



Aalto University
School of Engineering

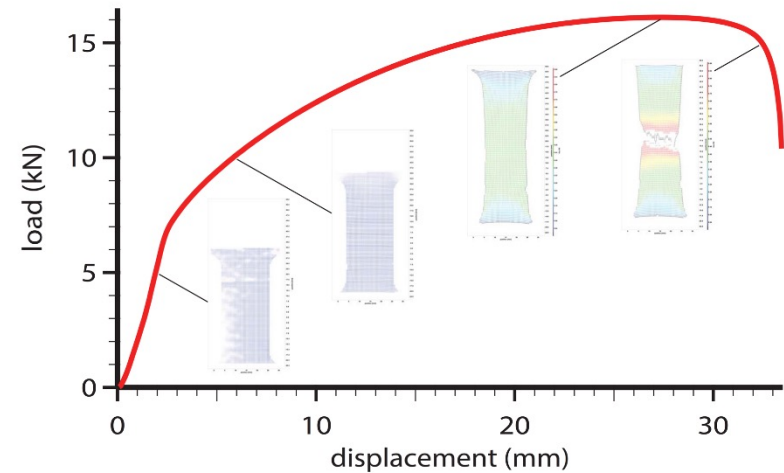
MEC-E6007

Mechanical Testing of Materials

Sven Bossuyt
April 2023

Course Content: *learning from breaking things*

- Load
 - *loadframes, actuators, and grips*
 - *quasi-static, dynamic, and cyclic loading*
- Measure
 - *measurement of force, displacement, and strain*
 - *digital image correlation and other full-field measurement techniques*
- Analyse
 - *selected special challenges in mechanical testing (ask for yours!)*
 - *introduction to inverse problem methodologies in experimental mechanics*



case study

summative assessment of the intended learning outcomes

- *You can treat writing this as the inverse problem to grading it.*

choose a question to be answered with a mechanical test

- provide context situating the question
- conclude how confident you would be to perform this test yourself
 - *or what help you would need*
- could be standardized, from a journal paper, from your research, or hypothetical
 - *cite your sources!*

describe the test

- specimen preparation
- qualitative and quantitative characterization of the specimen before and after the test
- physical quantities to be measured
 - *including measurement methods*
- instruments used
 - *criteria those must satisfy*
- relevant safety precautions
- how to analyse test results
- reasons the test results might not be valid
 - *how to detect that when it is the case*



Aalto University
School of Engineering

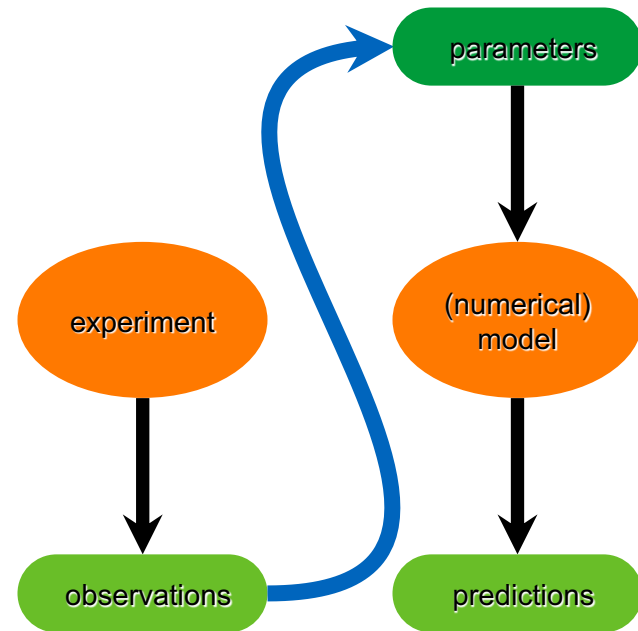
Inverse Problem Methodologies

Inverse Problems:

