# MULTILEVEL METHODS: READY FOR INDUSTRY?

F. Hülsemann, EDF R&D

École thématique Multigrid, November 2014

# SUMMARY

# CREDITS

| | |
|---|---|
| ▪ ***Sana Chaabane-Khelifi*** | **(EDF/École Centrale Paris)** |
| ▪ **Jean-François Deldon** | **(École des Ponts Paris Tech)** |
| ▪ **Frédéric Magoulès** | **(École Centrale Paris)** |
| ▪ **Emilien Santerre** | **(Supélec)** |
| ▪ **Pascal Tremblay** | **(Université Laval)** |
| ▪ **Didier Colmont** | **(EDF R&D)** |
| ▪ **Fabien Decung** | **(EDF R&D)** |
| ▪ **Namane Méchitoua** | **(EDF R&D)** |
| ▪ **Nicolas Tardieu** | **(EDF R&D)** |

**Affiliations at the time of intervention.**

**Funding from ANRT grant CIFRE Nº 1283/2009 is gratefully acknowledged.**

# CONTEXT: EDF

**EDF: Électricité de France**

now an international electricity utility (mainly: F, GB, I, PL, PRC)

**2013 numbers:**

**€72.7 billion** in sales

**39.3 million** customers

**> 159,000 employees** worldwide

**139.5 GWe** installed net production capacity

**642.6 TWh** generation 2013

# CONTEXT: EDF R&D

**EDF R&D:** A single R&D division for all Group businesses

- **Generation**
- **Energy management**
- **Customers and sales**
- **Renewable energies**
- **Electrical networks**
- **Information technology**

**R&D in 2013 numbers:**

- **2100 employees**
- **150 ongoing PhD thesis**
- **543 M€ budget**
- **8 centres ( 3xF, D, GB, PL, PRC, I)**

# CONTEXT: EDF R&D

**The topics of the R&D at EDF cover (potentially) everything that concerns**

- the generation of electricity (nuclear, hydroelectric, wind, …, )
- its distribution (different kinds of networks)
- its commercialization.

**As a matter of fact, EDF has decided**

- to develop certain simulation codes in house or as part of a consortium
- and, in certain cases, to distribute the codes under open source licenses.

| Application area | Code | URL | Licence |
|---|---|---|---|
| CFD | openTelemac | www.opentelemac.org | GPL/LGPL |
| | *Code_Saturne* | code-saturne.org | GPL(v2) |
| Structural mechanics | Code_Aster | www.code-aster.org | GPL(v2) |
| Thermodynamics | SYRTHES | rd.edf.com/syrthes | GPL |

# SCIENTIFIC COMPUTING AT EDF (R&D)

**Since 2006, EDF R&D has been present in the Top500 list.**

**In the latest list (November 2014), “we” have four entries**

| Top500 position | Type | #cores | Tflop/s (Linpack) |
|---|---|---|---|
| 73 | BlueGene/Q | 65536 | 715 |
| 123 | Xeon cluster | 14448 | 406 |
| 142 | Xeon cluster | 18144 | 391 |
| 394 | Xeon cluster | 16320 | 191 |

- **Note: No accelerators (GPGPU/Xeon Phi) in these machines for the time being.**
- **Convenient to have: Shared memory nodes with 512GB/ 1TB/ 2TB RAM in the Xeon clusters.**

# MULTILEVEL METHODS IN INDUSTRY (1/2)

Compared to research settings, the typical simulations in industry hardly qualify as **“heroic computing”** (Exascale, Tier0,  …. [your buzz word here] ).

Simulation tools in industry are used **all the time** and by **everybody**.

- **“All the time”   =>  large variety of applications**
- **“by everybody”   =>  not only specialists**

# MULTILEVEL METHODS IN INDUSTRY (2/2)

Earlier this year, **Klaus Stüben** summarized a number of lessons learned from industrial applications of the SAMG solver.

A personal selection:

- Parameters considered harmful. (my words)
- If in doubt, go for robustness rather than absolute speed.
- Documentation
- Clean error handling/messages.
- ….

# MULTILEVEL METHODS AT EDF (R&D)

**Over the years, we have looked at a number of methods for the different application areas. In the order of presentation:**

- **Wavelet-based algebraic multigrid method (WAMG)**
- **Hybrid geometric/algebraic multigrid (HMG) for structural mechanics**
- **Stabilized aggregation AMG**
  - **Algebraic stabilization**
  - **Finite volume stabilization**

# WAVELET-BASED ALGEBRAIC MULTIGRID METHOD

A certain number of publications on the so-called **“Wavelet-based algebraic multigrid method (WAMG)”** have appeared.

Upon closer inspection, the WAMG method based on the Haar basis (which is the version that was “promoted”) is nothing else but the plain aggregation method.

