From COMSOL to NGSolve: a very personal voyage

Philipp W. Schroeder

Institute for Numerical and Applied Mathematics
Georg-August-University Göttingen, Germany

2nd NGSolve User Meeting 2018

Georg-August-University Göttingen, Germany

4–6 July 2018

Outline

1. Background
2. Finite element method
3. Linear/nonlinear solvers
4. Time-stepping
5. Conclusion

Some brief COMSOL background

COMSOL Multiphysics

- commercial finite element package
- has a GUI from which everything can be controlled
- different packages (modules) available:
  - electromagnetics
  - structural mechanics & acoustics
  - fluid flow & heat transfer
  - chemical engineering
- but: basic package (no specific modules) has all core functionality
- there exists a MATLAB interface

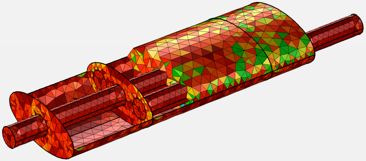
How did I come to COMSOL?

University

- used in (some) FEM lectures at University of Göttingen
  - students can compute FE solutions with only knowledge of PDE
  - everything else (weak formulation etc.) can be ignored

Industry

- internship and master's thesis in automotive industry
  - optimise thickness of thermal insulation for pressure tanks
  - use phase change material to improve performance of adsorber



Why do I use NGSolve now?

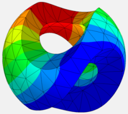
Christoph Lehrenfeld

- came to Göttingen in September 2016
- convinced me to come to 1st NGSolve user meeting

I am using NGSolve ever since!

General factors which are essential for me

- Python interface!!!
- forum on `ngsolve.org`
- (Christoph, of course ;))

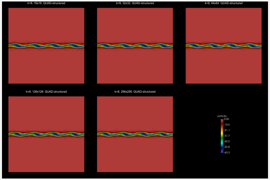


What do I use NGSolve for?

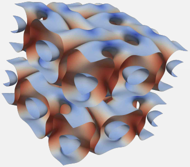
Incompressible Computational Fluid Dynamics

- time-dependent incompressible Navier–Stokes equations
- exactly divergence-free methods
- pressure-robustness
- treatment of dominant convection via upwinding

Kelvin–Helmholtz instability



Decaying homogeneous turbulence



Geometry and meshing

- 1D / 2D / 3D domains supported
- segm / trigs / quad / tet / hex / prism / pyramid
- hybrid meshes possible
- curved boundaries possible
- no hanging nodes allowed

- GUI very convenient to use
- ✘ periodicity complicated
- ✔ complicated meshes: relatively easy
- ✘ manual meshes: only after export
- h-adaptivity: works pretty well

- Python: sometimes complicated
- ✔ periodicity more natural
- ✘ manual meshes: complicated
- h-adaptivity: coarsening?

FE spaces

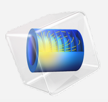
- H^1, L^2, $H(\text{div})$, $H(\text{curl})$

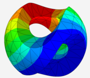
✘ cannot add more spaces manually
✘ H^1: $k \leqslant 7$
✘ L^2: $k \leqslant 10$ (2D); $\leqslant 7$ (3D)
✘ $H(\text{div})$: $k \leqslant 3$ (2D); $\leqslant 2$ (3D)
✘ $H(\text{curl})$: $k \leqslant 3$ (2D); $\leqslant 2$ (3D)
✘ no facet spaces, no hybrid methods

✔ can add whatever you like
✔ arbitrary high-order

This flexibility was my main reason to switch to NGSolve!

Weak form → discrete system





- variational formulation works pretty much the same way
- accuracy of numerical integration can be chosen

✘ only scalar-valued input possible
✔ simply add nonlinear expressions
✘ static condensation???

✔ `InnerProduct()`
✔ `Assemble()` / `Apply()`

RAM in COMSOL

possible not very economical...

Linear systems

- pardiso, mumps available

✔ pardiso and mumps are integrated
✔ automatically performs reiteration
✔ (F)GMRES, BiCGStab, CG
✔ (h/p) multigrid, algebraic multigrid
✔ all kinds of smoothers

✔ `sparsecholesky`
✔ BDDC works great for me

Iterative solvers in COMSOL

Work good for H^1 methods. But: I was not able to obtain good results for $H(\text{div})$...

Nonlinear systems

- ✔ Newton solver integrated
- ✔ detects nonlinearity and acts accordingly
- ✔ works very robustly

- basically no personal experience

Implicit and explicit methods

- ✔ BDF: adaptive order, adaptive time-step
- ✔ generalised-α: adaptive time-step
- ✔ (explicit) Dormand-Prince: adaptive time-step

- ✔ whatever you can imagine
- possible solution of nonlinear systems necessary

IMEX in COMSOL

BDF variant: possible, but not efficient. Runge–Kutta variant: not clear if possible.

Conclusion

convenience / usability	**flexibility / performance**
✔ easy to get started	✔ great for research
✔ very ‘foolproof’	✔ fast
✘ (possibly) not optimal for research	• extensive knowledge of FEM necessary
✘ performance not competitive	
• good and convenient if nothing ‘fancy’ is desired	
• transparent (for commercial tool)	
✔ huge reference manual	

Thank you for your attention

Questions