Mixed Numerical Experimental Techniques

**determine model parameters
from observed data**

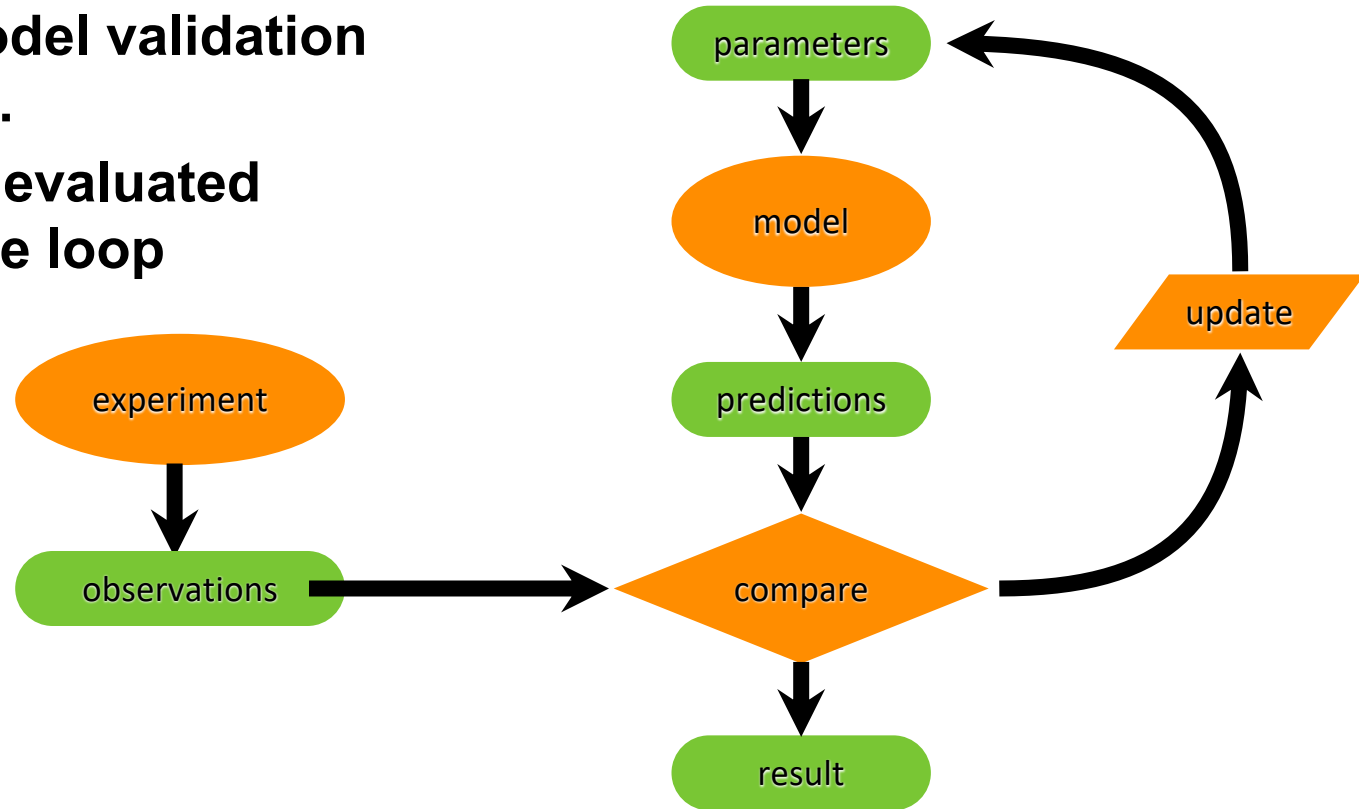
- forward problem predicts observations for given model parameters
 - *iterative solution to find model parameters that agree with observations*
- inverse problem is often ill-posed
 - *regularisation, preferably using a priori knowledge about actual experiment*



iterative solutions

model updating, parameter
identification, model validation
and verification...

forward problem evaluated
many times inside loop



difficulty of inverse problems

Hadamard conditions:

- a problem is well-posed if
 - *a solution exists*
 - *the solution is unique*
 - *the solution depends continuously on the data*
- inverse problems are often ill-posed

regularisation modifies the problem statement to make it more well-posed

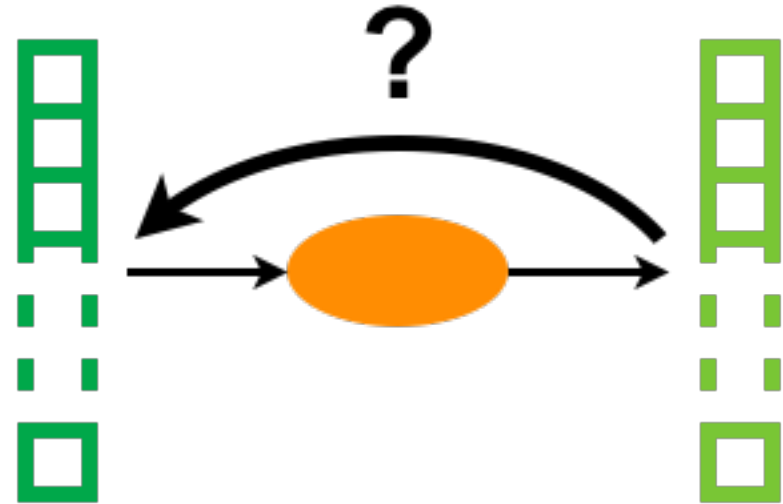
extensive literature and active research in mathematics

Ill-posed problems: Hadamard conditions

a problem is well-posed if

- a solution exists
- the solution is unique
- the solution is stable, i.e., it depends continuously on the data

inverse problems are often ill-posed

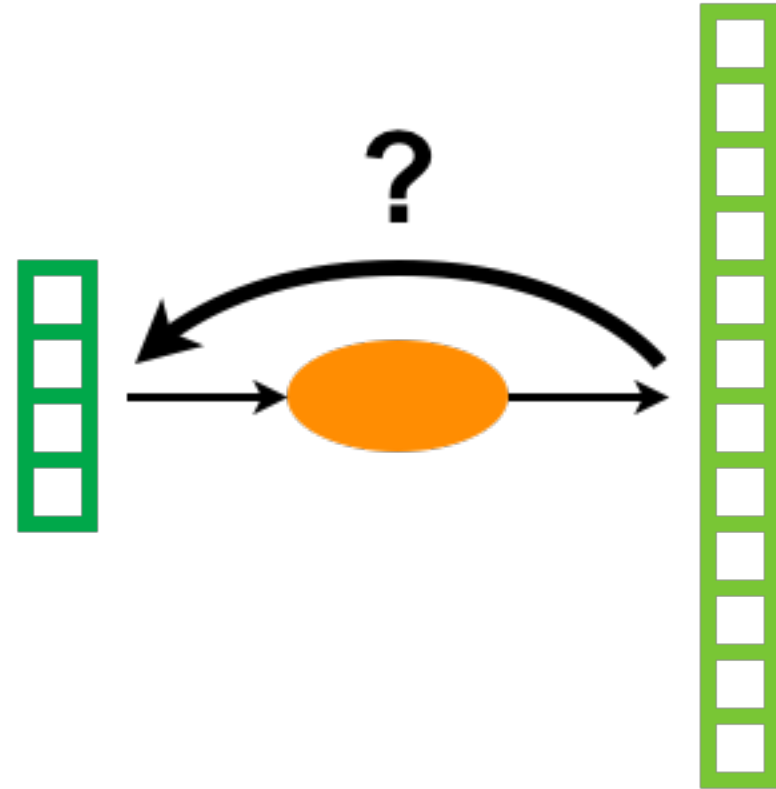


Ill-posed problems: Hadamard conditions

a problem is well-posed if

- a solution exists
- the solution is unique
- the solution is stable, i.e., it depends continuously on the data