**Conclusion:**

- A method is not optimal, just because it uses several levels.
- As WAMG(Haar) shows exactly the behavior that one expects from plain aggregation without stabilization: **No follow-up.**

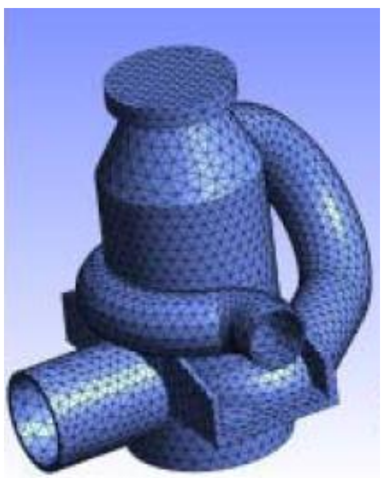
# HMG: HYBRID MULTIGRID PRECONDITIONER FOR STRUCTURAL MECHANICS (1/2)

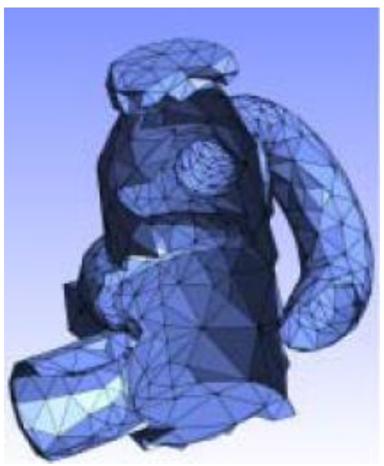
- The HMG approach of Tardieu/Tremblay combines aspects from geometric and algebraic multigrid in the following way:
  - **Algebraic:** Selection of coarse grid vertices based on matrix entries on the next finer level.
  - **Geometric:** Creation of a coarse grid based on the selected vertices (re-meshing).
- Linear interpolation is used in the construction of the inter-grid operators.
- The coarse grid operator is given by the Galerkin product.
- The prototype implementation can deal with a **mix of structural elements (shells) and 3D finite elements.**
- It has been tested as preconditioner for GMRES(30) and CR.
- Sequential implementation using VTK and PETSc.

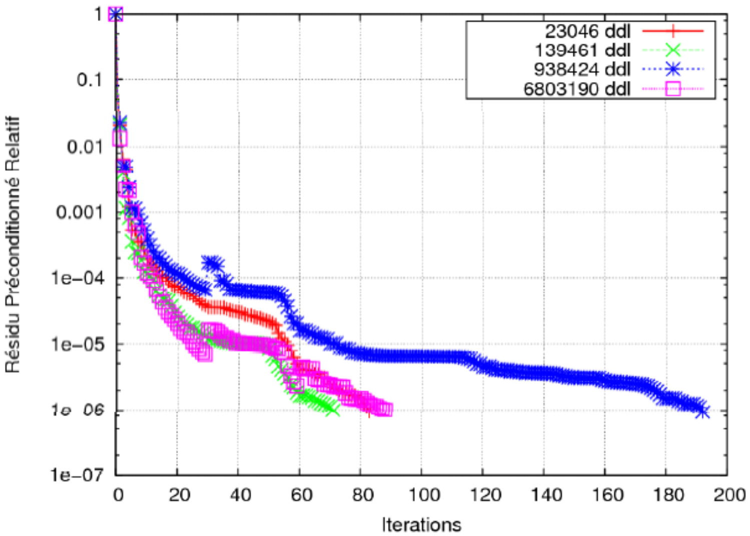
# HMG: HYBRID MULTIGRID PRECONDITIONER FOR STRUCTURAL MECHANICS (2/2)

## Results:

- Good convergence (< 100 it.) on 3D FE elasticity problems, even on industrially relevant geometries.








- Linear interpolation in the construction of the inter-grid operators is suboptimal for shells, but the convergence remains acceptable (250/350 it.), where GMRES(30)-SOR diverges.
- Lagrange multipliers remain difficult.

# STABILIZED AGGREGATION AMG

- For linear finite element discretizations of heterogeneous diffusion problems in 2D and 3D, the following plain aggregation AMG gave satisfactory results, even as stand-alone solver:
  - N-times pairwise aggregation (N = 3 or 4), based on strength of connection
  - W(1,1)-cycle (forward GS as pre-, backward GS as post-smoother)
  - Recombination of iterants as suggested by A.Brandt
- It performed better than CG(ILU0) or CG(V(1,1)), both in iteration count and in wall clock time.
- However, it was not quite as fast as the k-cycle AGMG.
- Scope for improvement:
  - Krylov acceleration
  - Optimization of the implementation

# AGGREGATION-BASED AMG FOR *CODE_SATURNE*

*Code_Saturne*: **EDF's general purpose CFD tool for incompressible and slightly compressible, single phase flows**

- **Navier-Stokes with various turbulence models**
- **Co-located finite volumes**
- **Arbitrary polyhedral meshes**
- **Semi-implicit schemes/operator splitting**

