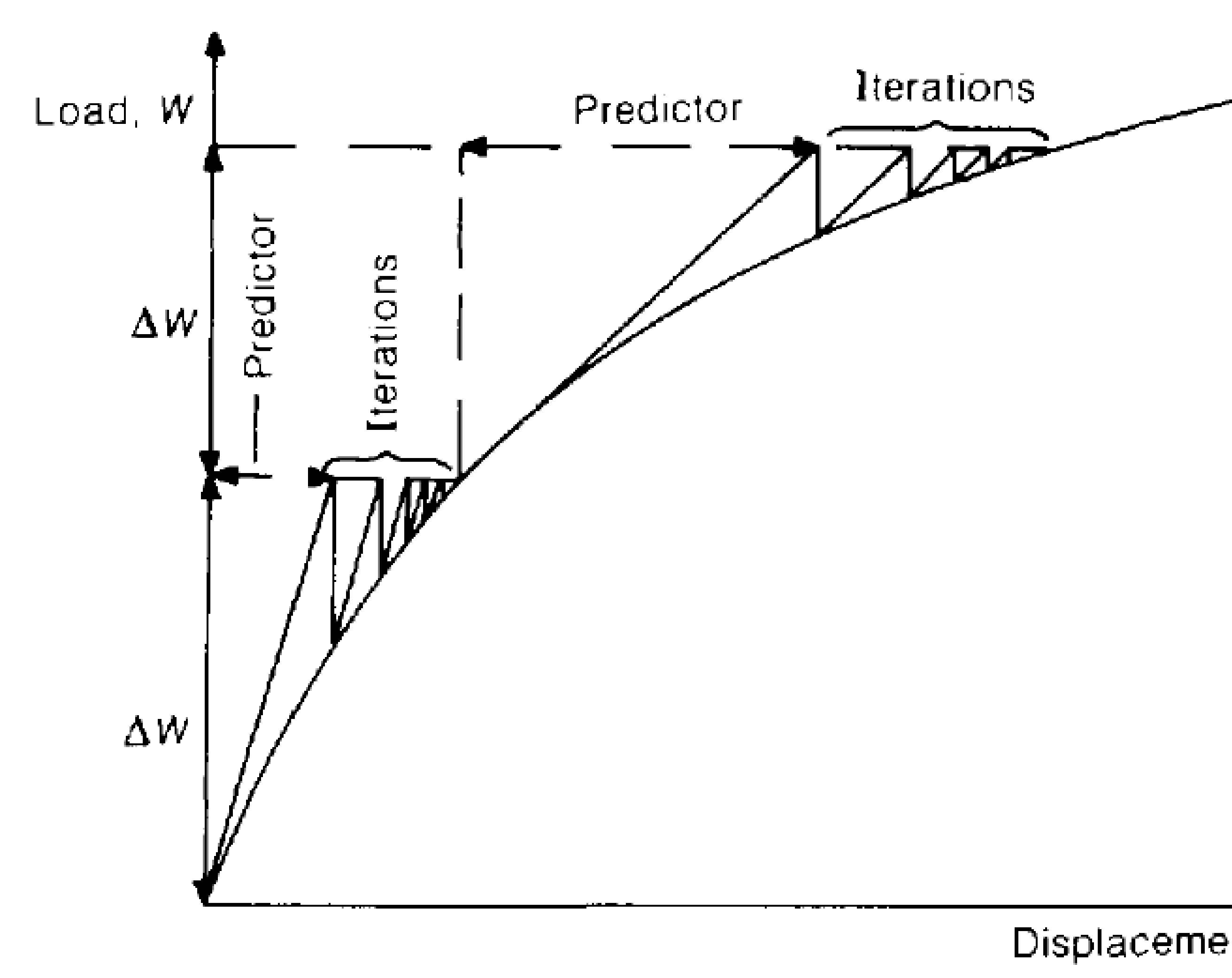
OVERVIEW

http://cml01.engr.ucdavis.edu/yuan/education_examples/

Nonlinear FEM Procedures:

First Level Iteration:

1. *External* Incremental Force.
2. Incremental Displacement from Tangent Stiffness and incr Force.
3. Incremental Strain from Incremental Displacement and strain-disp relation.
4. **Incremental Stress from Incremental strain**
5. *Internal* Incremental Force from Incremental Stress at Gauss Sampling.
6. Use Step 1 and Step 5 to obtain the unbalanced Force.
7. Input unbalance to Step 1 Again.
8. Until unbalanced Force is within the Tolerance.

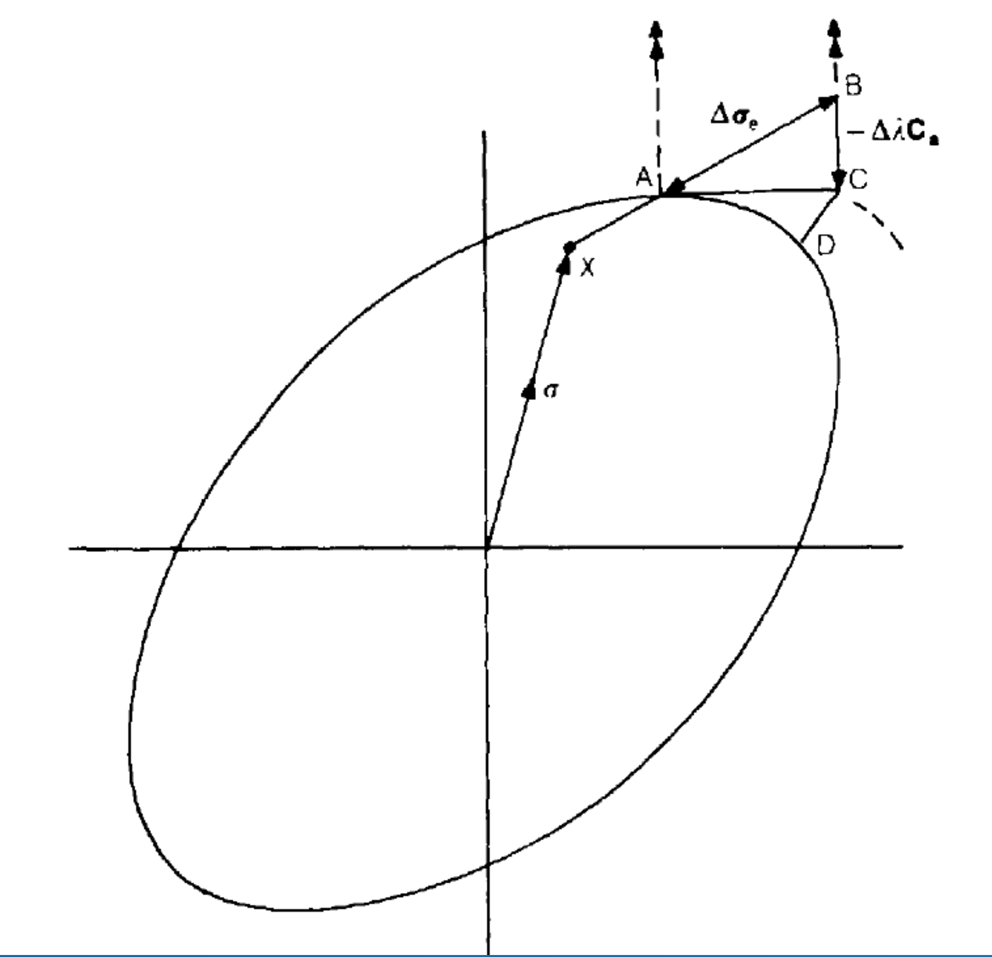


OVERVIEW

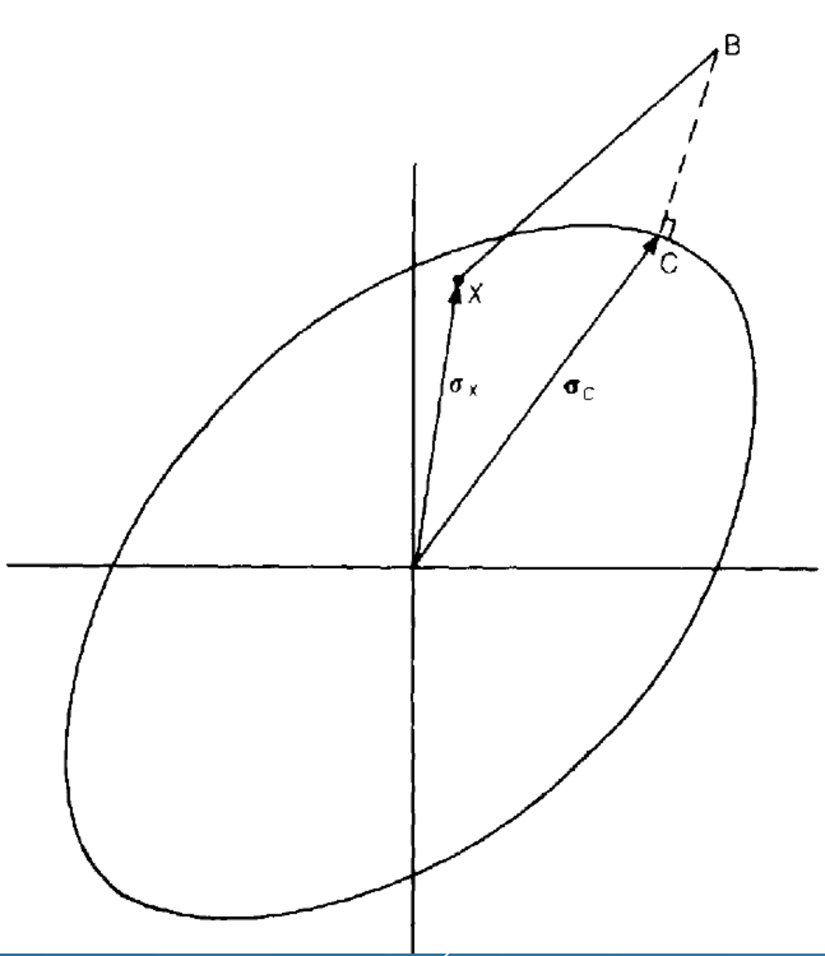
Overview

Nonlinear FEM Procedures:
Second Level Iteration:

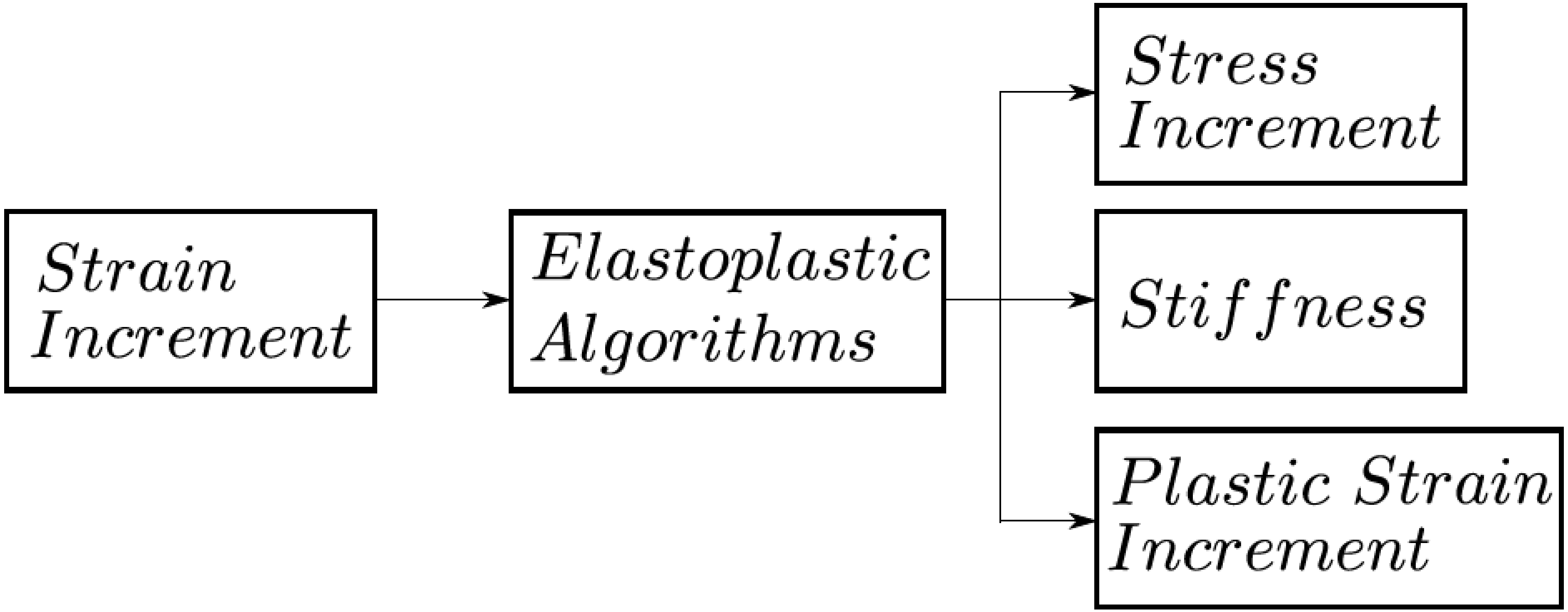
1. Elastic Predictor
2. Plastic Corrector



Forward Euler Algorithm



Backward Euler Algorithm


Strain Increment
Elastoplastic Algorithms
Stress Increment
Stiffness
Plastic Strain Increment

- Elastic Predictor
    - Elastic Stiffness
- Plastic Corrector
    - Yield Surface
    - Plastic Flow Rule
    - Hardening Law

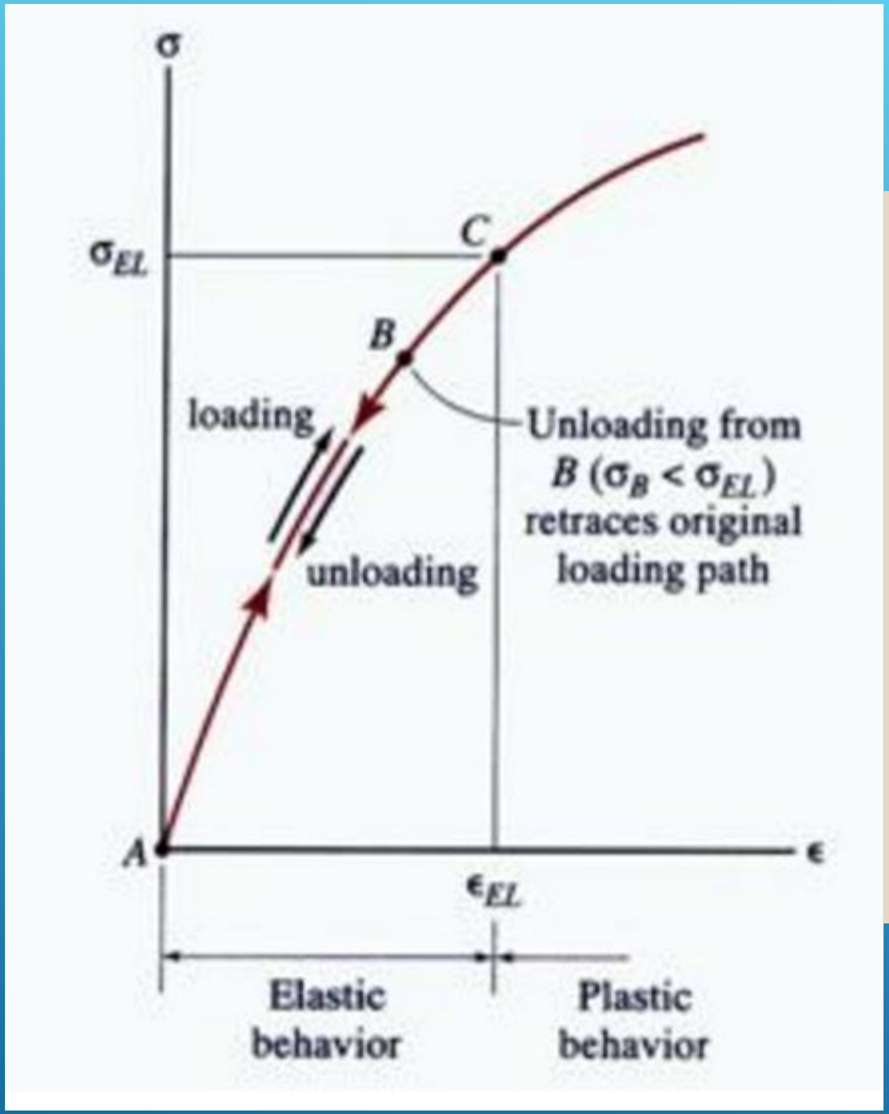
ELASTO-PLASTIC MATERIALS

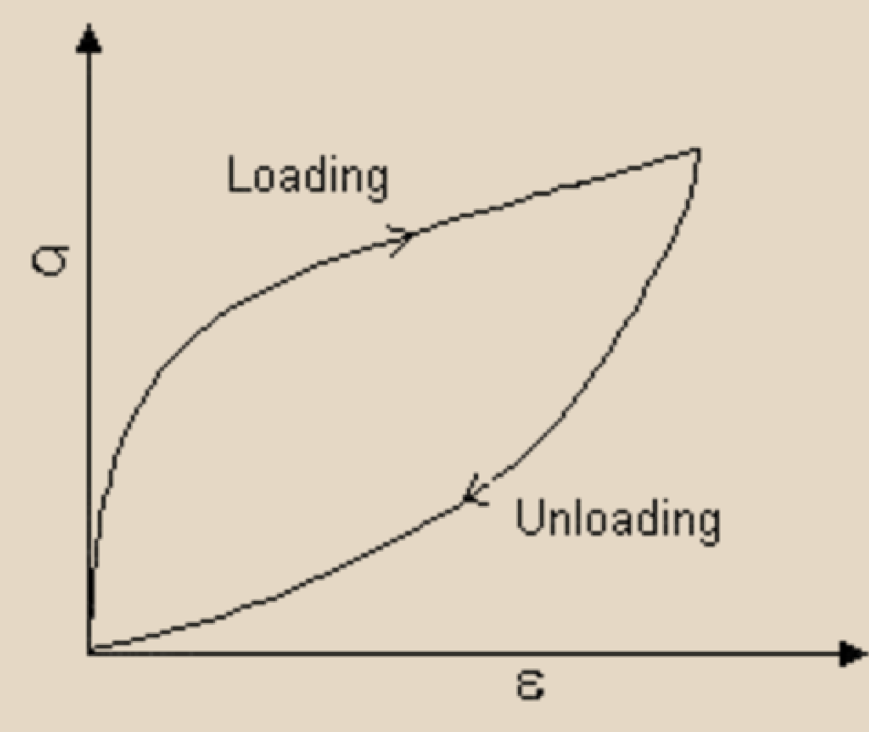
ELASTO-PLASTIC MATERIALS

- **Elasticity**
- Yield Surface
- Plastic Flow Rule
- Hardening Law

- Linear Elasticity
- Nonlinear Elasticity

- *Show Examples*





ELASTICITY

User-Input: Before the Translation

```
add material # 1 type linear_elastic_isotropic_3d
    mass_density = 2E3 * kg/m^3
    elastic_modulus = 2E7 * Pa
    poisson_ratio= 0.25 ;
```

DSL-Output: After the Translation

```
int add_constitutive_model_NDMaterial_linear_elastic_isotropic_3d(int MaterialNumber,
        double ElasticModulus,
        double nu,
        double rho)
{
    NDMaterial* theMaterial = 0;
    theMaterial = new ElasticIsotropic3D(MaterialNumber, ElasticModulus, nu, rho);
```

DOMAIN SPECIFIC LANGUAGE

EXECUTE ESSI BY COMMAND

```
essi -f input_filename.fei
```

ESSI: Earthquake Soil-Structure Interaction

FEI: Finite Element Interface

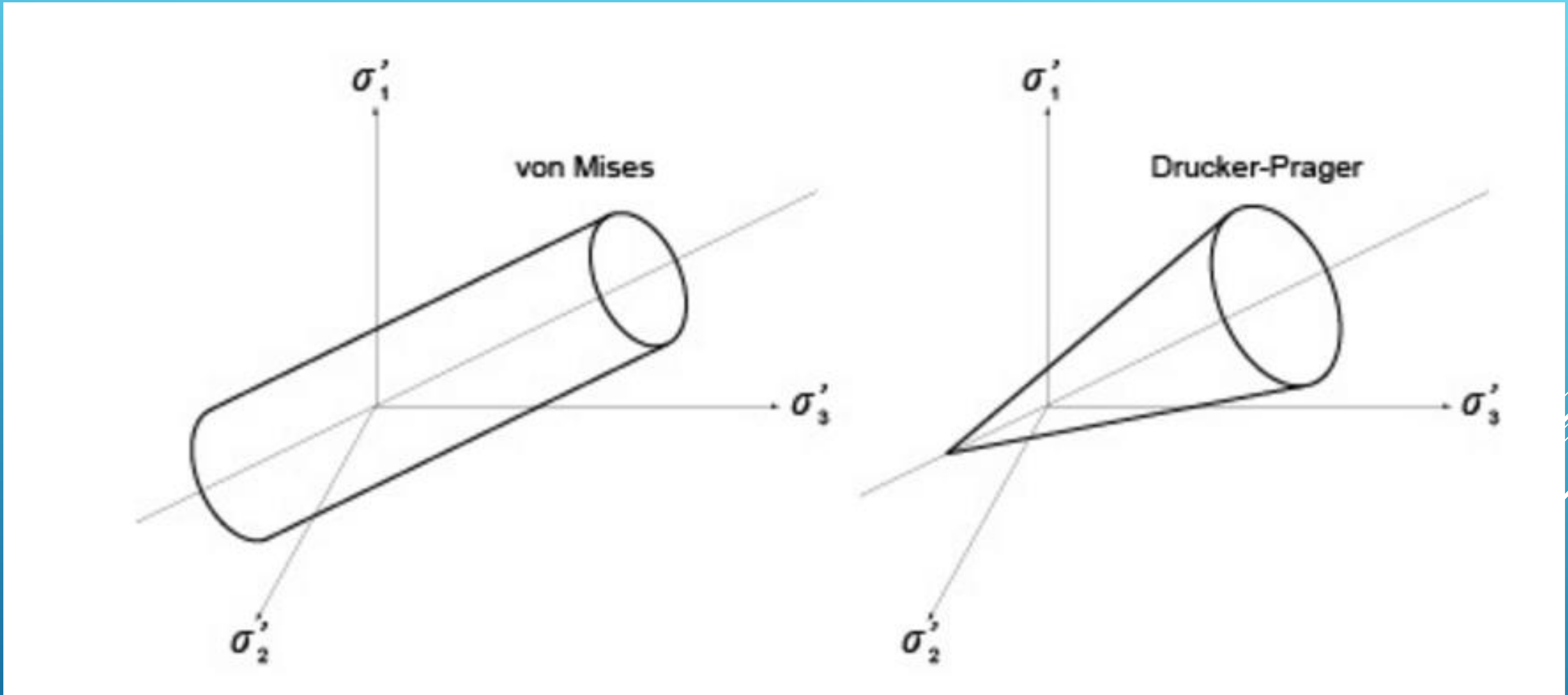
- Elasticity
- **Yield Surface**
- Plastic Flow Rule
- Hardening Law