inverse problems are often ill-posed

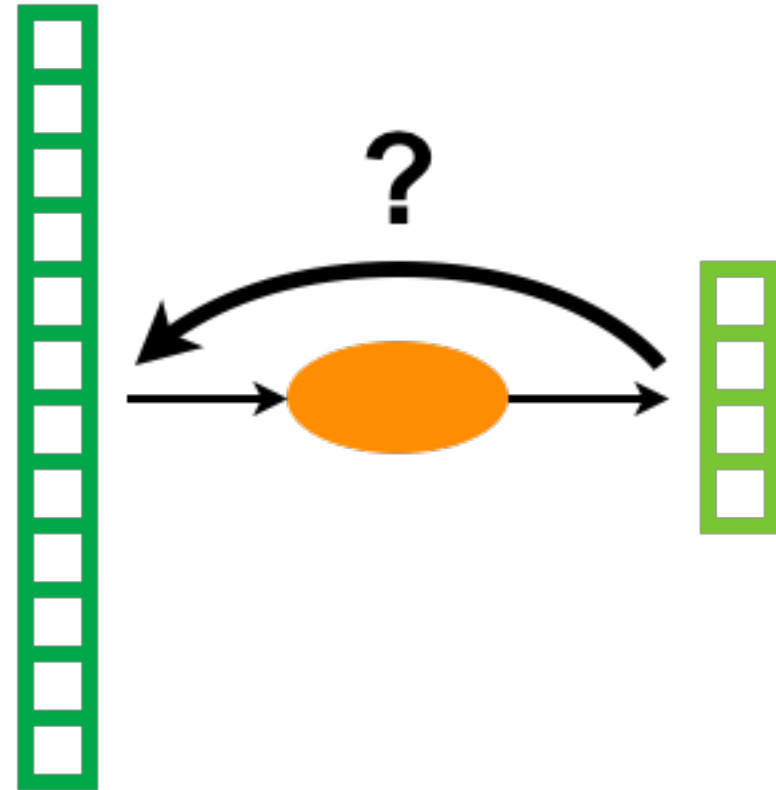


Ill-posed problems: Hadamard conditions

a problem is well-posed if

- a solution exists
- the solution is unique
- the solution is stable, i.e., it depends continuously on the data

inverse problems are often ill-posed

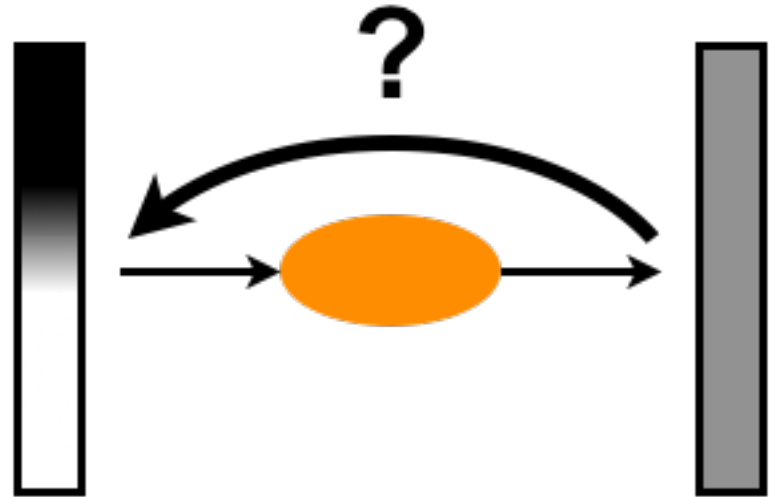


Ill-posed problems: Hadamard conditions

a problem is well-posed if

- a solution exists
- the solution is unique
- the solution is stable, i.e., it depends continuously on the data

inverse problems are often ill-posed

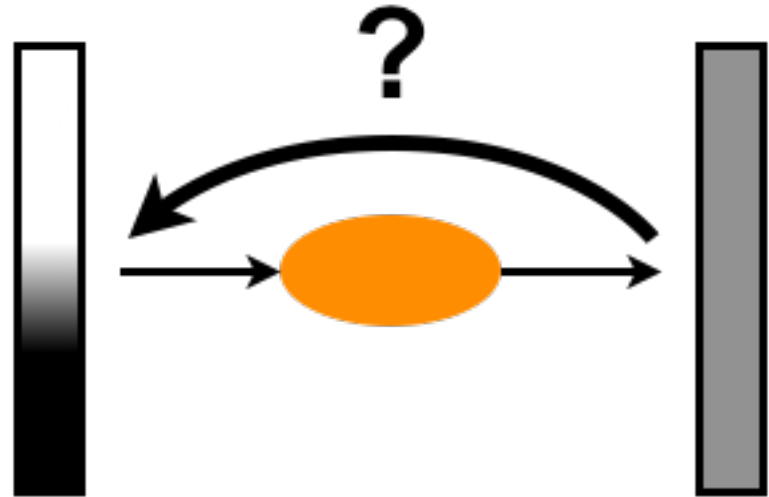


Ill-posed problems: Hadamard conditions

a problem is well-posed if

- a solution exists
- the solution is unique
- the solution is stable, i.e., it depends continuously on the data