- **Specific modules: combustion, radiative heat transfer, atmospheric flows**

- **Other CFD activities:**

  **Multiphase (or multifield) flows: water/steam. Models for interfacial momentum transfer, interfacial energy transfer terms, heat losses and porosity…**

# AMG METHODS FOR *CODE_SATURNE*

**Common features of the different in-house AMG methods for *Code_Saturne*:**

- **Aggregates based on “strength of connectivity”**
- **Exploitation of FV discretization data for scaling/construction of coarse grid *diffusion* operators**
- **2009: AMG for scalar diffusion operator (sym.)**
- **2012: AMG for scalar convection-diffusion operator (non-sym.)**
- **2013: AMG for scalar sum of weighted diffusion operators (non-sym.)**
- **When several operators are present, each operator is treated separately. Example : plain aggregation for convection part, scaled coarse grid operator for diffusion part.**
- **Although we refer to it as AMG methods, the approach takes into account the discretisation, the mesh and the operators. *(FV-AMG?)***

# SUM OF DIFFUSION OPERATORS

**In one formulation of multiphase flows, we have to solve the following scalar PDE:**

$$\sum_{k=1}^{N_p}\left[C_k(x,y)\,\mathrm{div}\left(-\mathrm{D}_\mathrm{k}(x,y)\nabla u(x,y)\right)\right]= f(x,y)$$

**with**

$$C_k(x,y) > 0 \ \text{ in } \ \Omega, \quad k = 1 : N_p$$

$$D_k(x,y) \geq 0 \ \text{ in } \ \Omega, \quad k = 1 : N_p \quad \text{ and } \quad \sum_{k=1}^{N_p} D_k(x,y) > 0 \ \text{ in } \ \Omega$$

**In our FV discretization, the resulting matrix is non-symmetric.**

# COMPARISON WITH DIFFERENT AMG CODES

**Overview of the implementations included in the comparison:**

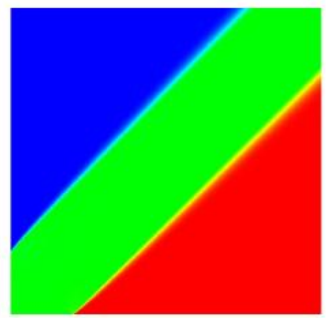
- PETSc v3.3.0p5 (GCC4.4.5, optimized build)
- HYPRE v2.8.0b (BoomerAMG)
- ML v6.2
- GAMG (part of PETSc v3.3.0p5)
- AGMG v3.1.2 (last version under GPL)

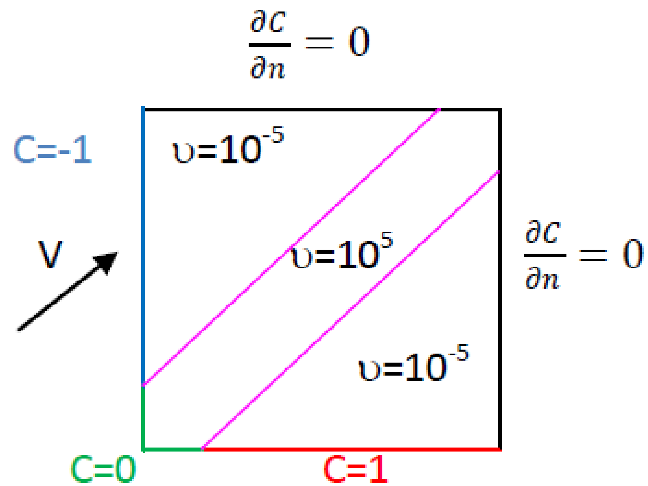
**Yes, other codes exist, but we did not (and do not) have the time to do more tests.**

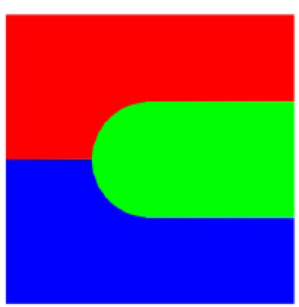
**However, we can share some test cases. If you are interested, get in touch.**

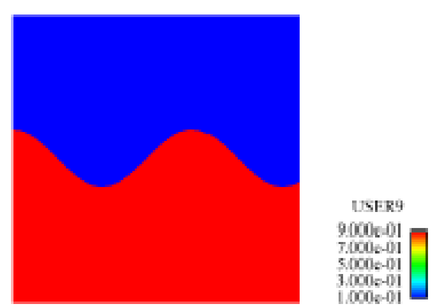
# TEST PROCEDURE

- Scalability tests on 5 simple 2D geometries (100x100, 500x500, 1000x1000)
- Application to 5 industrial, 3D data sets
- ML, GAMG and BoomerAMG as solver and PC for BiCGstab
- Criteria :
  - Operator complexity
  - Number of iterations
  - **Wall clock time** (sequential)