ELASTO-PLASTIC MATERIALS

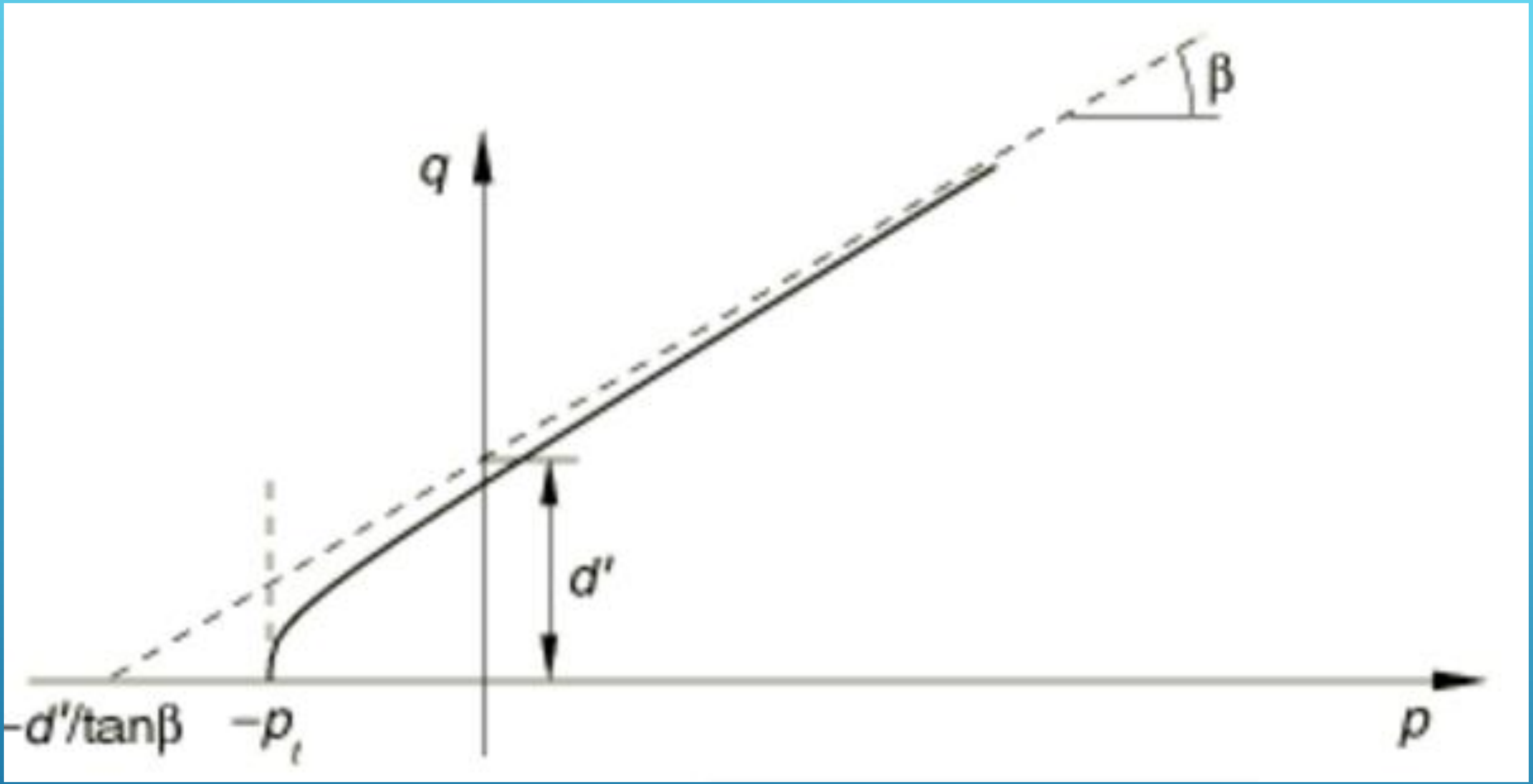
- Von-Mises Yield Surface
- Drucker-Prager Yield Surface
- Hyperbolic Drucker-Prager Yield Surface

- *Show Examples*

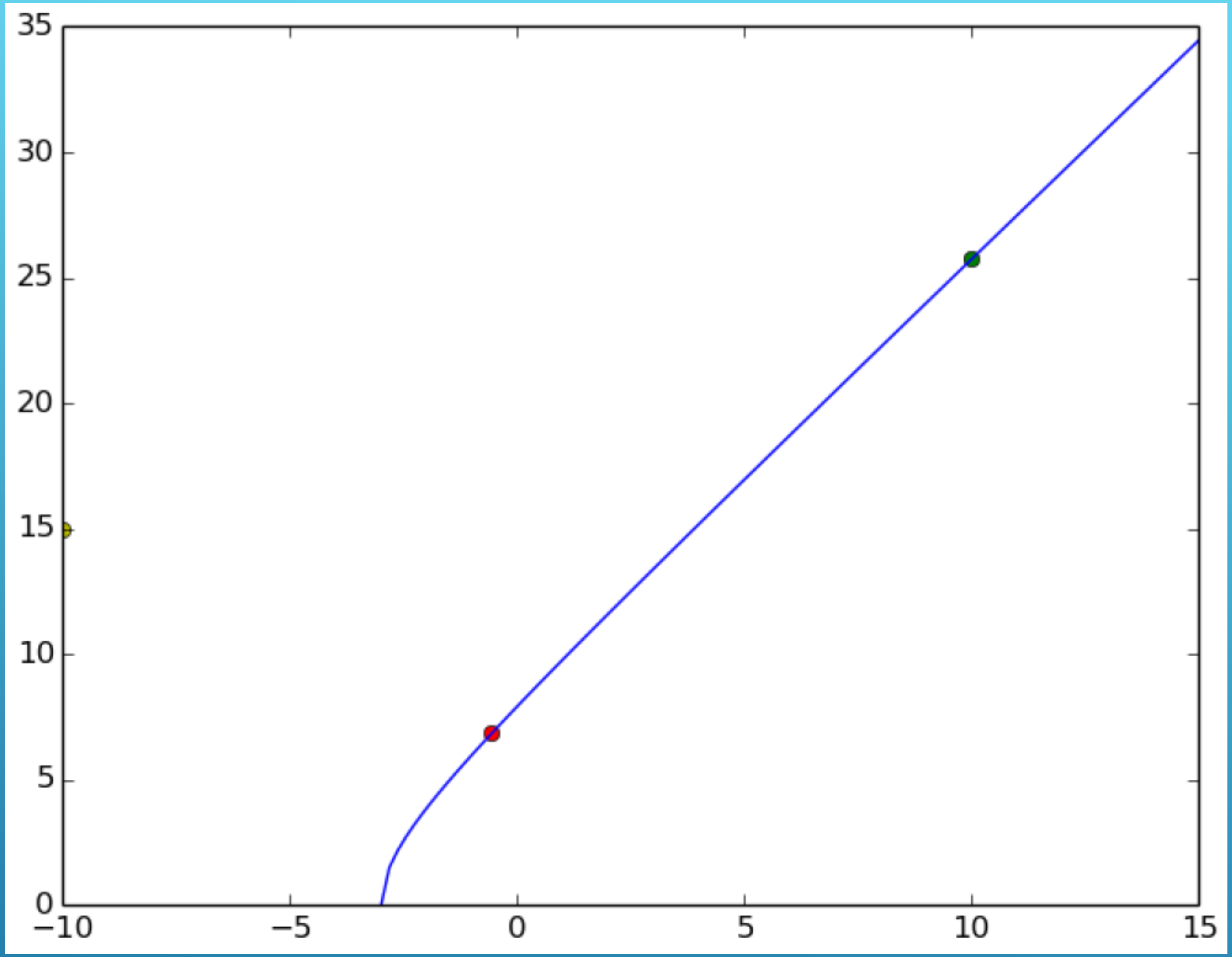
YIELD SURFACE



von Mises
Drucker-Prager
σ'1
σ'2
σ'3
σ'1
σ'2
σ'3



HYPERBOLIC DRUCKER-PRAGER



HYPERBOLIC DRUCKER-PRAGER

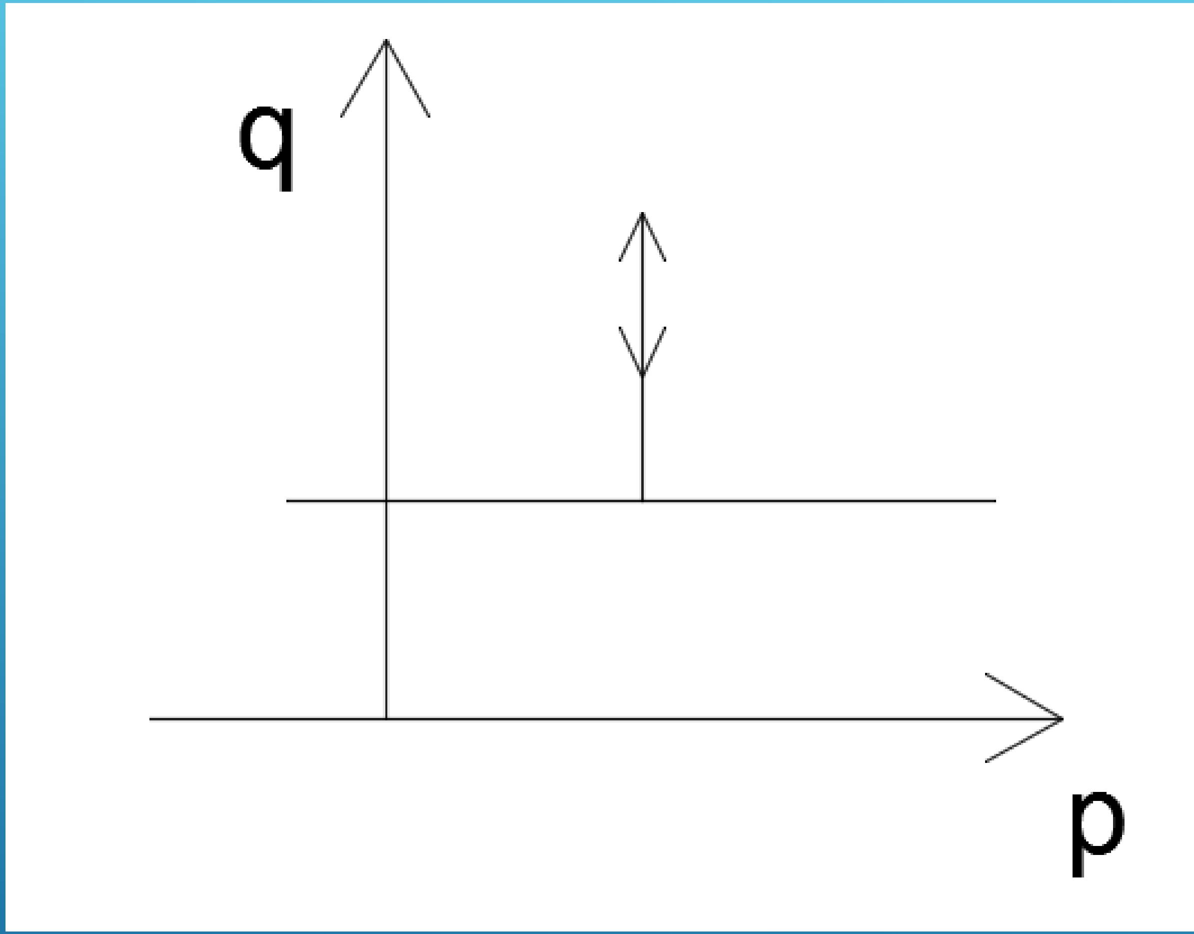
- Elasticity
- Yield Surface
- **Plastic Flow Rule**
- Hardening Law