inverse problems are often ill-posed

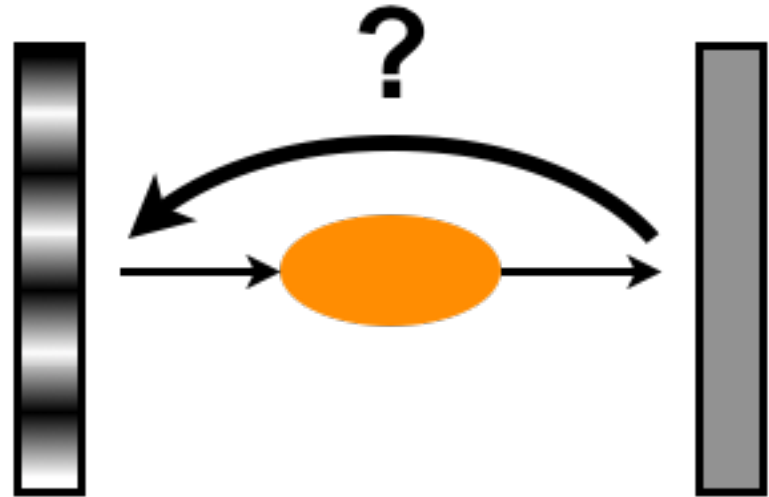


Ill-posed problems: Hadamard conditions

a problem is well-posed if

- a solution exists
- the solution is unique
- the solution is stable, i.e., it depends continuously on the data

inverse problems are often ill-posed



regularization

modify an ill-posed problem to make it well-posed

- explicitly modify equation, e.g., by adding regularization term
- implicitly modify problem statement or solution search space

define a new problem that is “close” to original problem

- recover original problem in limit where regularization parameter \rightarrow zero

codify prior information about solution

- e.g., look for smoothly varying fields
- extreme case: vary simulation parameters to reproduce observations

Inverse Problems in Experimental Mechanics

parameter identification

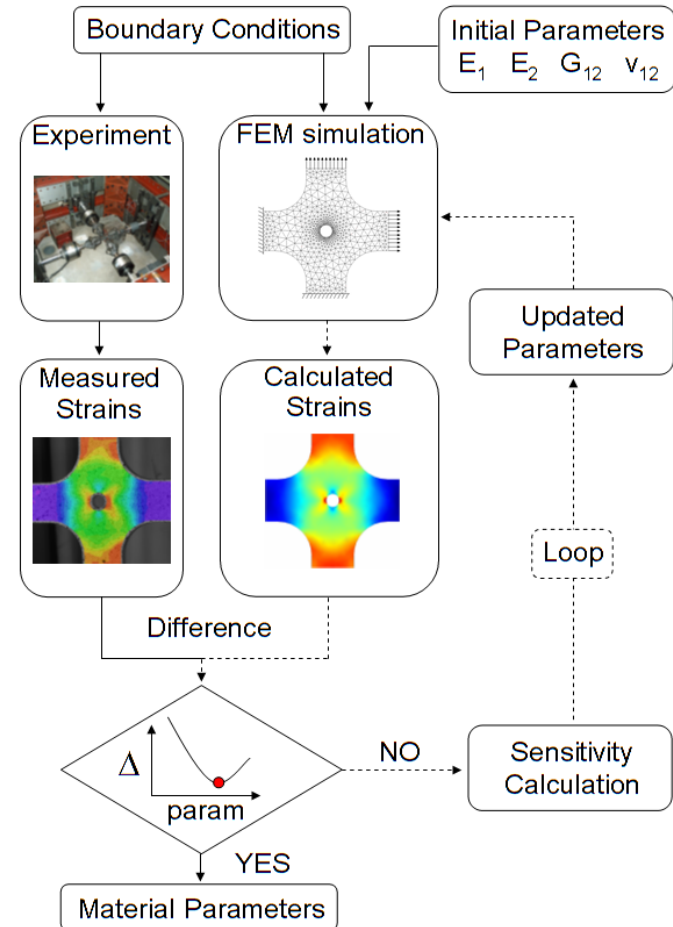
- over-determined
- ill-posed only with non-linearities

full-field measurements

- under-determined
- excessive regularization causes artifacts

forward problems solved by finite element models

- computationally intensive
- shape functions act as regularization
- finer mesh requires more computation and gives less regularization



identification of mechanical properties

simple tests with uniformly loaded gauge section

- inverse problem is trivial
- not always possible experimentally
- lots of tests when many parameters are needed

inverse problem and test with complex loading

- make use of computation power and full-field measurements
- sensitivity depends on test design (e.g., specimen shape)
 - *can test actual components of products in relevant load cases*
- coupled measurements of different parameters

inverse problem methodologies

for identification of mechanical properties

finite element model updating

- iterative solution comparing measured strain or displacement fields with finite element model results to update model parameters

equilibrium gap method

- if material model is wrong, then stresses calculated from measured strains are not in equilibrium, so iterate parameters in model to fix that

integrated DIC

- use finite element model parameters as parametrization of admissible displacement fields in DIC calculation

virtual fields method

- analytically calculate sensitivities of model parameters to measured displacements
- directly write system of equations relating measurements to sought parameters