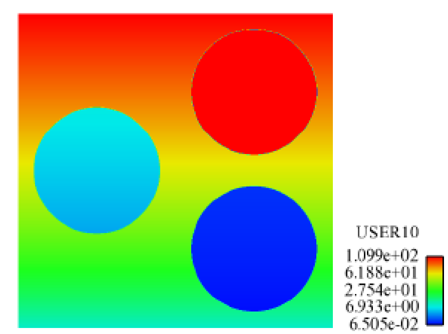






(a) $D_1$



(b) $C_2$

# GENERAL OBSERVATIONS

- All solvers converged for all test cases **(after parameter tuning!).**
- The number of user definable parameters varies considerably:

| AGMG | BoomerAMG | GAMG | ML |
|---|---|---|---|
| 5 (+ 13) | 30 | 10+58+84=152 | 16+196=212 |

(number of options in the PETSc interface, including mg parameters for GAMG and ML)

- We did our best, but given these numbers, it was impossible to test all parameters (and their combinations).
- We tried to find **one set of parameters for each solver** that minimizes the (sequential) wall clock time for our test cases. Changes were only made in case of breakdown.

# SOLVER CONFIGURATIONS

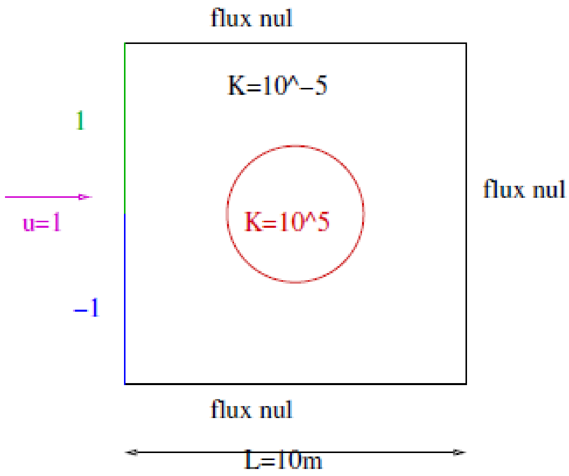
| ML | GAMG |
|---|---|
| Energy minimization | MIS (max. independent set) |
| Smoothed aggregation | Std. aggregation |
| V(2,2) GSlex | V(1,1) GSlex |
| LU (PETSc) on coarsest level | LU (PETSc) on coarsest level |

| BoomerAMG | AGMG |
|---|---|
| Aggressive coarsening (2 levels) | Double pair-wise agg. (default) |
| V(1,1) symGS | K-cycle (default) |
| Direct solve on coarsest level | forwardGS ↓,backwardGS ↑ (def.) |
| | MUMPS on coarsest level (def.) |

# 2D RESULTS: H-DEPENDENCY

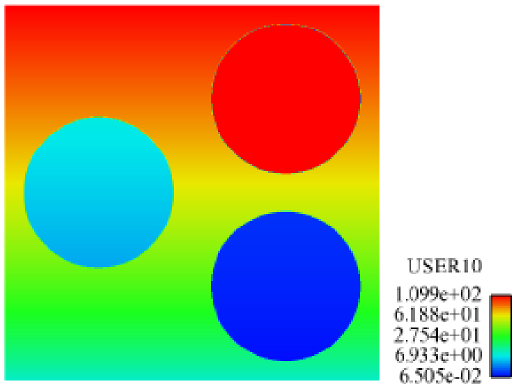
**H-dependent convergence did occur in isolated cases.**