ELASTO-PLASTIC MATERIALS

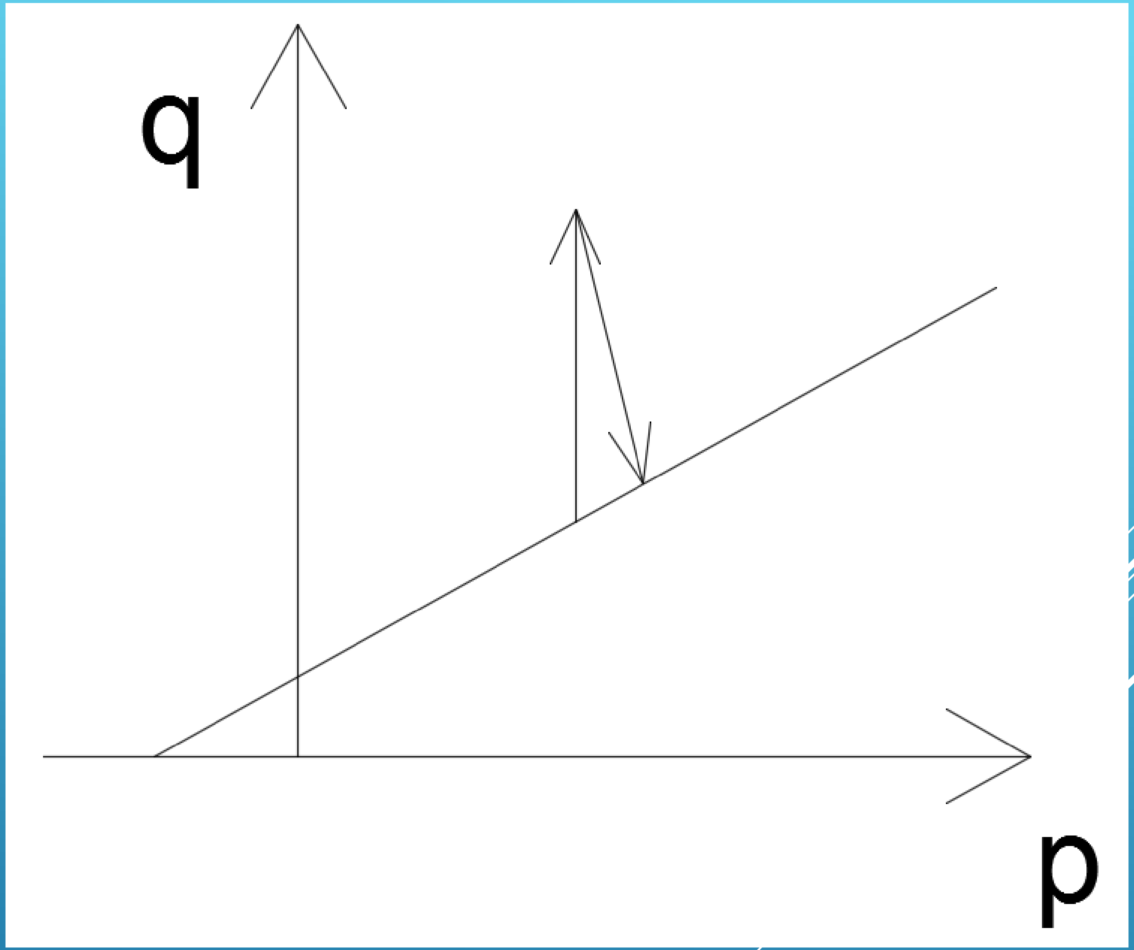
- Associative Plastic Flow Rule
- Non-associative Plastic Flow Rule

- *Show Examples*

PLASTIC FLOW RULE

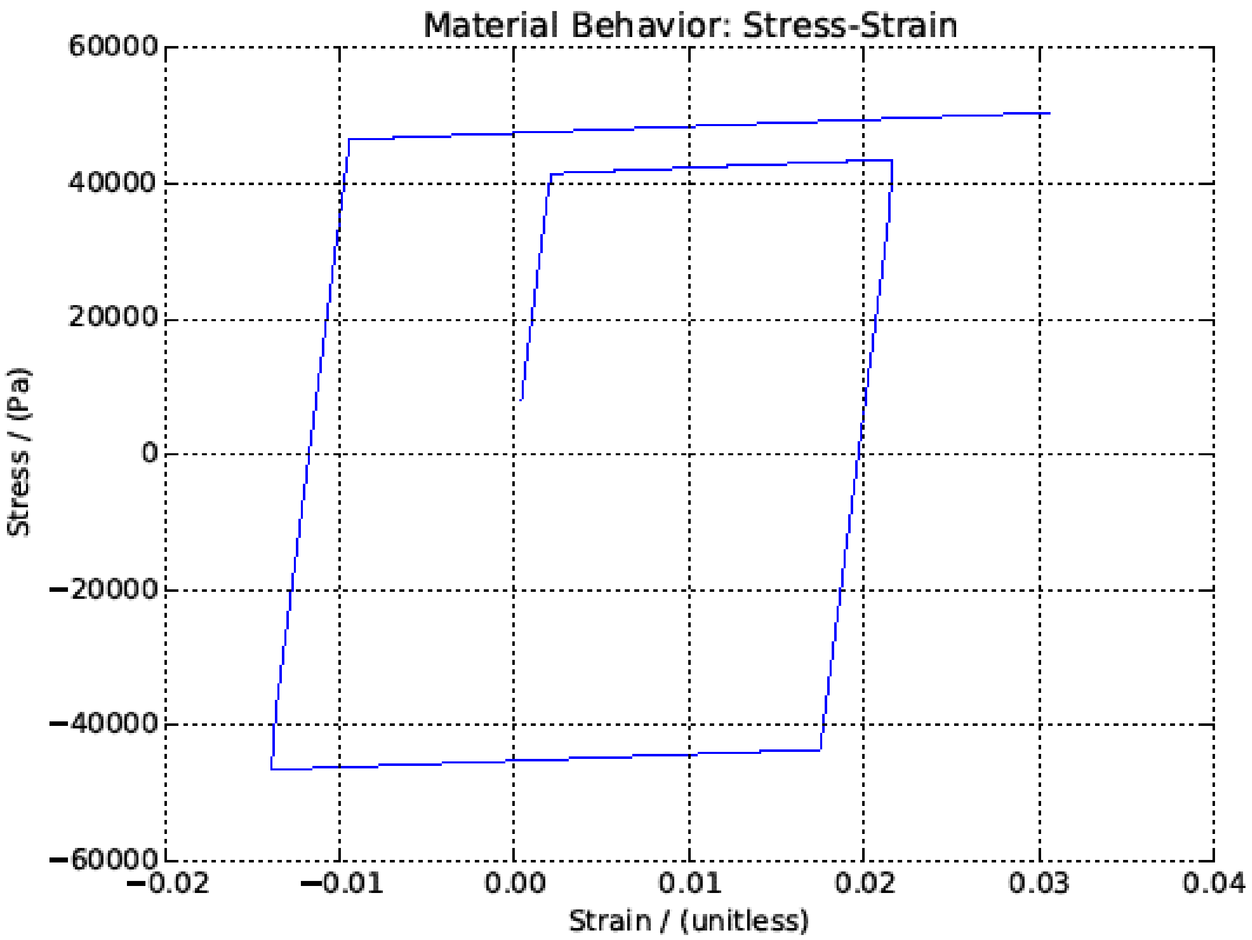


Von-Mises

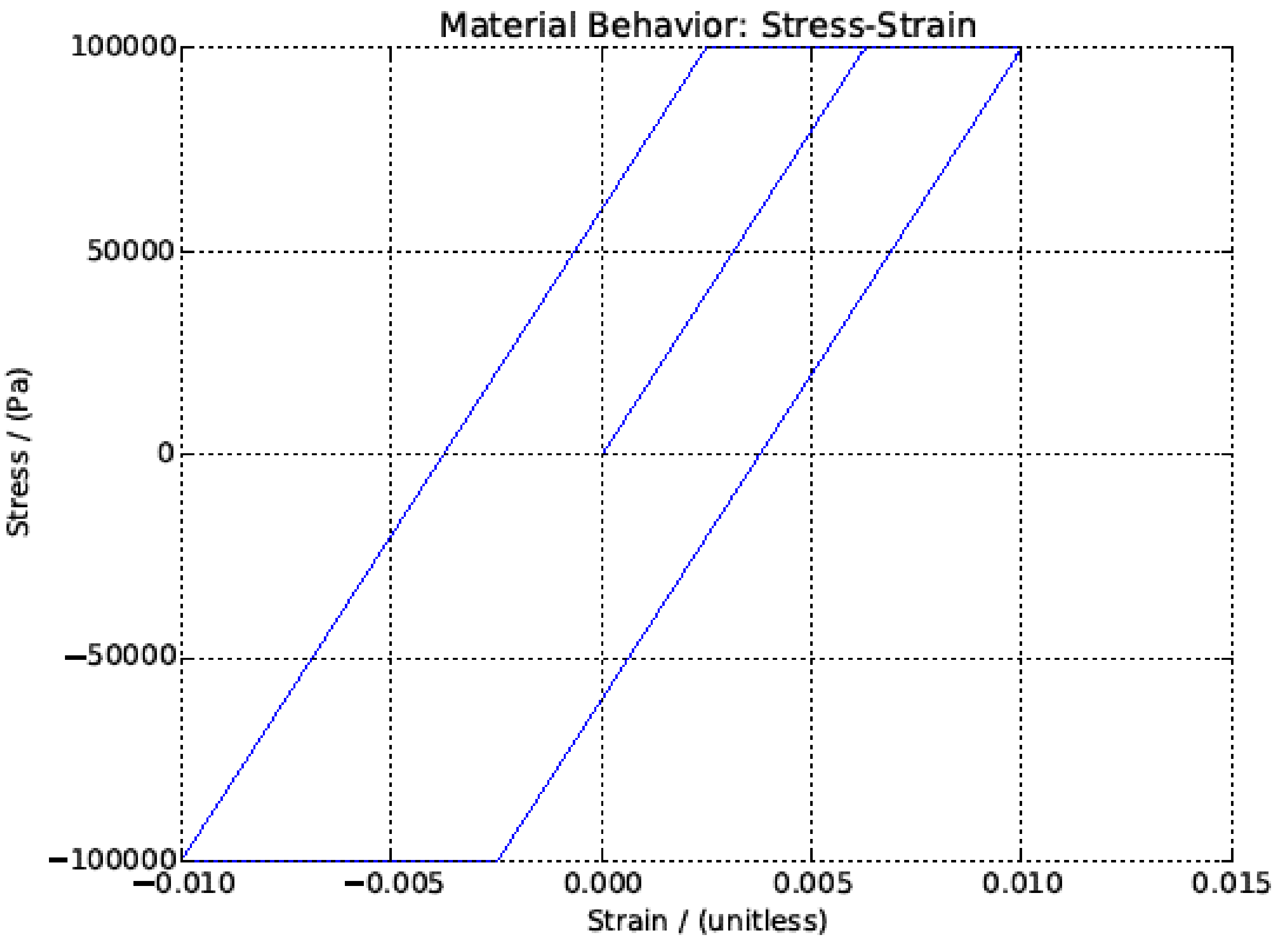


Drucker-Prager

- $p = -1/3 * (sigma_{11} + sigma_{22} + sigma_{33})$
- $q = sqrt(3/2*s_{ij} * s_{ij})$