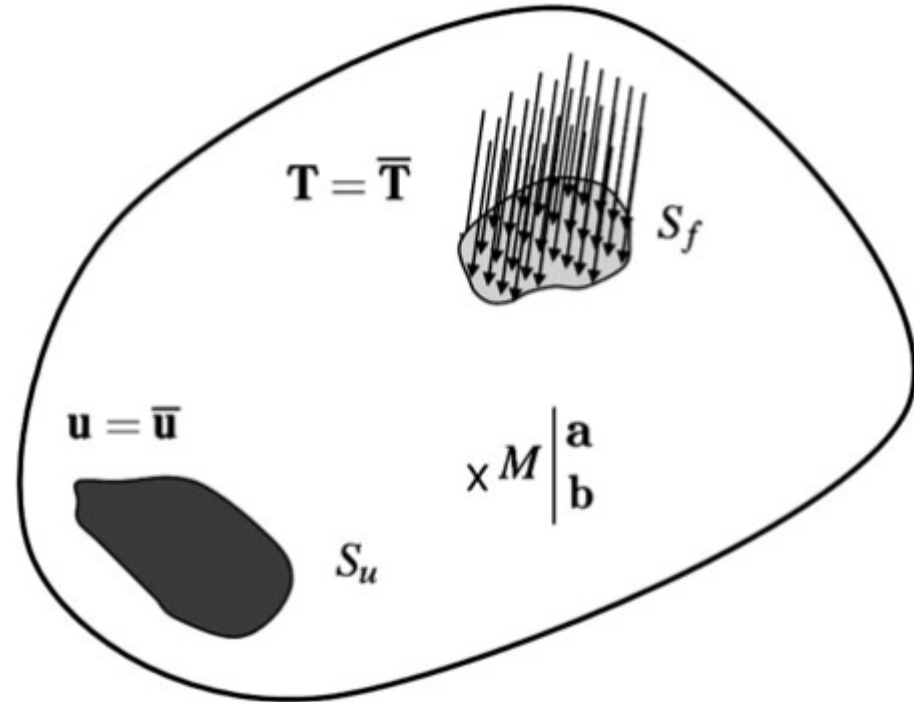


Aalto University
School of Engineering

Virtual Fields Method

mechanics of deformable solids

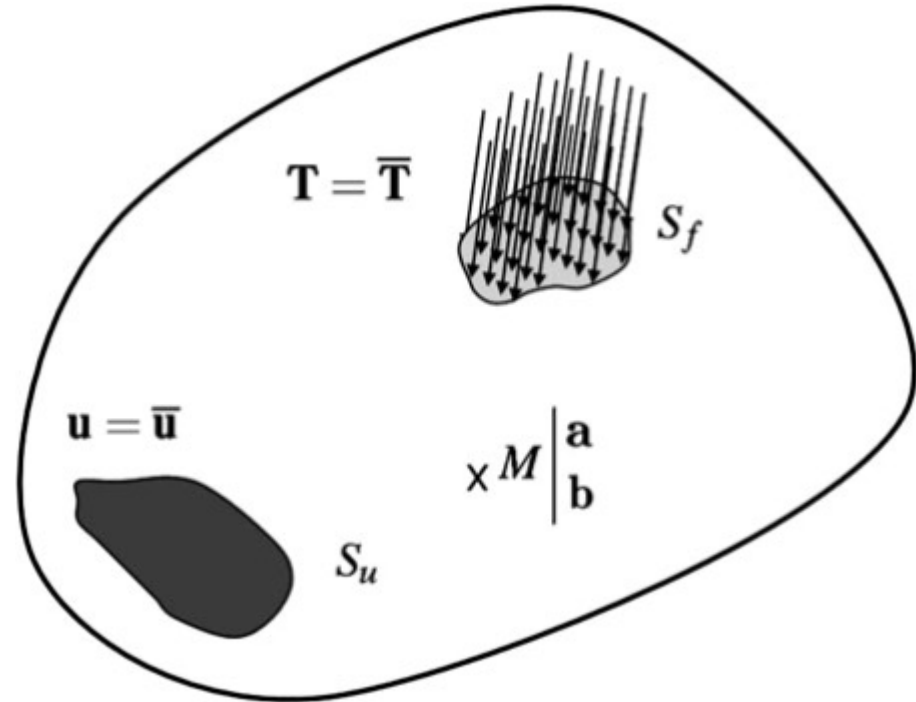
- solid of any shape, subjected to mechanical load
 - displacement field \mathbf{u}
 - strain field $\boldsymbol{\varepsilon}$
 - stress field $\boldsymbol{\sigma}$
- body forces
 - acceleration \mathbf{a} in dynamic problems
 - other body forces \mathbf{b} (e.g. gravity)
- external surface $S = S_f \cup S_u$
- tractions \mathbf{T} specified on S_f
 - free surface has zero traction
- displacements \mathbf{u} specified on S_u



3 types of problems in mechanics of deformable solids

solid of any shape,
subjected to mechanical load

1. find fields \mathbf{u} , $\boldsymbol{\varepsilon}$ and $\boldsymbol{\sigma}$ for known boundary conditions and constitutive behaviour
 - *classical FEM problem*
2. identify constitutive behaviour from measured deformations
3. determine reaction forces



principle of virtual work

variational formulation of theoretical mechanics

- “weak form”
 - *weaker constraints on solution*
- multiply local equilibrium equation with test function and integrate

$$\nabla \cdot \boldsymbol{\sigma} + \mathbf{b} = \rho \mathbf{a}$$

- for vector-valued test function \mathbf{u}^*

$$\int_V (\nabla \cdot \boldsymbol{\sigma}) \cdot \mathbf{u}^* dV + \int_V \mathbf{b} \cdot \mathbf{u}^* dV = \int_V \rho \mathbf{a} \cdot \mathbf{u}^* dV$$

principle of virtual work

- integrate by parts to get $\text{div } \mathbf{u}^*$ instead of $\text{div } \boldsymbol{\sigma}$

$$\int_V \nabla \cdot (\boldsymbol{\sigma} \cdot \mathbf{u}^*) dV = \int_V (\nabla \cdot \boldsymbol{\sigma}) \cdot \mathbf{u}^* dV + \int_V \boldsymbol{\sigma} \cdot \nabla \mathbf{u}^* dV$$

