

Numerical methods for algebraic problems from structural mechanics

C. Vuik, M. van Gijzen, A. Scarpas, C. Kasbergen and T.B. Jönsthövel

Delft Centre for Computational Science and Engineering, Delft University of Technology, the Netherlands

Keywords: deflation, preconditioning, conjugate gradient

A framework for calculating material response, stresses and strains can be found in (Scarpas 2004). In this framework both small and large strains are considered. The balance of forces and conservation of energy yields the virtual work equation. The virtual work equation is highly non-linear and is linearized in order to solve it.

The linearized virtual work equation is discretized by the finite element method with second order shape functions, hence the unknowns lie on the corners and in the centre of the vertices of the elements. The linearized virtual work equation has to be solved for the unknown displacement field. The displacement field has three dimensions, therefore the unknowns on the grid nodes have three components for the x , y and z direction respectively. The discretization of the linearized virtual work equation results into a short hand notation of the static, linear system that has to be solved,

$$K\Delta u = \Delta f \quad (1)$$

where K is the stiffness matrix, Δu the unknown incremental displacement field and Δf the difference between the internal and external stresses.

The stiffness matrix K is symmetric positive definite for all simulations. This is an important property as it is crucial for obtaining good convergence rates for many numerical solution methods. For small meshes the dimension of K allows for direct solution methods if the matrix is non-singular. However, with the refinement of the meshes the dimension and complexity of the linear system increases significantly and other numerical solution methods need to be found. Increasing the dimension of K induces great difficulties for direct solvers because computer memory and CPU power are limited. The condition number is defined as the quotient between the largest and smallest eigenvalues of a matrix. Large condition numbers yield ill-conditioned systems which are therefore difficult to solve. Not only grid refinement but also the non-linear material properties affect the solvability of system (1). When plasticity and viscosity builds up or hyper-elasticity applies, stiffness of the materials changes and stiffness matrix K will have to be reassembled. Large differences in stiffness between material ingredients, e.g. stone and bitumen, will result in large condition numbers and thus in slow converging solvers.

Finding a suitable numerical method for solving an arbitrary linear system can be difficult if the linear system is ill-conditioned. Neither direct solution methods nor iterative solvers will perform, if a solution can be computed at all. Preconditioning of the system can reduce the condition number of the matrix and hence improve convergence rates significantly. Multiply the matrix with the preconditioner to obtain the better conditioned system,

$$M^{-1}K\Delta u = M^{-1}\Delta f \quad (2)$$

Matrix M is the left preconditioner and is preferably an approximation of matrix K . However, mostly preconditioner M is not known explicitly but the operation M^{-1} is replaced by solving v from system $Mv = w$ where M can be any linear solver.

The choice of numerical solvers can be overwhelming and there is still no standard recipe for solving an arbitrary set of equations. Many well known engineering problems like fluid dynamics, the Helmholtz and the Maxwell equations have their own, tailored solution methods. Multi-grid, conjugate gradient (CG), deflation and domain decomposition dominate the spectrum of

modern solvers. It is common practice to combine these methods to take full advantage of their single properties. Only the preconditioned CG method is taken into account, as it is most suitable for positive, semi definite symmetric systems.

Multigrid, domain decomposition and deflation can be used as preconditioners for CG, (Tang 2008). All three methods share the same principle: projection of the linear system onto smaller subspace where the reduced system will be solved and back propagation of the reduced solution to obtain a solution for the original system. Two requirements are satisfied, robustness and the ability of handling very large systems of equations, at the cost of speed per iteration. Obviously, bringing in extra linear solvers will increase the complexity of the solver and the computation time. However, when tailored in the right way, the reduction of the number of iterations due to stability will increase the overall speed of the solver.

Where multigrid and domain decomposition are too complex or lack the agility to be embedded as preconditioner into the CG method, deflation suits best. The deflation method has a strong resemblance with both multigrid and domain decomposition. The philosophy behind deflation is filtering out the slow converging components of the solution. It has been proved that for CG these components correspond to the smallest eigenvalues in the spectrum of the linear system.

If the original system has dimension n a smaller system is introduced with dimension $n - k$ where k corresponds to the

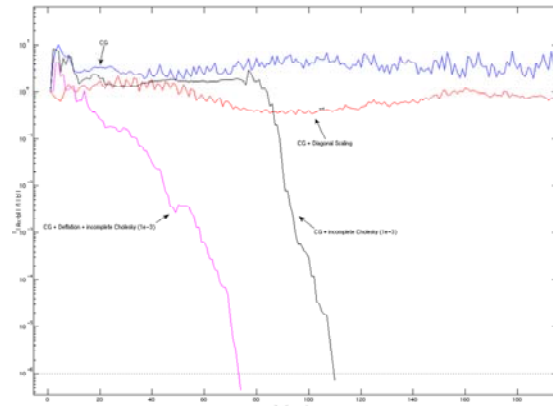


Figure 1 Compression test, two materials with deflated CG solver

number of eigenvalues that have been filtered out. The reduced system is better conditioned and hence convergence rates improve. Both the stability and reduction requirements are satisfied. Unfortunately, calculation of the eigenvalues of large linear systems is both time and resource consuming. Approximations of the eigenvectors are used instead. The eigenvectors correspond to physical aspects of the problem

A small test case shows promising results for preconditioned, deflated CG. Deflation methods DCG (deflated CG) and DPCG (deflated preconditioned CG) are applied to a very simple strain-stress test. A cube containing two materials is subjected to a load. The outside is rubber like and has only elastic material properties. The material within the centre and core of the cube is incompressible, one could think of steel or stone. A distributed load is being applied at the top of the cube and is forcing it downwards within several load steps (static mechanics).

The CAPA-3D software and Matlab are used to calculate the internal and external forces acting on the body during this experiment. The large differences in elasticity between the two materials yield a stiffness matrix with a very high condition number, hence bad convergence rates are expected. The experiment uses a combination of CG, two different preconditioners - incomplete Cholesky and diagonal scaling - and deflation. The eigenvalues and eigenvectors corresponding to the two smallest eigenvalues are selected as deflation vectors and have been computed with Matlab. Consider Figure 1. The CG method with and without diagonal scaling is performing poorly and does not show any progress even after 200 iterations. Preconditioning with incomplete Cholesky yields an increase in speed and the method converges in about 110 iterations. However, when deflation is being applied to CG with incomplete Cholesky preconditioning, another increase in speed is observed and the method converges in about 80 iterations. It is to be expected that increasing the number of deflation vectors would yield even better converge rates but the application of the deflation operator becomes more expensive.

REFERENCES

1. Scarpas, A. *A Mechanics based computational platform for pavement engineering*. Delft: TU Delft, 2004.
2. Tang, Y.M. *Two-Level Preconditioned Conjugate Gradient Methods with Applications to Bubbly Flow Problems*. Delft: TU Delft, 2008.