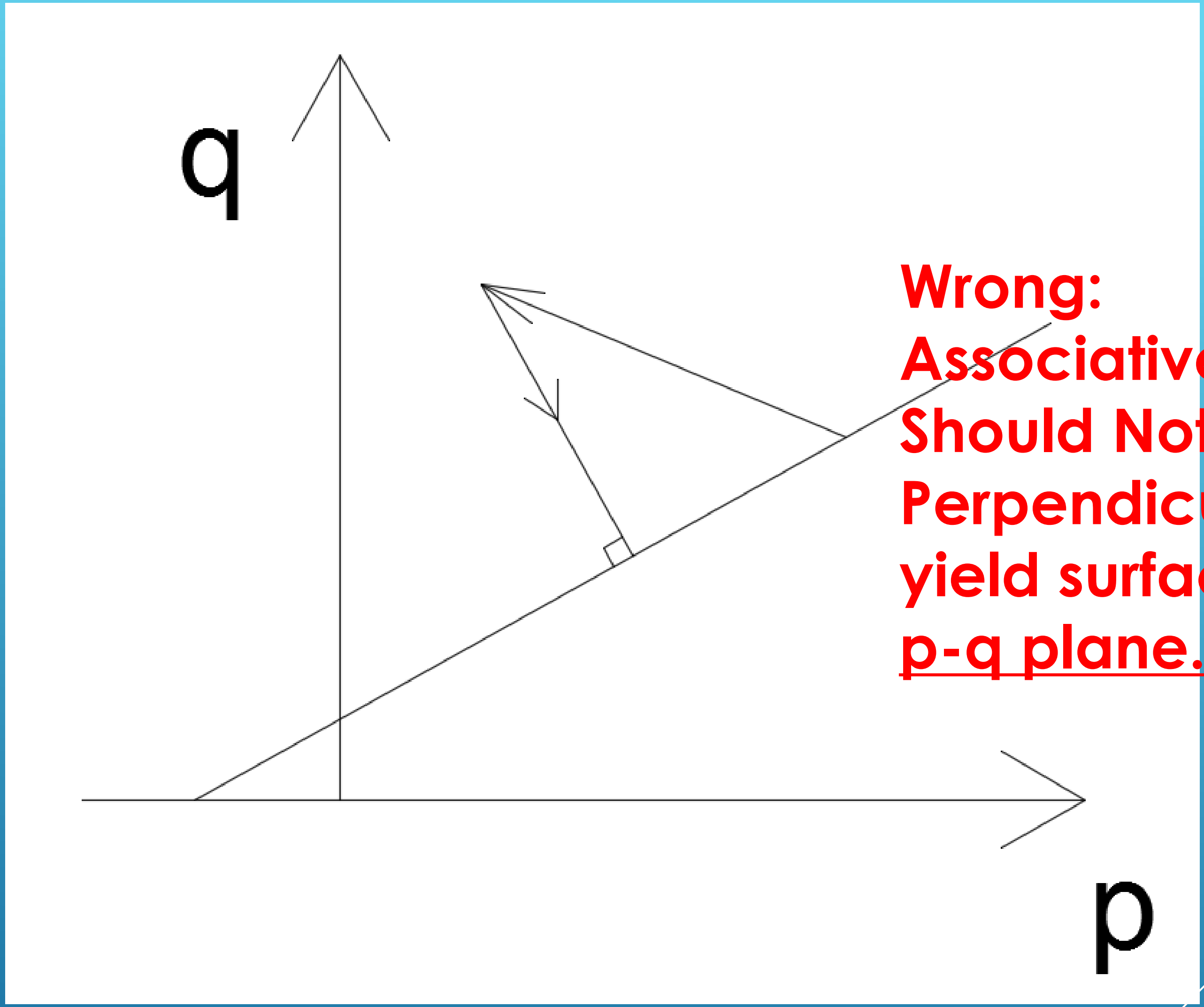


Material Behavior: Stress-Strain
Stress / (Pa)
Strain / (unitless)
60000
40000
20000
0
−20000
−40000
−60000
−0.02
−0.01
0.00
0.01
0.02
0.03
0.04



Material Behavior: Stress-Strain
Stress / (Pa)
Strain / (unitless)
100000
50000
0
−50000
−100000
−0.010
−0.005
0.000
0.005
0.010
0.015

In Drucker-Prager associative plastic flow rule, should the plastic corrector be perpendicular to the yield surface in the p-q plane?



Wrong:
Associative Flow Rule Should Not Return Perpendicularly to the yield surface <u>in the p-q plane.</u>

DRUCKER-PRAGER YIELD SURFACE

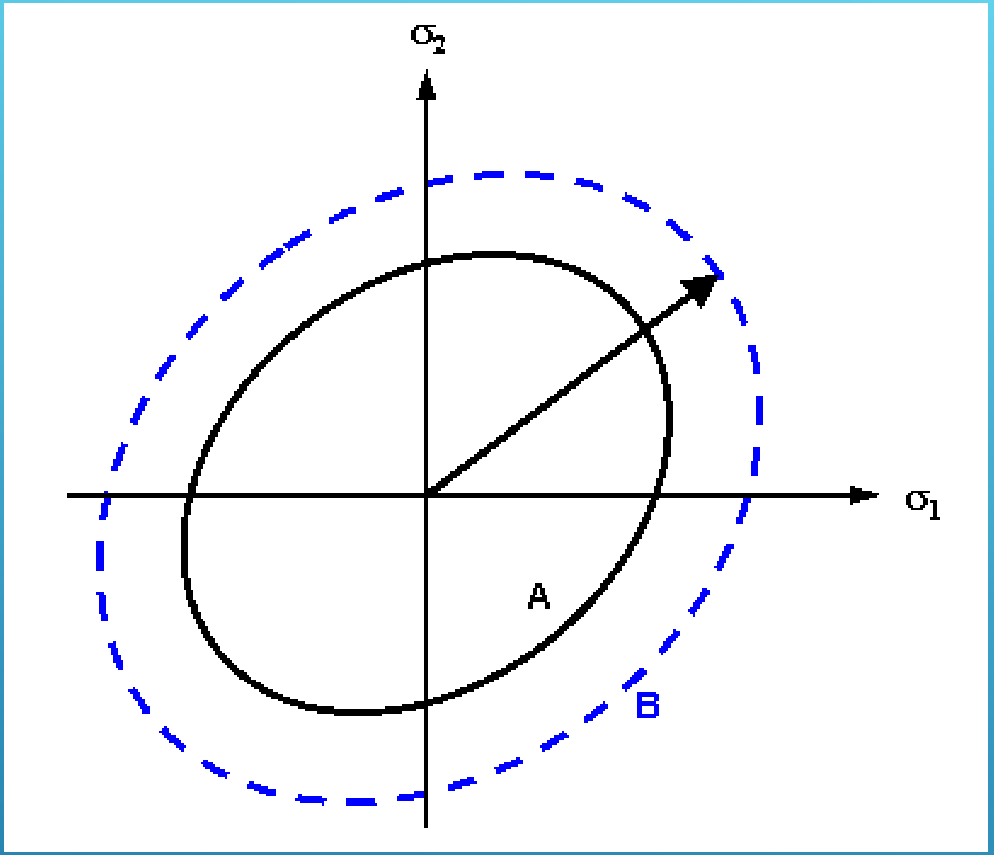
Reference:
Analytical CPP in energy-mapped stress space: application to a modified Drucker–Prager yield surface

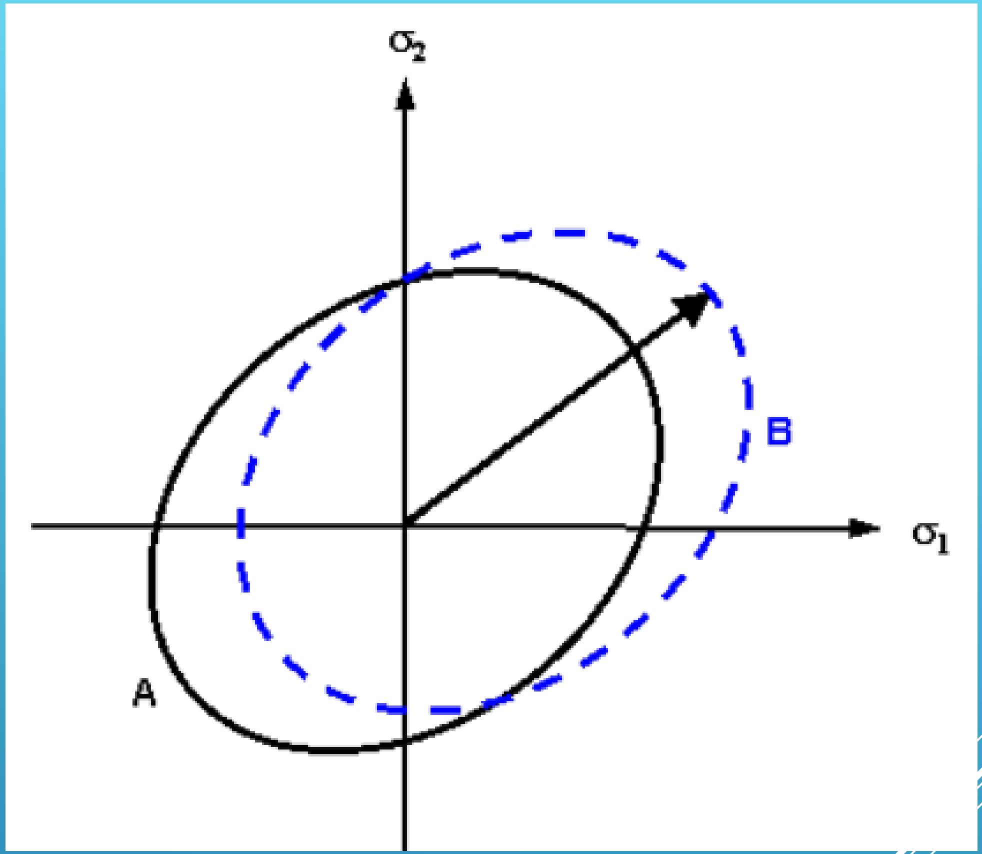
- Elasticity
- Yield Surface
- Plastic Flow Rule
- **Hardening Law**

ELASTO-PLASTIC MATERIALS

- Isotropic Hardening Rule
- Kinematic Hardening Rule
- Armstrong-Frederick Hardening Rule
- Multi-Surface Hardening Rule