- and use Gauss theorem to convert to surface integral of traction \mathbf{T}

$$\int_V \nabla \cdot (\boldsymbol{\sigma} \cdot \mathbf{u}^*) dV = \oint_S \mathbf{n} \cdot \boldsymbol{\sigma} \cdot \mathbf{u}^* dS = \oint_S \mathbf{T} \cdot \mathbf{u}^* dS$$

- so that

$$\int_V (\nabla \cdot \boldsymbol{\sigma}) \cdot \mathbf{u}^* dV = \oint_S \mathbf{T} \cdot \mathbf{u}^* dS - \int_V \boldsymbol{\sigma} \cdot \nabla \mathbf{u}^* dV$$

principle of virtual work

$$\int_V (\nabla \cdot \boldsymbol{\sigma}) \cdot \mathbf{u}^* dV = \oint_S \mathbf{T} \cdot \mathbf{u}^* dS - \int_V \boldsymbol{\sigma} \cdot \nabla \mathbf{u}^* dV$$

- for symmetric $\boldsymbol{\sigma}$ and kinematically admissible \mathbf{u}^* simplifies to

$$\int_V (\nabla \cdot \boldsymbol{\sigma}) \cdot \mathbf{u}^* dV = \oint_{S_f} \mathbf{T} \cdot \mathbf{u}^* dS - \int_V \boldsymbol{\sigma} \cdot (\nabla^{\text{sym}} \mathbf{u}^*) dV$$

- which yields

$$\oint_{S_f} \mathbf{T} \cdot \mathbf{u}^* dS - \int_V \boldsymbol{\sigma} \cdot (\nabla^{\text{sym}} \mathbf{u}^*) dV + \int_V \mathbf{b} \cdot \mathbf{u}^* dV = \int_V \rho \mathbf{a} \cdot \mathbf{u}^* dV$$

principle of virtual work

weak form of mechanical equilibrium equation in deformable solid

- applied load $\int_{S_f} \mathbf{T} \cdot \mathbf{u}^* dS$
- reaction forces $+ \int_{S_u} \mathbf{T} \cdot \mathbf{u}^* dS$
- internal stresses $- \int_V \boldsymbol{\sigma} \cdot (\nabla^{\text{sym}} \mathbf{u}^*) dV$
- body forces $+ \int_V \mathbf{b} \cdot \mathbf{u}^* dV$
- acceleration $= \int_V \rho \mathbf{a} \cdot \mathbf{u}^* dV \quad \forall \mathbf{u}^*$

remarks

- A continuous body is at equilibrium if the virtual work of all forces acting on the body is null in any kinematically admissible virtual displacement.
- If \mathbf{u}^* is in meters, then the virtual work is in Joules.
- The test functions \mathbf{u}^* are arbitrary weight functions, not related to the actual displacement field \mathbf{u} and independent of stress field $\boldsymbol{\sigma}$.
 - *Expanding \mathbf{u}^* in FEM shape functions is exact, not an approximation.*
 - *The mesh used for these shape functions does not have to conform to the object.*
- The reaction force term disappears for kinematically admissible \mathbf{u}^* .
- If \mathbf{u}^* is discontinuous, then tractions along the discontinuity must be introduced to apply Gauss theorem.
- The Galerkin method and FEM can be derived from this.

linear elastic constitutive equations

$$\sigma = Q\epsilon$$

$$\begin{pmatrix} \sigma_1 \\ \sigma_2 \\ \sigma_3 \\ \sigma_4 \\ \sigma_5 \\ \sigma_6 \end{pmatrix} = \begin{bmatrix} Q_{11} & Q_{12} & Q_{13} & Q_{14} & Q_{15} & Q_{16} \\ Q_{12} & Q_{22} & Q_{23} & Q_{24} & Q_{25} & Q_{26} \\ Q_{13} & Q_{23} & Q_{33} & Q_{34} & Q_{35} & Q_{36} \\ Q_{14} & Q_{24} & Q_{34} & Q_{44} & Q_{45} & Q_{46} \\ Q_{15} & Q_{25} & Q_{35} & Q_{45} & Q_{55} & Q_{56} \\ Q_{16} & Q_{26} & Q_{36} & Q_{46} & Q_{56} & Q_{66} \end{bmatrix} \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \\ \epsilon_4 \\ \epsilon_5 \\ \epsilon_6 \end{pmatrix}$$

linear elastic constitutive equations

- orthotropic

$$\sigma = Q\epsilon$$

$$\begin{pmatrix} \sigma_1 \\ \sigma_2 \\ \sigma_3 \\ \sigma_4 \\ \sigma_5 \\ \sigma_6 \end{pmatrix} = \begin{bmatrix} Q_{11} & Q_{12} & Q_{13} & 0 & 0 & 0 \\ Q_{12} & Q_{22} & Q_{23} & 0 & 0 & 0 \\ Q_{13} & Q_{23} & Q_{33} & 0 & 0 & 0 \\ 0 & 0 & 0 & Q_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & Q_{55} & 0 \\ 0 & 0 & 0 & 0 & 0 & Q_{66} \end{bmatrix} \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \\ \epsilon_4 \\ \epsilon_5 \\ \epsilon_6 \end{pmatrix}$$