**In one case for GAMG/BiCGstab:**




| 100x100 | 500x500 | 1000x1000 |
|---|---|---|
| 25 | 34 | 79 |

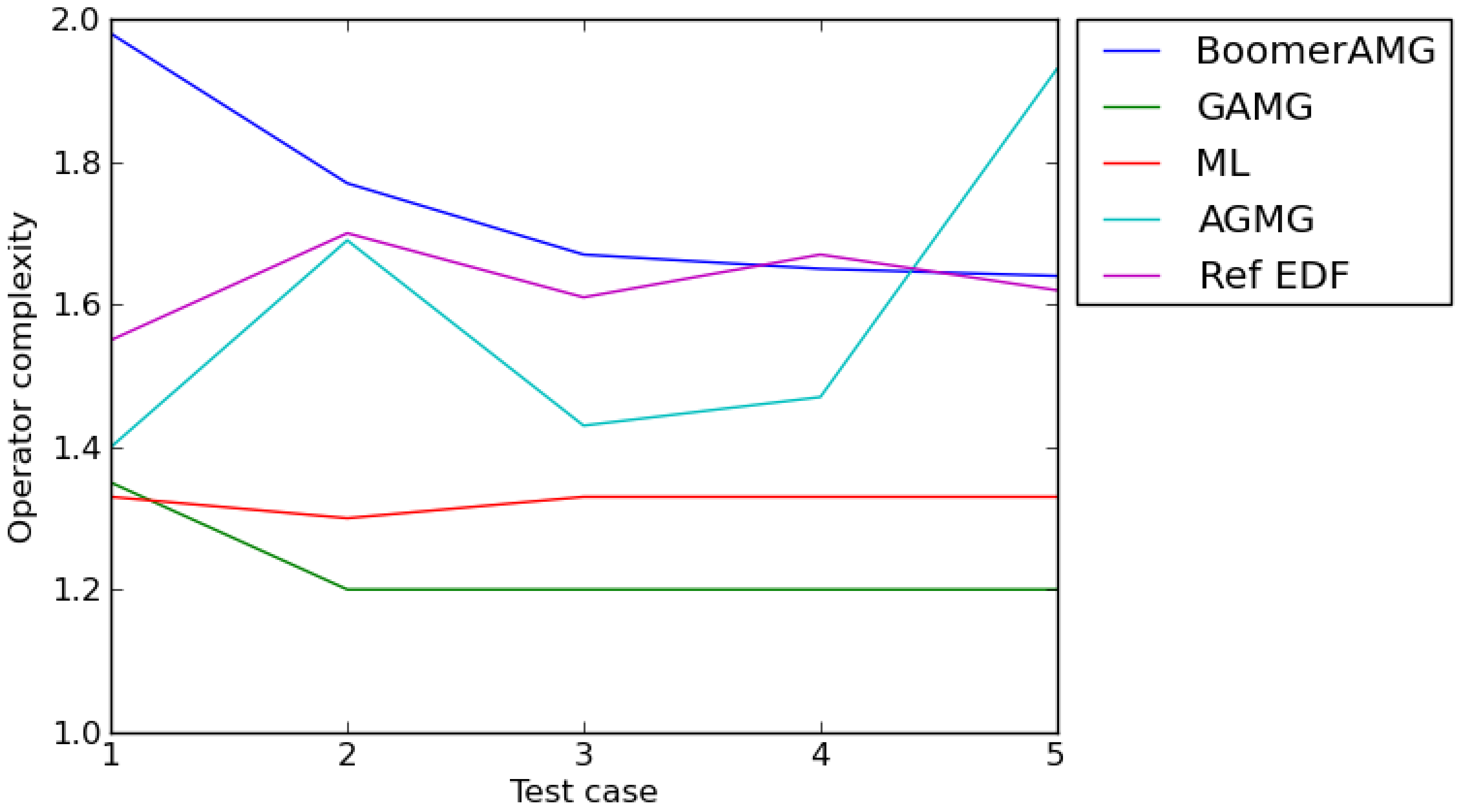
**In another for AGMG:**




(b) $C_2$

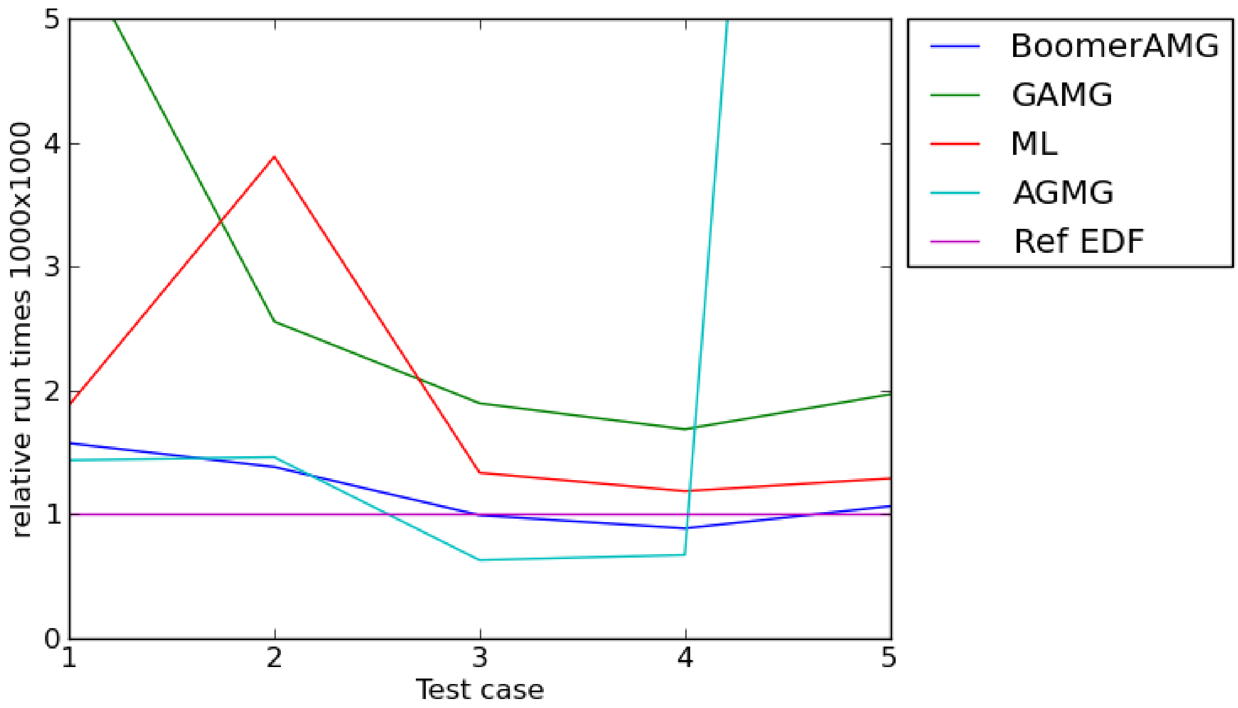
| 100x100 | 500x500 | 1000x1000 |
|---|---|---|
| 47 | 145 | 266 |

