Massively parallel solving of the Pressure-Poisson equation on unstructured meshes

Mathias Malandain – Vincent Moureau
CORIA – CNRS/UMR 6614
Simulation of the PRECCINSTA burner

- PRECCINSTA burner:
  - Rectangular cuboid chamber: 110mm x 86mm x 86mm
  - Air-methane mixture, Re=45000

Simulation of PRECCINSTA

- Successive grid refinements lead from 1.7 million cells to 2.6 billion\textsuperscript{[1]}:

- Chosen among other possible solutions:
  
  - direct import of large meshes
  - reconstruction from partial meshes on the domain

\textsuperscript{[1]} Rivara, « Mesh refinement processes based on the generalized bisection of simplices », 1984
Simulation of PRECCINSTA (Large-eddy simulation)

- Cold flow:

  - PRECCINSTA burner
  - Large-Eddy Simulation with 2.6 billion tetrahedrons
  - Cell size of 100 microns
  - 12288 CPUs of Babel

V. Moureau, UMR6614 – CORIA
Simulation of PRECCINSTA (Large-eddy simulation)

- Cold flow:
Simulation of PRECCINSTA (Large-eddy simulation)

- Movie of a combustion:
Speed-up on PRECCINSTA (solver used: Deflated PCG)

Most of the computational time is spent in the Pressure-Poisson solver.
Plan

- Numerical schemes

- **Double Domain Decomposition** methodology

- « Standard » domain decomposition **deflation**

- Improvements:
  - **linear/quadratic deflation**: idea, implementation and results
  - **stabilization** of PCG algorithm
  - **multi-level deflation**: concept, implementation of three-level deflation

- Conclusion and **perspectives**
Numerical schemes

- **Temporal scheme**: fourth order time integration
  - Runge-Kutta 4 (stable on a centered space discretization, provided that a CFL condition is satisfied)
  - **TRK4** (RK4 with tunable diffusion, obtained by combination of RK4 and Lax-Wendroff-type schemes)
    - Runge-Kutta schemes known to be unstable but non-dissipative, and Lax-Wendroff scheme known to be stable but dissipative
    - Idea: *creating an affine combination of RK and LW second-order spatial derivatives*, and adjusting the coefficients to tune the stability and diffusion of the scheme
    - High CFL numbers for TRK4 schemes can make up for the additional computational costs involved
    - Work to be published by Moureau and coworkers
Numerical schemes

- **Spatial scheme**: centered finite volume method of fourth order
  - conservative: ideal for discontinuities arising in compressible flows (Variable Density Solver) for instance
  - well-adapted to unstructured meshes: RHS computed as a sum of contributions on edges or faces

- **Stabilization**: Cabot & Cook fourth-order artificial viscosity
  - Will approximate the cusp in the turbulent energy spectrum induced by fourth-order error
Discretization : DDD

- Unstructured 1D, 2D and 3D solver with different cell geometries afforded (e.g., tetrahedra and hexahedra in 3D).

- **Double Domain Decomposition** methodology :
Discretization : DDD

- To deal with this structure, internal communicators are added to the « usual » external communicators
Discretization : DDD

- Advantages of DDD :
  - reduces costs related to parallel use
  - massively parallel comp. (load balancing, local mesh refinement)
  - well-adapted to preconditioned/deflated solvers
Linear solvers

- **Deflated PCG**:
  - Preconditioner: inverse of the diagonal
  - Domain Decomposition deflation method on the cell groups

\[ E = W^T A W \ ; \ Q = W E^{-1} W^T \ ; \ P = I - AQ \]

- Quite a common use of deflation (introduced with a mathematical idea of using eigenvectors\(^1\), but widely used as a preconditioner on a coarse grid\(^2\))

- **Deflated BiCGStab(2)**:
  - Family of BiCGStab(L) introduced in 1993\(^3\) in order to overcome the BiCGStab2 algorithm\(^4\)

\(^1\) Nicolaides, « Deflation of Conjugate Gradients with Application to Boundary Value Problems », 1987
\(^2\) Vermolen, Vuik & Segal, « Deflation in preconditioned conjugate gradient methods for Finite Elements Problems », 2002
\(^3\) Sleijpen & Fokkema, « BiCGStab(L) for linear equations involving unsymmetric matrices with complex spectrum », 1993
\(^4\) Gutknecht, « Variants of BiCGStab for matrices with complex spectrum », 1991
Linear and quadratic deflation

- Idea: adding deflation vectors changes the solution of the deflated system
Linear and quadratic deflation: application (2D structured code)

- Expected solution
- First solution of DD deflation
- First solution of linear deflation
- First solution of quadratic deflation
Linear and quadratic deflation

- Very promising: number of iterations substantially decreased

- Numerical instabilities in the 3D unstructured solver Yales2 → divergences

- Attempts to stabilize the deflated algorithm (A-DEF2)
Improvement of the DPCG algorithm

- Stabilization of Deflated PCG thanks to a 2009 article from Tang and coworkers\[1\]

- Deflation, Domain Decomposition and Multigrid Methods are written as preconditioners

- Multiplicative combination: \( C_1 \) and \( C_2 \) two preconditioners, then
  \[
  x^{i+1/2} = x^i + C_1 (b - Ax^i) \quad \text{and} \quad x^{i+1} = x^{i+1/2} + C_2 (b - Ax^{i+1/2})
  \]
  give:
  \[
  x^{i+1} = x^i + (C_1 + C_2 - C_2 AC_1)(b - Ax^i)
  \]

- Applied to the « usual » inverse of the diagonal \( M^{-1} \) and the deflation matrix \( Q \) gives birth to the adapted deflations \( P_{A\text{-DEF}1} = M^{-1}P + Q ; P_{A\text{-DEF}2} = P^T M^{-1} + Q \)

- The \( A\text{-DEF}2 \), although being quite costly, is shown to be the most stable among nine different methods on a porous media problem and a bubbly flow problem

\[1\] Tang, Nabben, Vuik & Erlangga, « Comparison of Two-Level Preconditioners Derived from Deflation, Domain Decomposition and Multigrid Methods », 2009
Multi-level deflation

- 3-Level Deflated PCG:

- Inspired from the multigrid conception, with the benefits of deflation (for unstructured meshes)

  - Thanks to the METIS library, every cell group is split into «subgroups», thus providing an intermediate level between the fine mesh and the coarse mesh

  - The solver on the fine mesh uses deflation on the intermediate mesh, whose solver uses deflation on the coarse mesh

- Aims at reducing the time spent on the deflation on the cell groups

  - Size of cell groups to be adapted carefully, in order not to spend too much time in communications
Multi-level deflation

- Number of iterations of the deflated solver at each call, for identical cell group sizes (3D cartesian grid):

![Graph showing iterations of the deflated solver for each call by the fine solver.](image)
Conclusion and perspectives

- **Encouraging results** obtained on the multi-level deflation when applied to test-cases

- Evolution of the existing Conjugate Gradient solvers

- Exposed weaknesses on PCG solvers that are to be corrected

- **Next perspective**: recycling the residuals at each iteration

  - Sequence of linear systems $Ax_n=b_n$ to be solved ($n$ stands for a time step e.g.)
  - $b_n$ is a linear combination of the vectors $\{b_{n-1}, \ldots, b_{n-1}\}$ and an additional vector $\beta_n$
  - Projection technique: assured not to deteriorate the system to be solved (contrarily to the standard change $A(x_n-x_{n-1})=b_n-b_{n-1}$)

Thank you for your attention!