linear elastic constitutive equations

- cubic

$$\sigma = Q\epsilon$$

$$\begin{pmatrix} \sigma_1 \\ \sigma_2 \\ \sigma_3 \\ \sigma_4 \\ \sigma_5 \\ \sigma_6 \end{pmatrix} = \begin{bmatrix} Q_{11} & Q_{12} & Q_{12} & 0 & 0 & 0 \\ Q_{12} & Q_{11} & Q_{12} & 0 & 0 & 0 \\ Q_{12} & Q_{12} & Q_{11} & 0 & 0 & 0 \\ 0 & 0 & 0 & Q_{66} & 0 & 0 \\ 0 & 0 & 0 & 0 & Q_{66} & 0 \\ 0 & 0 & 0 & 0 & 0 & Q_{66} \end{bmatrix} \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \\ \epsilon_4 \\ \epsilon_5 \\ \epsilon_6 \end{pmatrix}$$

$$Q_{11} = \frac{E}{1 - \nu^2}$$

$$Q_{12} = \frac{\nu E}{1 - \nu^2}$$

$$Q_{66} = G$$

linear elastic constitutive equations

- isotropic

$$\boldsymbol{\sigma} = \mathbf{Q}\boldsymbol{\epsilon}$$

$$\begin{Bmatrix} \sigma_1 \\ \sigma_2 \\ \sigma_3 \\ \sigma_4 \\ \sigma_5 \\ \sigma_6 \end{Bmatrix} = \begin{bmatrix} Q_{11} & Q_{12} & Q_{12} & 0 & 0 & 0 \\ Q_{12} & Q_{11} & Q_{12} & 0 & 0 & 0 \\ Q_{12} & Q_{12} & Q_{11} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{Q_{11}-Q_{12}}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{Q_{11}-Q_{12}}{2} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{Q_{11}-Q_{12}}{2} \end{bmatrix} \begin{Bmatrix} \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \\ \epsilon_4 \\ \epsilon_5 \\ \epsilon_6 \end{Bmatrix}$$

$$Q_{11} = \frac{E}{1 - \nu^2}$$

$$Q_{12} = \frac{\nu E}{1 - \nu^2}$$

$$Q_{66} = G = \frac{E}{2(1 + \nu)}$$

linear virtual fields method

- substitute linear elastic constitutive equations
- apply principle of virtual work

for simplicity, consider static in-plane cubic case without body forces

$$\oint_{S_f} \mathbf{T} \cdot \mathbf{u}^* dS - \int_V \boldsymbol{\sigma} \cdot (\nabla^{\text{sym}} \mathbf{u}^*) dV + \cancel{\int_V \mathbf{b} \cdot \mathbf{u}^* dV} = \cancel{\int_V \rho \mathbf{a} \cdot \mathbf{u}^* dV}$$

$$\begin{Bmatrix} \sigma_1 \\ \sigma_2 \\ \sigma_6 \end{Bmatrix} = \begin{bmatrix} Q_{11} & Q_{12} & 0 \\ Q_{12} & Q_{11} & 0 \\ 0 & 0 & Q_{66} \end{bmatrix} \begin{Bmatrix} \epsilon_1 \\ \epsilon_2 \\ \epsilon_6 \end{Bmatrix}$$

linear virtual fields method

- substitute linear elastic constitutive equations
- apply principle of virtual work

for simplicity, consider static in-plane cubic case without body forces

$$\begin{aligned} & \int_V \boldsymbol{\sigma} \cdot (\nabla^{\text{sym}} \mathbf{u}^*) dV \\ &= \int_V (Q_{11} \epsilon_1 \epsilon_1^* + Q_{11} \epsilon_2 \epsilon_2^* + Q_{12} (\epsilon_1 \epsilon_2^* + \epsilon_2 \epsilon_1^*) + Q_{66} \epsilon_6 \epsilon_6^*) dV \\ &= \oint_{S_f} \mathbf{T} \cdot \mathbf{u}^* dS \end{aligned}$$

linear virtual fields method

- substitute linear elastic constitutive equations
- apply principle of virtual work

for simplicity, consider static in-plane cubic case without body forces

$$\begin{aligned} & \left(\int_V (\epsilon_1 \epsilon_1^* + \epsilon_2 \epsilon_2^*) dV \right) Q_{11} + \left(\int_V (\epsilon_1 \epsilon_2^* + \epsilon_2 \epsilon_1^*) dV \right) Q_{12} + \left(\int_V (\epsilon_6 \epsilon_6^*) dV \right) Q_{66} \\ & = \oint_{S_f} \mathbf{T} \cdot \mathbf{u}^* dS \end{aligned}$$

linear virtual fields method

- substitute linear elastic constitutive equations
- apply principle of virtual work
- for as many virtual fields as independent coefficients

$$\left(\int_V (\epsilon_1 \epsilon_1^{*(1)} + \epsilon_2 \epsilon_2^{*(1)}) dV \right) Q_{11} + \left(\int_V (\epsilon_1 \epsilon_2^{*(1)} + \epsilon_2 \epsilon_1^{*(1)}) dV \right) Q_{12} + \left(\int_V (\epsilon_6 \epsilon_6^{*(1)}) dV \right) Q_{66} = \oint_{S_f} \mathbf{T} \cdot \mathbf{u}^{*(1)} dS$$