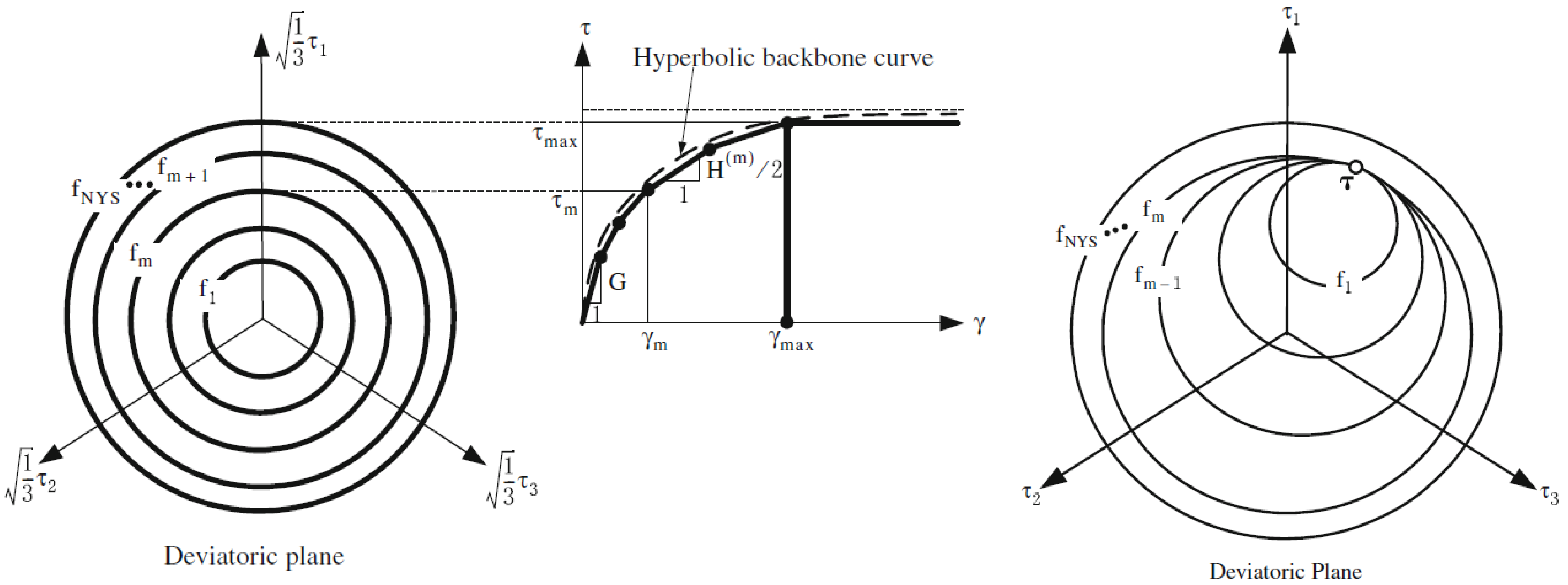
HARDENING LAW



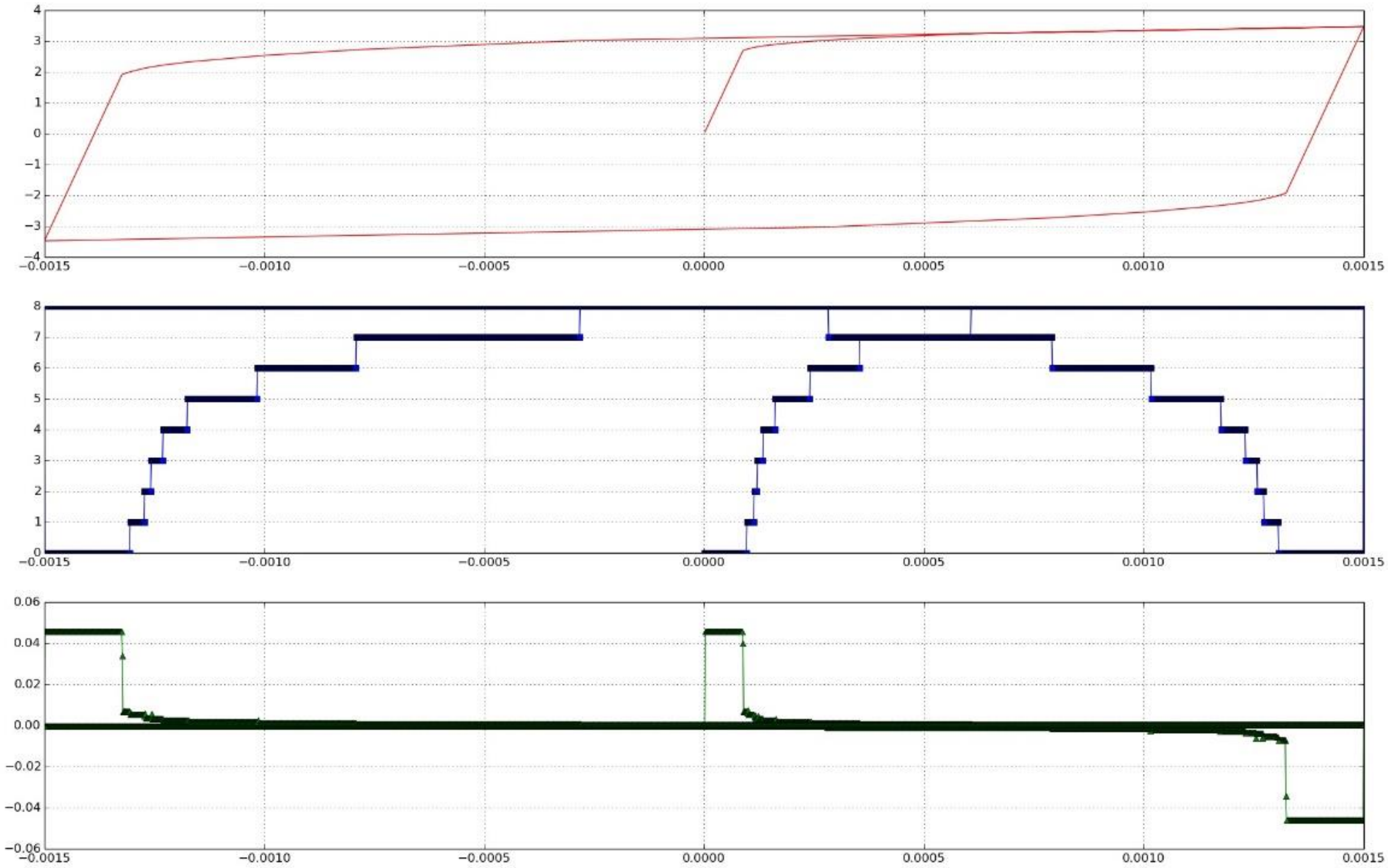


Show Examples

ISOTROPIC & KINEMATIC HARDENING



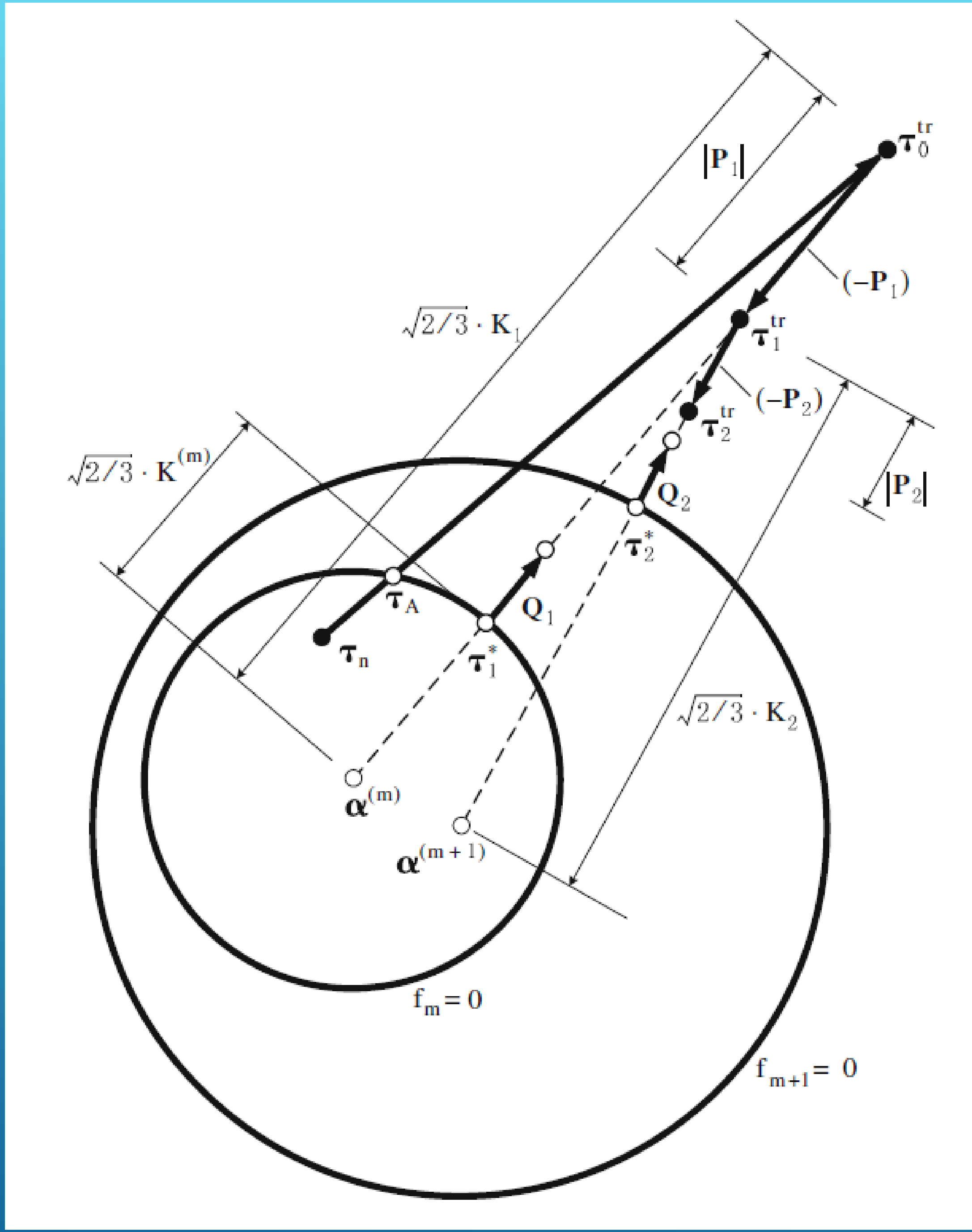
MULTI-YIELD-SURFACE HARDENING LAW



MULTI-YIELD-SURFACE HARDENING LAW

MULTI-YIELD-SURFACE ALGORITHMS

- Reference
- Prevost. JH. A simple plasticity theory for frictional cohesionless soils. 1985.