# 2D RESULTS: OPERATOR COMPLEXITY




From our point of view, no critical behavior.

# 2D RESULTS: WALL CLOCK TIME

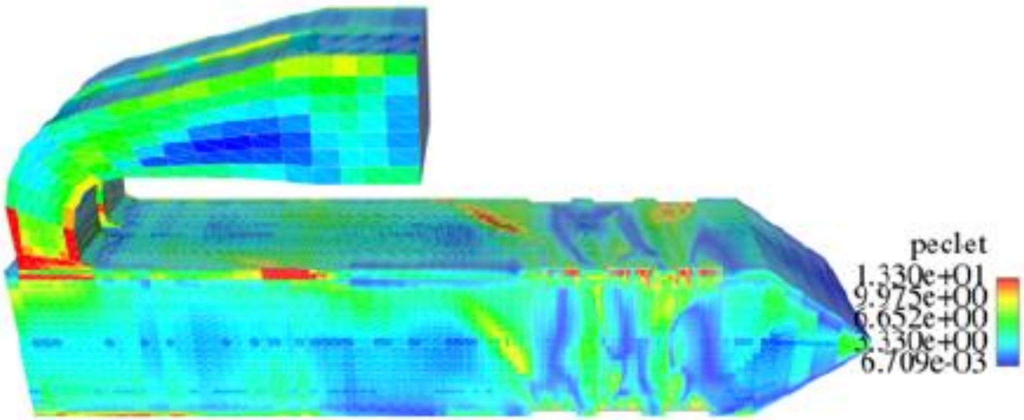



**ML and GAMG have the highest setup times.**

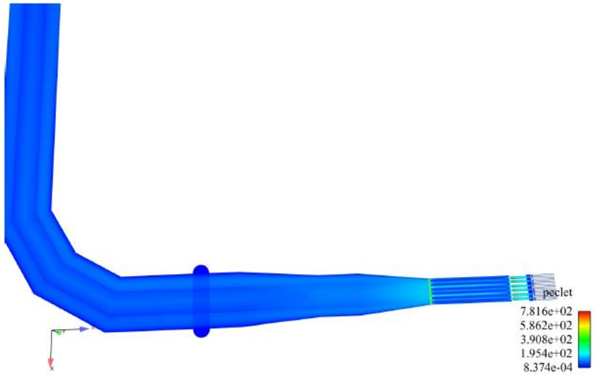
**The overall execution times are rather short, between 3.3 and 3.8 s for the reference solution.**

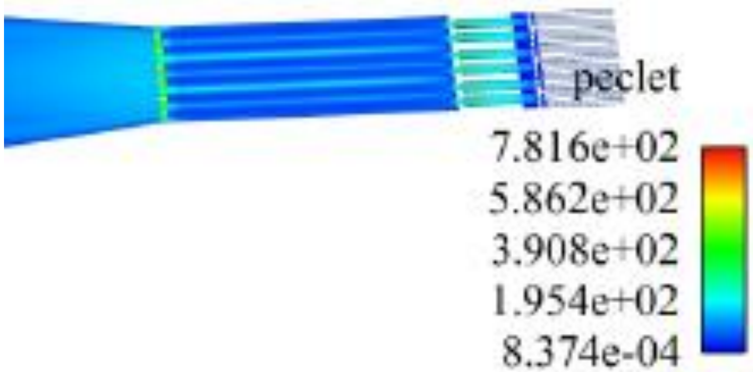
**When a method can be used as solver or as preconditioner, the shortest run time of the two options is taken into account.**

