ELMER - A finite element solver for multiphysics

Mikko Lyly, Juha Ruokolainen and Esko Järvinen
CSC - Scientific Computing Ltd.

1 Introduction
ELMER is a software package for solving coupled partial differential equations generated by multiphysics problems in continuum mechanics. The program was developed at CSC in collaboration with Finnish universities, research centers and industry [1].

The program consists of the following three separate parts (see Figure 1):

- The physical models are defined in ELMER Front (1 and 2).
- The resulting PDEs are discretized and solved by ELMER Solver (3).
- The results are postprocessed and visualized by ELMER Post (4).

The purpose of this article is to give an overview of the graphical user interfaces, basic solution strategies, and latest development of the software. A more detailed description of the program with extensive numerical examples can be found from [2, 3, 4] and by visiting our web site at http://www.csc.fi/elmmer/.

![Figure 1. Work flow of ELMER.](image)

2 ELMER Front
ELMER Front is a tool for initializing the computational process in ELMER. The program reads geometrical data from CAD-files, allows the user to build the mathematical models graphically, generates finite element meshes, and finally produces input data for ELMER Solver. The graphical user interface of ELMER Front is depicted in Figure 2.
3 ELMER Solver

The physical models defined in ELMER Front are systems of coupled non-linear partial differential equations. The equations can be solved by ELMER Solver either directly (strong coupling), or by using the following iterative solution algorithm consisting of time integration and three nested iteration loops (weak coupling):

- Discretize the problem implicitly in time (e.g. backward Euler). For each time step:
- Split the coupled system into a sequence of small uncoupled subproblems by relaxation methods (e.g. Jacobi or Gauss-Seidel). For each relaxation:
- Solve the non-linear subproblems iteratively by successive linearizations (e.g. Picard or Newton iterations). For each linearization:
- Discretize the linearized equations by domain decomposition and adaptive stabilized finite element methods. Solve the algebraic systems iteratively in parallel by appropriate Krylov space methods (e.g. CG, GMRES or BiCGStab)
The outer relaxation loop is terminated when the accuracy of the solution reaches the *steady state convergence tolerance* defined by the user. The middle loop is terminated when the error gets smaller than the *nonlinear system convergence tolerance*. The stopping criterion for the inner iteration is given by the *linear system convergence tolerance*. By choosing these parameters properly with respect to the *mesh parameters* and *time step sizes* the user can flexibly control the accuracy of the numerical solution.

The current version of ELMER Solver provides stabilized finite element discretizations and weakly coupled solutions of the following physical models:

- Transport (diffusion-advection-reaction equations)
- Fluid flow (incompressible and compressible Navier-Stokes equations)
- Elasticity (large deformations and displacements, modal analysis)
- Acoustic scattering (Helmholtz equations)
- Electric and magnetic fields (Maxwell equations)

Other physical models can easily be added to the list by the user.

![Figure 3. Monitor of ELMER Solver.](image)

## 4 ELMER Post

The results produced by ELMER Solver can finally be visualized by ELMER Post, see Figure 4. ELMER Post operates with the data specific to the unknown functions (temperature, velocity, pressure, displacement etc.) defined in the mathematical model. The program is able to plot e.g. contours and vector fields, animate the results of transient analysis, and manipulate the computed data using the built-in MAT/C-language (for instance computation of heat fluxes from temperature distributions).
5 Example

Let us next consider a numerical example about fluid-structure-interaction (FSI). We consider a part of a human aorta and use ELMER to simulate the pulsatile flow inside the blood vessel. The example is related to the DynAMo project (Dynamic and Adaptive Modelling of the Human Body) coordinated by the Academy of Finland (more details can be found from http://www.tut.fi/dynamo/).

In addition to the evident blood transporting function, the large arteries of a human body behave as “cushions”, softening the pulsatile flow generated by the contracting left ventricle of the heart into a steady flow at the periferal arteries. Aorta and other viscoelastic arteries store about a half of the volume of the pulse during the systolic phase and drain it forward during diastole to the peripheral tissues.

We approximate the geometry of the aorta by a 90-degrees curved tube with diameter 20 mm and wall thickness 2 mm (see Figure 5). The material of the blood vessel is assumed elastic with Young modulus $4.5 \cdot 10^5$ Pa, Poisson ratio 0.3 and density 1100 kg/m$^3$. The blood is considered incompressible with density $1050$ kg/m$^3$ and viscosity $3.5 \cdot 10^{-3}$ Pa s. The systolic phase of the heart beat
is modeled by a 5 ms x 1333 Pa (10 mmHg) pressure pulse at the inlet of the conduit.

The partial differential equations to be solved are in this case the incompressible Navier-Stokes equations coupled with the non-linear elasticity equations. The splitting of the problem is done by the Gauss-Seidel relaxation method (also known in multiphysics as the method of sequential iteration). The subproblems are linearized by Picard’s and Newton’s methods and solved numerically by triquadratic stabilized finite elements.

The results from the analysis are shown in Figure 5. The gray scales on top left represent the domain decomposition used in the parallel solution of the problem. The three other subplots show the deformation of the blood vessel (the displacements are exaggerated). The gray scales in the figures represent blood pressure. The velocity field is indicated by the arrows.

![Figure 5. Geometry of the aorta model (top left) and the solution 3 ms (top right), 6 ms (bottom left) and 9 ms (bottom right) after the pressure pulse has entered the conduit. The gray scales in the top left figure represent the domain decomposition used in the parallel solution of the problem. The displacements are exaggerated.](image_url)
6 Conclusions

The development of ELMER was originally part of a Finnish national CFD programme coordinated by the Technology Development Centre of Finland (TEKES) in 1995-1999 [1]. The current development is strongly related to the Miksu and DynAMo programs coordinated by TEKES and the Academy of Finland (see http://www.csc.fi/miksu/ and http://www.tut.fi/dynamo/).

In the Miksu project the objective is to develop ELMER towards a computational tool capable of tackling problems related to Micro Electro Mechanical Systems, MEMS. In the DynAMo project the program is used to model the pulsatile flow in human blood vessels. In both cases, the modeling of fluid-structure-interaction (FSI) plays an important role.

The ELMER package is available from CSC; for academic use the program is provided without charge.

References