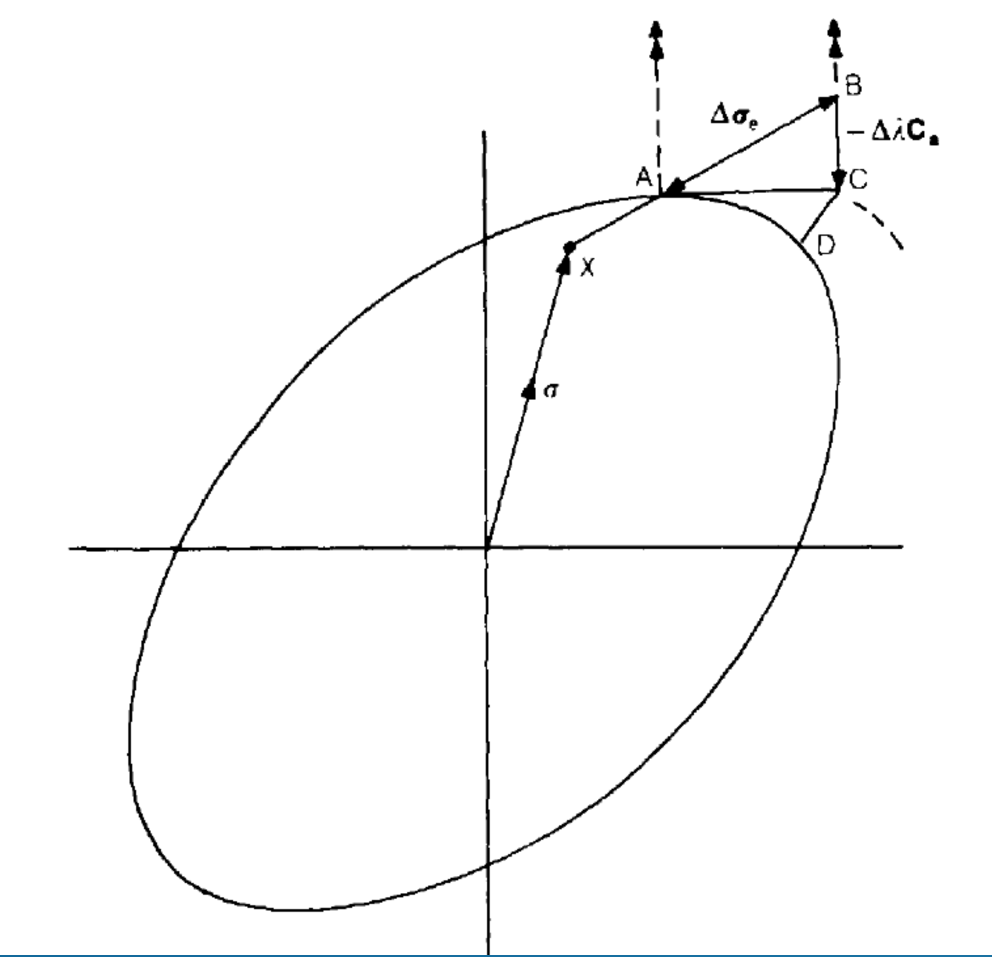


APPLICATION IN FINITE ELEMENT SOLID BRICK

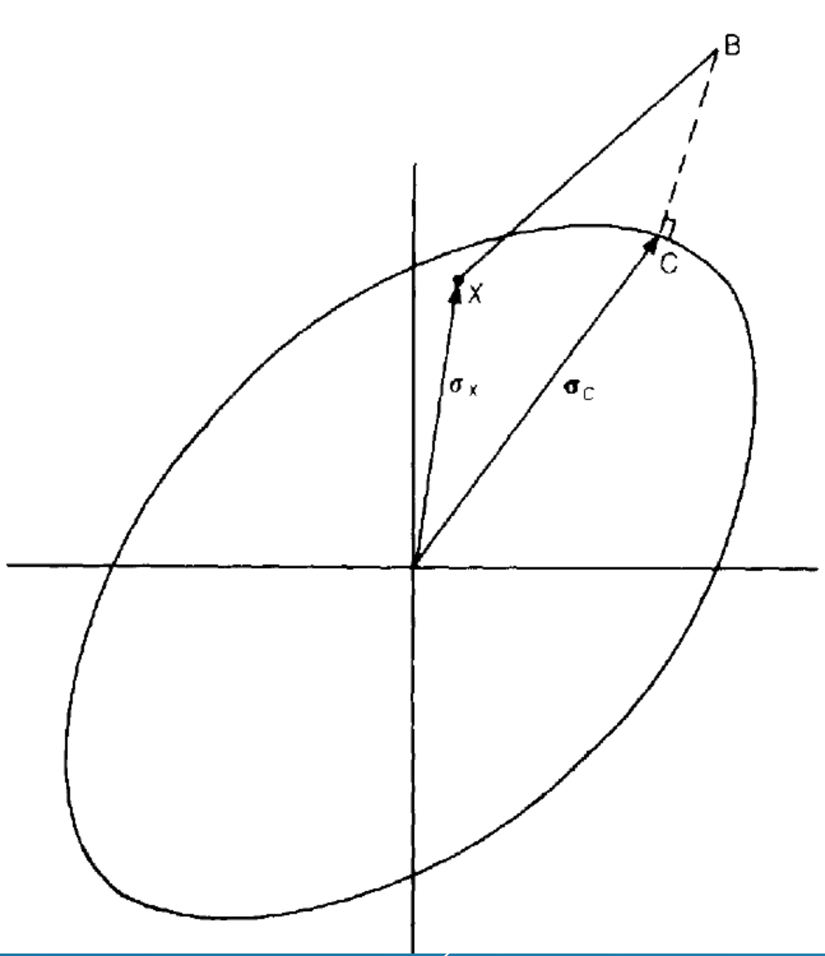
OVERVIEW

Nonlinear FEM Procedures:
Second Level Iteration:

1. Elastic Predictor
2. Plastic Corrector



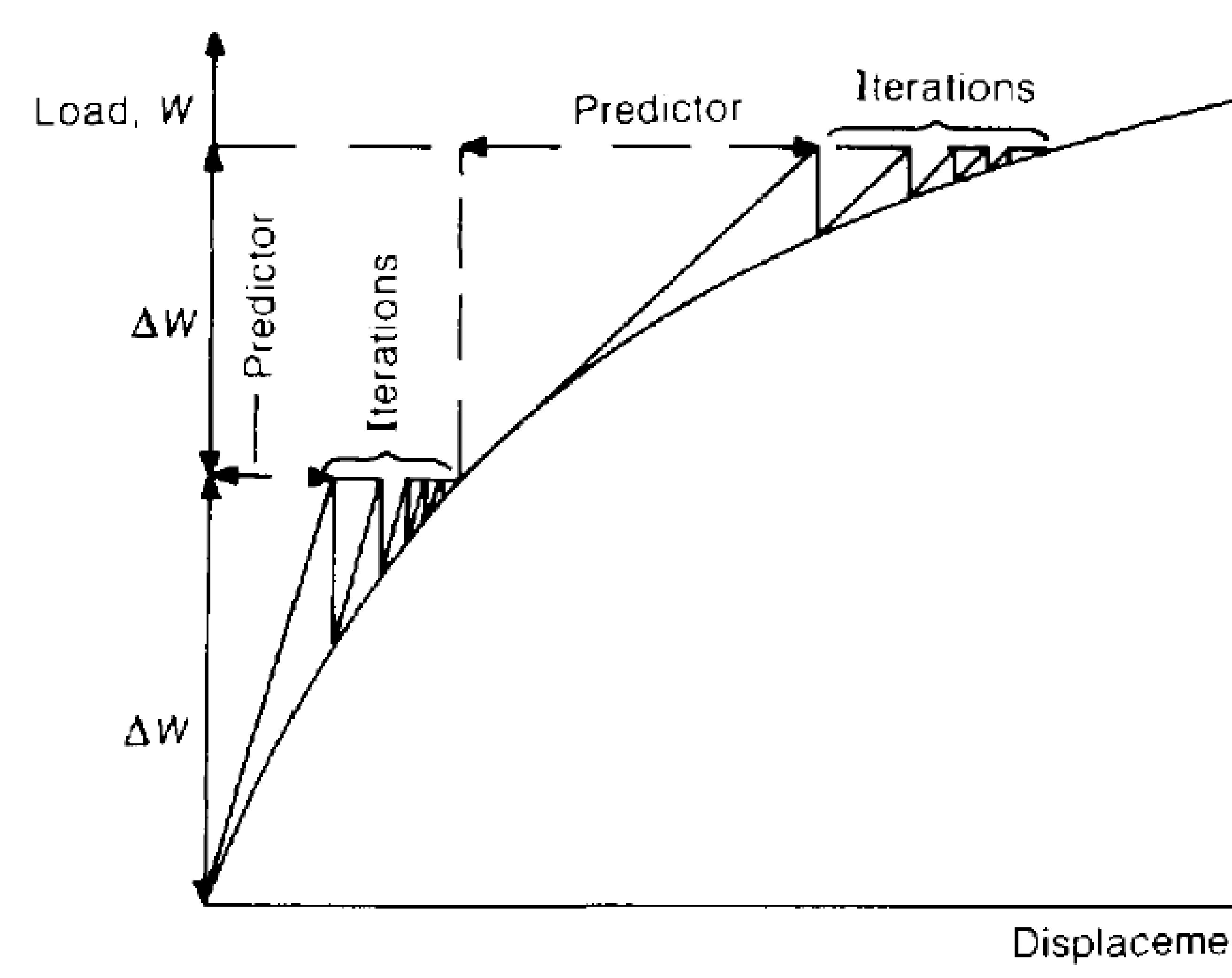
Forward Euler Algorithm



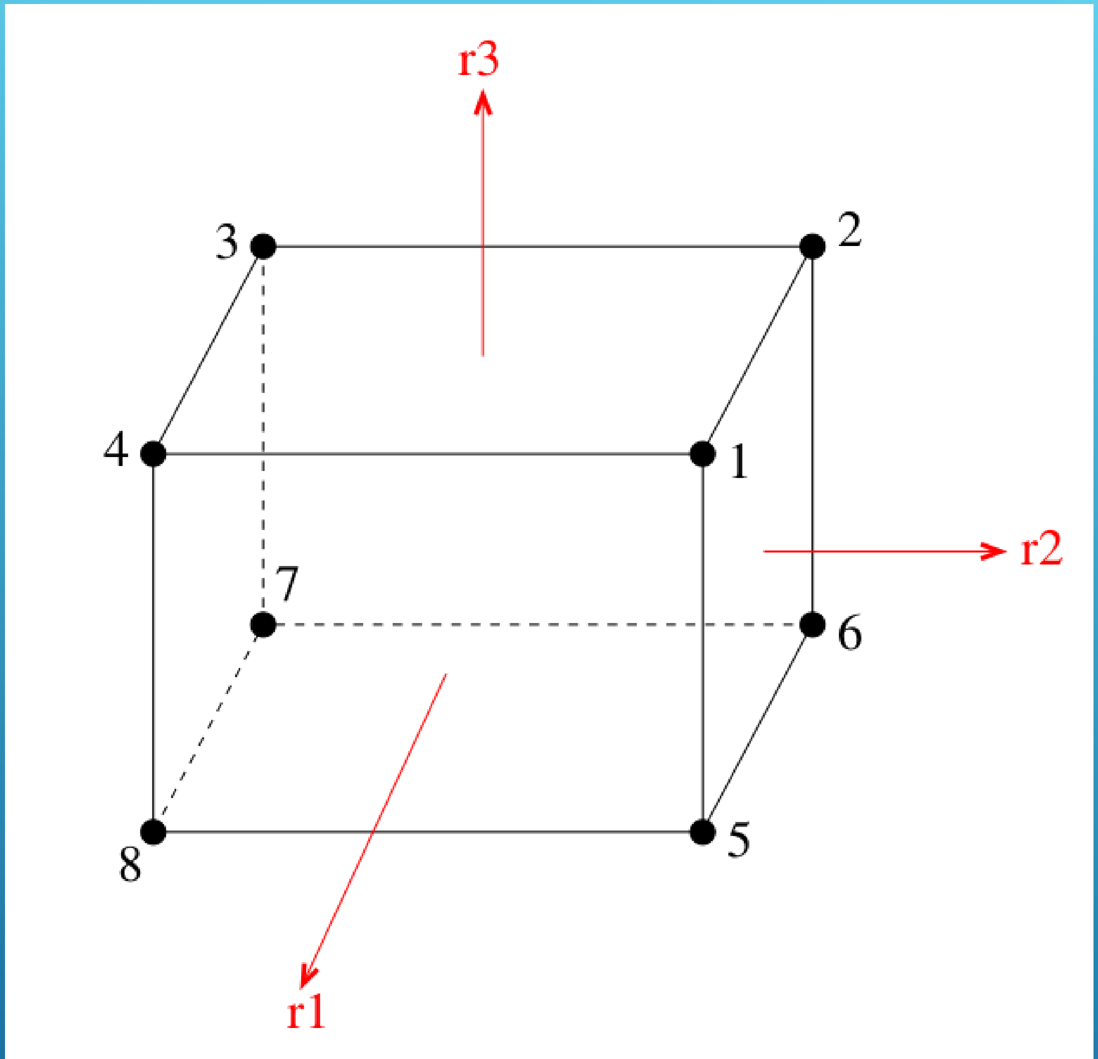
Backward Euler Algorithm

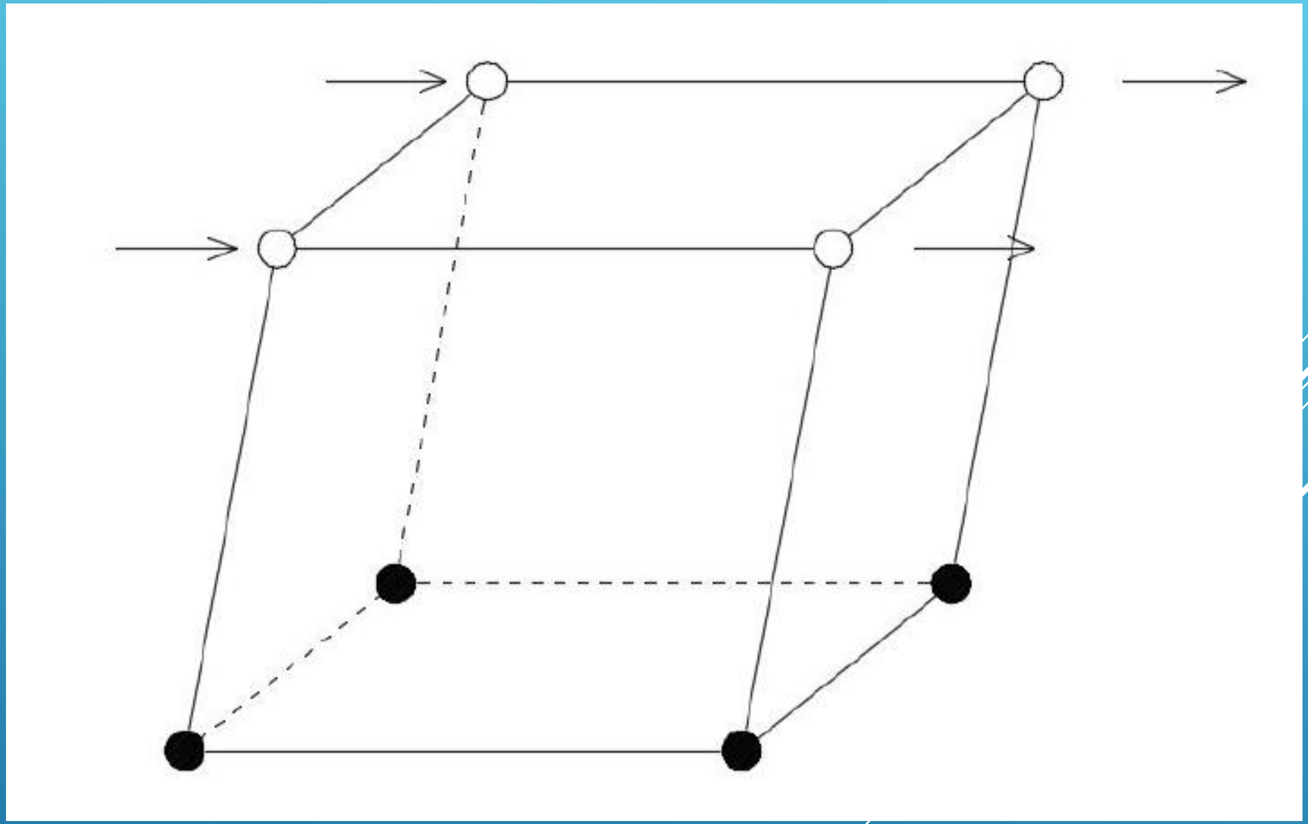
<u>Nonlinear FEM Procedures:</u>
<u>First Level Iteration:</u>

1. *External* Incremental Force.
2. Incremental Displacement from Tangent Stiffness and incr Force.
3. Incremental Strain from Incremental Displacement and strain-disp relation.
4. **Incremental Stress from Incremental strain**
5. *Internal* Incremental Force from Incremental Stress at Gauss Sampling.
6. Use Step 1 and Step 5 to obtain the unbalanced Force.
7. Input unbalance to Step 1 Again.
8. Until unbalanced Force is within the Tolerance.



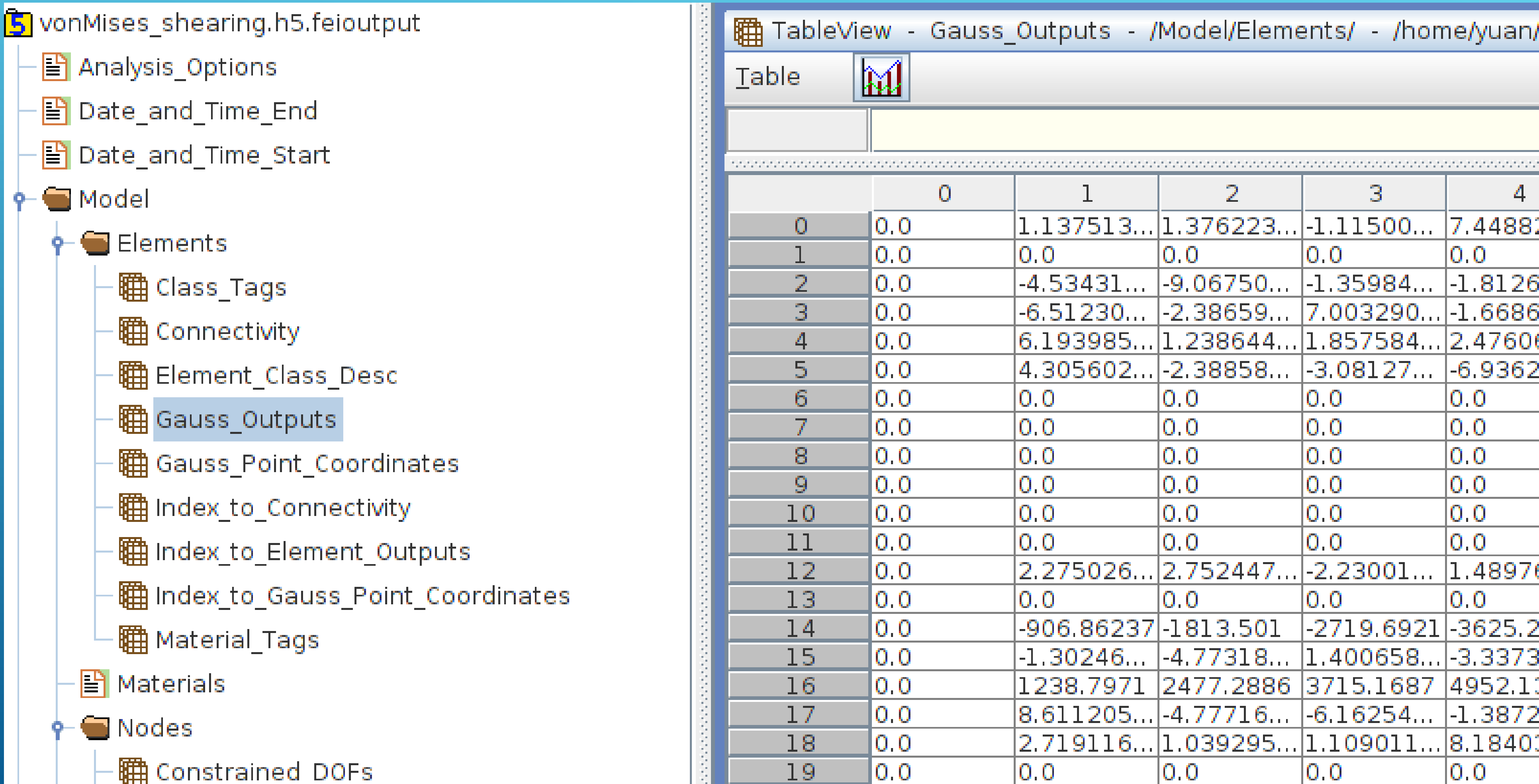
OVERVIEW





BRICK ELEMENT EXAMPLE

HDF5 OUTPUT: Hierarchical Data Format



	0	1	2	3	4
0	0.0	1.137513...	1.376223...	-1.11500...	7.44882
1	0.0	0.0	0.0	0.0	0.0
2	0.0	-4.53431...	-9.06750...	-1.35984...	-1.8126
3	0.0	-6.51230...	-2.38659...	7.003290...	-1.6686
4	0.0	6.193985...	1.238644...	1.857584...	2.47606
5	0.0	4.305602...	-2.38858...	-3.08127...	-6.9362
6	0.0	0.0	0.0	0.0	0.0
7	0.0	0.0	0.0	0.0	0.0
8	0.0	0.0	0.0	0.0	0.0
9	0.0	0.0	0.0	0.0	0.0
10	0.0	0.0	0.0	0.0	0.0
11	0.0	0.0	0.0	0.0	0.0
12	0.0	2.275026...	2.752447...	-2.23001...	1.48976
13	0.0	0.0	0.0	0.0	0.0
14	0.0	-906.86237	-1813.501	-2719.6921	-3625.2
15	0.0	-1.30246...	-4.77318...	1.400658...	-3.3373
16	0.0	1238.7971	2477.2886	3715.1687	4952.1
17	0.0	8.611205...	-4.77716...	-6.16254...	-1.3872
18	0.0	2.719116...	1.039295...	1.109011...	8.1840
19	0.0	0.0	0.0	0.0	0.0

Each Gauss Point has

- 18 Items * Number_of_Timestep
- 18 items includes
  - 6 total strain
  - 6 plastic strain
  - 6 stress

- Order of output in every 6 output
  - Sigma_11
  - Sigma_22
  - Sigma_33
  - Sigma_12
  - Sigma_13
  - Sigma_23

GAUSS OUTPUT FORMAT

READ HDF5 BY PYTHON OR MATLAB

```python
import numpy as np
import matplotlib.pyplot as plt
import h5py
h5in_filename = "vonMises_shearing.h5.feioutput"
h5in=h5py.File(h5in_filename,"r")
outputs_all=h5in['/Model/Elements/Gauss_Outputs'][()]
stress = outputs_all[16 , :-1]
strain = outputs_all[4  , :-1]
plt.plot(strain, stress)
```

```matlab
function [strain] = h52strain(filename)
    resultTr=h5read(filename,'/Model/Elements/Outputs');
    result=resultTr';
    strainAll=[];
    for i=1:size(result,1)/18
        strainAll=[strainAll;result(18*i-17:18*i-12,:)];
    end
    % Extract the first step output:
    strainOne=strainAll(:,1);
    strain=reshape(strainOne,[6,size(strainOne,1)/6]);
end
```

- 1. Forget to add confinement on the Drucker-Prager Materials.
- 2. Use load control with the perfectly plastic materials or materials with plateau.
- 3. Use large strain increment without sub-increments.
  - 3.1 Newton Algorithms requires a good initial guess
  - 3.2 Infinitesimal Strain Assumptions

COMMON MODELING MISTAKES

The infinitesimal strain tensor is defined by

$$\varepsilon = \frac{1}{2}(u_{i,j} + u_{j,i}) \tag{1}$$

The actual strain tensor is defined by

$$e_{ij} = \frac{1}{2}(u_{i,j} + u_{j,i} - u_{k,i}u_{k,j}) \tag{2}$$

Define the unit deformation is $u_{i,j} = u_{j,i} = u_{k,i} = u_{k,j} = d$.
The error of the infinitesimal strain tensor increases with the unit deformation d.

Table 1: Error increase with the unit deformation

d	0.02	0.04	0.06	0.08	0.1	0.12	0.14	0.16	0.18	0.2
Strain	0.0198	0.0392	0.0582	0.0768	0.095	0.1128	0.1302	0.1472	0.1638	0.18
Infinitesimal Strain	0.02	0.04	0.06	0.08	0.1	0.12	0.14	0.16	0.18	0.2
Error	1.01%	2.04%	3.09%	4.17%	5.26%	6.38%	7.53%	8.70%	9.89%	11.11%

INFINITESIMAL STRAIN ERROR ESTIMATION

http://cml01.engr.ucdavis.edu/yuan/education_examples/

MATERIALS AVAILABLE

THANKS!