# 3D TEST CASES



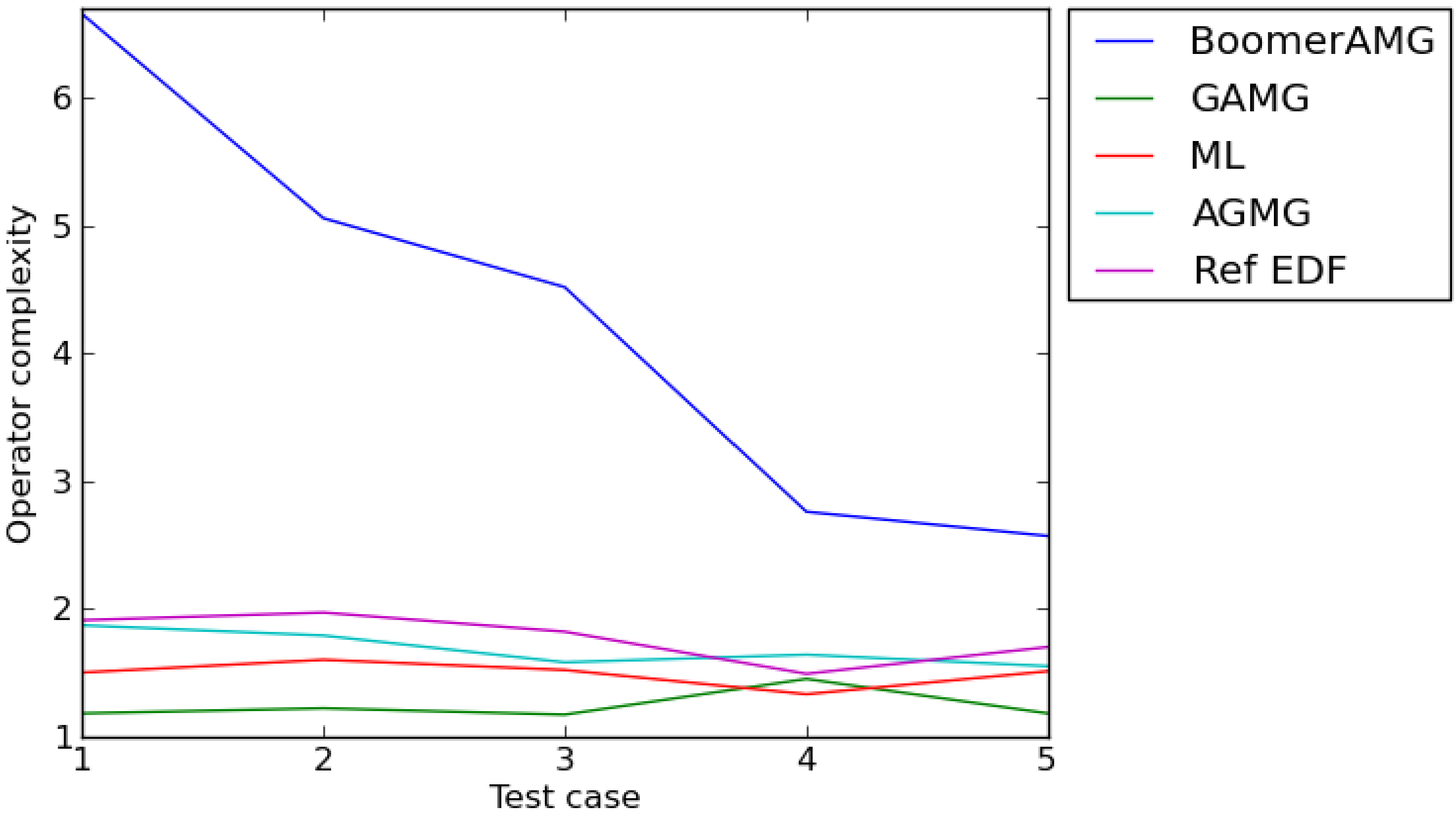

| No. | Reference | #Dof |
|---|---|---|
| 1 | ConvDiff1 | 462 786 |
| 2 | ConvDiff2 | 712 266 |
| 3 | ConvDiff3 | 10 196 476 |
| 4 | EllSum1 | 256 000 |
| 5 | EllSum2 | 587 596 |