$$\left(\int_V (\epsilon_1 \epsilon_1^{*(2)} + \epsilon_2 \epsilon_2^{*(2)}) dV \right) Q_{11} + \left(\int_V (\epsilon_1 \epsilon_2^{*(2)} + \epsilon_2 \epsilon_1^{*(2)}) dV \right) Q_{12} + \left(\int_V (\epsilon_6 \epsilon_6^{*(2)}) dV \right) Q_{66} = \oint_{S_f} \mathbf{T} \cdot \mathbf{u}^{*(2)} dS$$

$$\left(\int_V (\epsilon_1 \epsilon_1^{*(3)} + \epsilon_2 \epsilon_2^{*(3)}) dV \right) Q_{11} + \left(\int_V (\epsilon_1 \epsilon_2^{*(3)} + \epsilon_2 \epsilon_1^{*(3)}) dV \right) Q_{12} + \left(\int_V (\epsilon_6 \epsilon_6^{*(3)}) dV \right) Q_{66} = \oint_{S_f} \mathbf{T} \cdot \mathbf{u}^{*(3)} dS$$

linear virtual fields method

- substitute linear elastic constitutive equations
- apply principle of virtual work
- for as many virtual fields as independent coefficients

$$\begin{bmatrix} \int_V (\epsilon_1 \epsilon_1^{*(1)} + \epsilon_2 \epsilon_2^{*(1)}) dV & \int_V (\epsilon_1 \epsilon_2^{*(1)} + \epsilon_2 \epsilon_1^{*(1)}) dV & \int_V (\epsilon_6 \epsilon_6^{*(1)}) dV \\ \int_V (\epsilon_1 \epsilon_1^{*(2)} + \epsilon_2 \epsilon_2^{*(2)}) dV & \int_V (\epsilon_1 \epsilon_2^{*(2)} + \epsilon_2 \epsilon_1^{*(2)}) dV & \int_V (\epsilon_6 \epsilon_6^{*(2)}) dV \\ \int_V (\epsilon_1 \epsilon_1^{*(3)} + \epsilon_2 \epsilon_2^{*(3)}) dV & \int_V (\epsilon_1 \epsilon_2^{*(3)} + \epsilon_2 \epsilon_1^{*(3)}) dV & \int_V (\epsilon_6 \epsilon_6^{*(3)}) dV \end{bmatrix} \begin{Bmatrix} Q_{11} \\ Q_{12} \\ Q_{66} \end{Bmatrix} = \begin{Bmatrix} \oint_{S_f} \mathbf{T} \cdot \mathbf{u}^{*(1)} dS \\ \oint_{S_f} \mathbf{T} \cdot \mathbf{u}^{*(2)} dS \\ \oint_{S_f} \mathbf{T} \cdot \mathbf{u}^{*(3)} dS \end{Bmatrix}$$

linear virtual fields method

- substitute linear elastic constitutive equations
- apply principle of virtual work
- for as many virtual fields as independent coefficients

$$\mathbf{A} \cdot \mathbf{Q} = \mathbf{B}$$

$$\mathbf{A} = \begin{bmatrix} \int_V (\epsilon_1 \epsilon_1^{*(1)} + \epsilon_2 \epsilon_2^{*(1)}) dV & \int_V (\epsilon_1 \epsilon_2^{*(1)} + \epsilon_2 \epsilon_1^{*(1)}) dV & \int_V (\epsilon_6 \epsilon_6^{*(1)}) dV \\ \int_V (\epsilon_1 \epsilon_1^{*(2)} + \epsilon_2 \epsilon_2^{*(2)}) dV & \int_V (\epsilon_1 \epsilon_2^{*(2)} + \epsilon_2 \epsilon_1^{*(2)}) dV & \int_V (\epsilon_6 \epsilon_6^{*(2)}) dV \\ \int_V (\epsilon_1 \epsilon_1^{*(3)} + \epsilon_2 \epsilon_2^{*(3)}) dV & \int_V (\epsilon_1 \epsilon_2^{*(3)} + \epsilon_2 \epsilon_1^{*(3)}) dV & \int_V (\epsilon_6 \epsilon_6^{*(3)}) dV \end{bmatrix}$$

$$\mathbf{Q} = \begin{Bmatrix} Q_{11} \\ Q_{12} \\ Q_{66} \end{Bmatrix} \quad \mathbf{B} = \begin{Bmatrix} \oint_{S_f} \mathbf{T} \cdot \mathbf{u}^{*(1)} dS \\ \oint_{S_f} \mathbf{T} \cdot \mathbf{u}^{*(2)} dS \\ \oint_{S_f} \mathbf{T} \cdot \mathbf{u}^{*(3)} dS \end{Bmatrix}$$

special virtual fields

- The equation $\mathbf{A} \cdot \mathbf{Q} = \mathbf{B}$ still depends on the choice of virtual fields.
 - *valid for any choice of virtual fields*
 - *solvable for any linearly independent choice of virtual fields*
 - *numerically stable if condition number of \mathbf{A} is close to 1*
- It is possible to choose the virtual fields such that \mathbf{A} is perfectly conditioned and trivially solvable by making $\mathbf{A} = \mathbf{I}$ the identity matrix.
 - *$\mathbf{A} = \mathbf{I}$ is a set of linear equations*
 - *as many equations as elements of \mathbf{A}*
- It is possible to assess *a priori* how sensitive to noise the calculation is.
 - *coefficients depend on material coefficients \mathbf{Q}*
 - *optimized virtual fields minimize noise sensitivity*
 - *iteratively solving material coefficients and optimizing converges quickly*