CE394M: FEM solvers and errors

Krishna Kumar

University of Texas at Austin

krishnak@utexas.edu

February 10, 2019

Overview

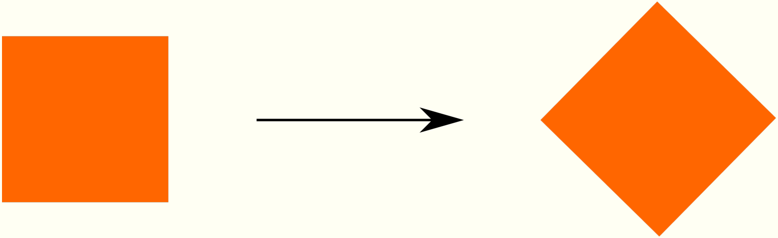
1. Solving non-linear problems
   - Tangent stiffness
   - Newton Raphson

Linear and non-linear problems

- **Linear problems**
  - The response can only be approximated as linear if its deformations/motions are small.
  - In linear analyses, the response to individual load cases can be scaled and added to the results from other linear analyses, which is the principle of superposition.
- **Non-linear problems**
  - Superposition is invalid.
  - The solution is an incremental/iterative process.
  - An iteration is the solution of a system of equations linearised about the current state of the nonlinear physical problem.

Non-linearity in geotechnical engineering

- **Material non-linearity**
  - plasticity
- **Contact**
  - discontinuous source of non-linearity
- **Large deformations and motions of a geotechnical structures**
  - rotation, rigid body motion
  - often ignored, still in the research area.



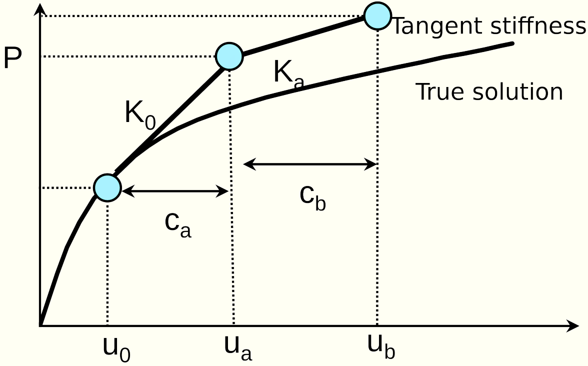
No strains if linear strain-displacement relations are used.

Linear solvers

To solve a system of linear equations of the form $\mathbf{K}\mathbf{a} = \mathbf{b}$, there are two families of methods of solvers that can be used: direct and iterative.

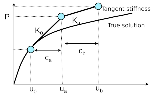
- Direct solvers solve a system of linear equations in a predefined number of steps.
- Methods are based on Gauss elimination, with the most common method being LU decomposition.
- The time required to solve a linear system increases with the number of computer operations performed.
- For a direct linear solver applied to a dense system of size $n \times n$: CPU time $= Cn^3$.
- If the number of degrees of freedom in a finite element simulation is doubled, the time required to solve the system will increase by a factor of 8!

Tangent stiffness method



- Many small increments are need to obtain accurate solution
- Need to perform a parametric study to find the optimum incremental
- Defining increments is very important
- Program - SAGE CRISP

Tangent stiffness method

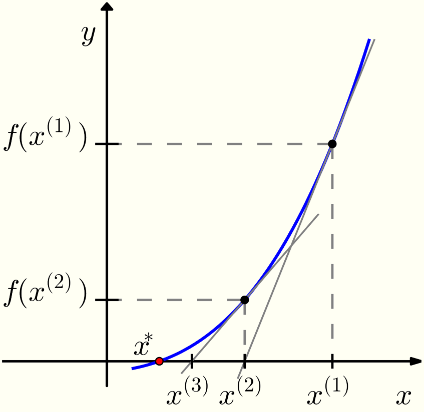


- Many small increments are need to obtain accurate solution
- Need to perform a parametric study to find the optimum incremental
- Defining increments is very important
- Program - SAGE CRISP

In the incremental solution we divide the load into increments $\Delta P = \lambda P$, where λ is also known as a load factor and apply a repeated solution of $\Delta u = K^{-1} \Delta P$.

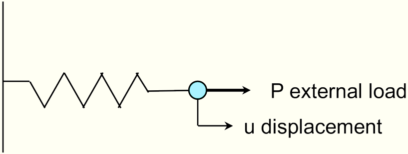
Basically we divide the load into substeps, and treat each as linear - but that is usually not accurate enough and inefficient.

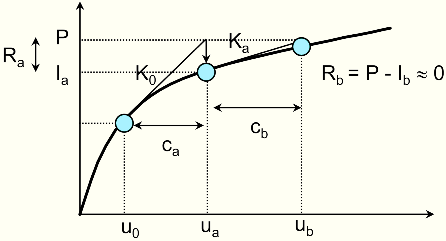
Newton Raphson method



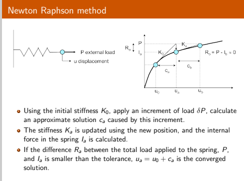
$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

Newton Raphson method





- Using the initial stiffness K_0, apply an increment of load δP, calculate an approximate solution c_a caused by this increment.
- The stiffness K_a is updated using the new position, and the internal force in the spring I_a is calculated.
- If the difference R_a between the total load applied to the spring, P, and I_a is smaller than the tolerance, $u_a = u_0 + c_a$ is the converged solution.



- If R_a is not small, a new displacement correction c_b is calculated by solving $c_b = R_a/K_a$
- The new displacement u_b is updated, and the internal force I_b in the updated configuration is calculated.
- The new force residual R_b is obtained. If $R_b < tolerance$, the solution is converged. If not, continue the iteration.

Newton Raphson method: Tolerance and Convergence

- Program - ABAQUS
- Newton-Raphson is the most standard method to solve nonlinear problems in FE.
- A large error in the initial estimate can contribute to non-convergence of the algorithm.
- **Tolerance**
  - must be small enough to ensure that the approximate solution is close to the exact mathematical solution.
  - must be large enough so that reasonable number of iterations are performed.
- **Quadratic convergence**
  - If the tangent stiffness is calculated correctly, R should reduce quadratically from one iteration to the next.