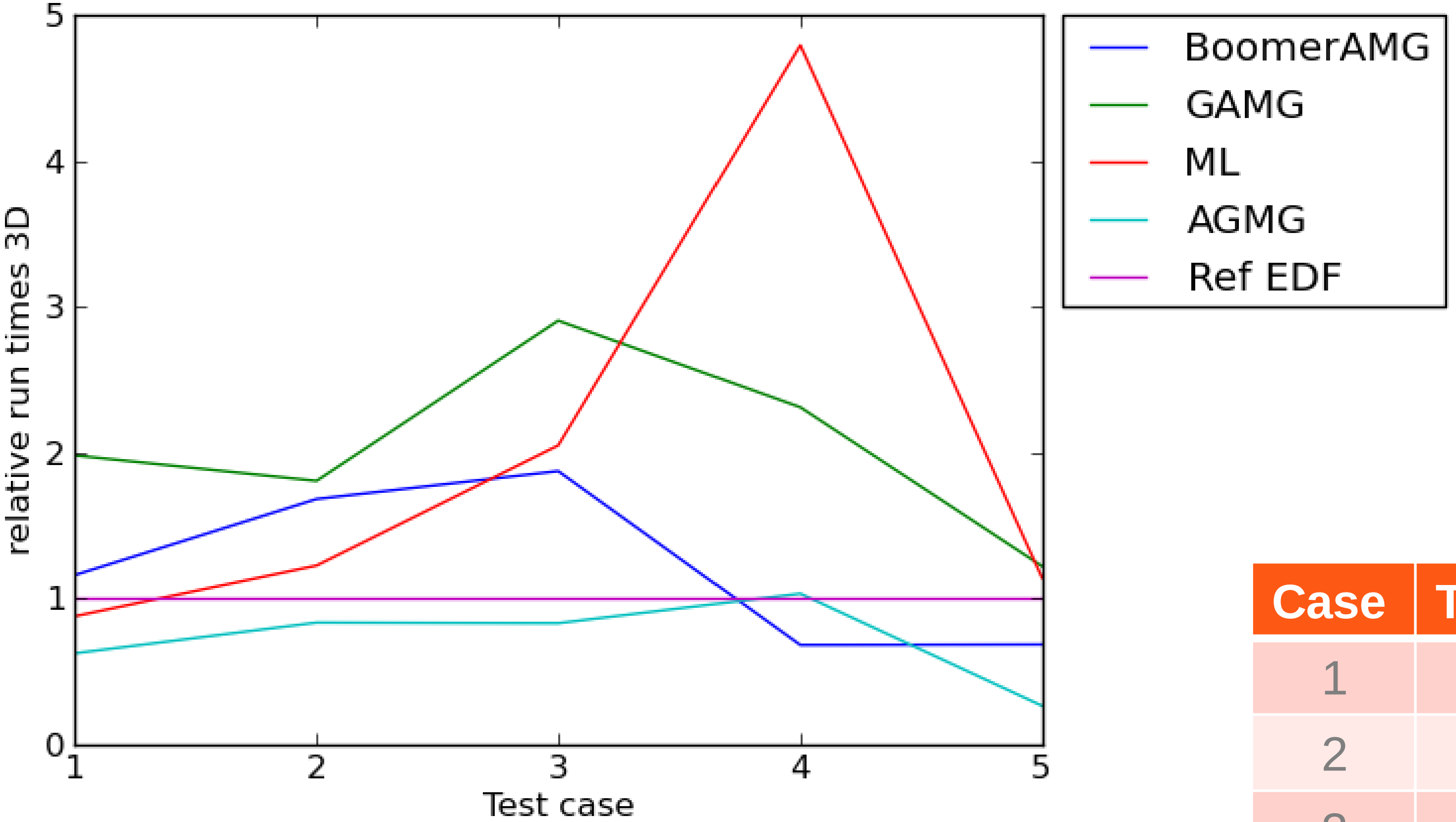

All convection-diffusion cases contain areas of dominant diffusion as well as areas of dominant convection.

# 3D RESULTS: OPERATOR COMPLEXITY




Convection-diffusion problems are obviously harder for BoomerAMG than non-sym. elliptic equations.

# 3D RUN TIME RESULTS




Once again, the absolute execution times were rather short.

| Case | Time [s] |
|---|---|
| 1 | 5.3 |
| 2 | 5.1 |
| 3 | 52.4 |
| 4 | 1.2 |
| 5 | 6.4 |

# CONCLUSIONS FOR THE NON-SYMMETRIC TESTS

- All methods/libraries converged on our test cases, despite the lack of symmetry.
- All four libraries are (highly) configurable. AGMG and BoomerAMG required the least user intervention.
- For non-initiated users, “discontinuous” consequences (OK $\Rightarrow$ fail) of some parameter choices are unacceptable. If the experts (i.e. the developers) do not know how to spot and how to deal with a problem at runtime, who can?
- On our test cases, our in-house methods are competitive.

# SUMMARY

**"Multigrid works, when you have Achi Brandt sitting next to you."**

**(Gene Golub)**

# SUMMARY

- When designed carefully, multilevel methods ARE ready for industry.
- However, not every multilevel method is automatically fast.
- In our applications, we have access to more than purely algebraic information. HMG and FV-AMG are successful examples of how to take problem specific information into account.
- Disadvantage of tightly integrated schemes: Changes to the discretization scheme imply adaptation of the linear solver.
- For our applications, the freely available MG solvers are applicable to non-symmetric, scalar problems.
- Open questions:
  - Fast and robust solvers for structural mechanics (model mix, Lagrange multipliers).
  - Fast solvers for compatible discrete operators in CFD.

# Thank you for your attention.