R-adaptation for unsteady compressible flow simulations in three dimensions

Luca Cirrottola, Algiane Froehly, Alberto Guardone, Giuseppe Quaranta, Barbara Re, Mario Ricchiuto

R-adaptation for unsteady compressible flow simulations in three dimensions

Luca Cirrottola[a*], **Algiane Froehly**[b], **Alberto Guardone**[c], **Giuseppe Quaranta**[d], **Barbara Re**[e], **Mario Ricchiuto**[f],

[a,b,f] Inria Bordeaux Sud-Ouest
Team CARDAMOM
F-33405 Talence, France
[a]e-mail: luca.cirrottola@inria.fr
[b]e-mail: algiane.froehly@inria.fr
[f]e-mail: mario.ricchiuto@inria.fr

[c,d] Department of Aerospace Science and Technology
Politecnico di Milano
20156 Milano, Italy
[c]e-mail: alberto.guardone@polimi.it
[d]e-mail: giuseppe.quaranta@polimi.it

[e] Institute of Mathematics
University of Zürich
8057 Zürich, Switzerland
e-mail: barbara.re@math.uzh.ch

ABSTRACT

We present an algorithm to perform PDE-based r-adaptation in three-dimensional numerical simulations of unsteady compressible flows on unstructured meshes. A Laplacian-based model for the moving mesh is used to follow the evolving shock-wave patterns in the fluid flow, while the finite volume ALE formulation of the flow solver is employed to implicitly perform a conservative remapping of the solution from the previous to the current mesh, at each time step of the simulation.

We show the application of this method to compressible flows on three-dimensional geometries. To this aim, an improved relaxation scheme has been developed in order to preserve the validity of the mesh throughout the time simulation in three dimensions, where the geometrical constraints typically restrict the allowable mesh motion.

1 Introduction

Mesh adaptation is widely used in fluid dynamics simulations to improve numerical accuracy and to capture relevant flow patterns. In this context, *r-adaptation*, i.e. mesh adaptation by node relocation with constant connectivity, offers the possibility to bring mesh adaptation capabilities into existing flow solvers with minimal intrusivity, as there is no need for adaptive data structures.

We consider partial differential equations (PDE) models for r-adaptation which can be obtained from the minimization of an adaptation functional. While most of these models are formulated for the parametric coordinates $\boldsymbol{\xi}(\mathbf{x})$ in the physical domain $\Omega_{\mathbf{x}}$, we follow the approach shown in [2, 3, 1] to directly formulate the problem for the mesh coordinates $\mathbf{x}(\boldsymbol{\xi})$ in the reference domain $\Omega_{\boldsymbol{\xi}}$, so that the resulting problem becomes similar to Lagrangian methods in computational mechanics, thus easier to solve.

2 Numerical model for mesh movement

The r-adaptation method presented in this work is implemented in the `Fmg` library, using the data structures provided by the more general `Mmg` [4] remeshing library. In the following, we briefly describe the PDE model and the relaxation method to enforce mesh validity.

2.1 Laplacian model in the reference domain

We look for a mapping $\mathbf{x} : \Omega_{\boldsymbol{\xi}} \rightarrow \Omega_{\mathbf{x}}$ from the reference domain $\Omega_{\boldsymbol{\xi}}$ (the original mesh) to the computational domain $\Omega_{\mathbf{x}}$ (the adapted mesh), through the solution of a variable-diffusion Laplace equation

$$\boldsymbol{\nabla}_{\boldsymbol{\xi}} \cdot (\omega(\mathbf{x}) \boldsymbol{\nabla}_{\boldsymbol{\xi}} \mathbf{x}) = \mathbf{0} \qquad \text{in } \Omega_{\boldsymbol{\xi}} \tag{1}$$

The boundary is split as $\partial\Omega_{\boldsymbol{\xi}} = \Gamma^D_{\boldsymbol{\xi}} \cup \Gamma^S_{\boldsymbol{\xi}}$ so that Dirichlet conditions are imposed on $\Gamma^D_{\boldsymbol{\xi}}$ and slip conditions are imposed on $\Gamma^S_{\boldsymbol{\xi}}$

$$\mathbf{x} = \boldsymbol{\xi} \quad \text{on } \Gamma^D_{\boldsymbol{\xi}}, \qquad \hat{\mathbf{n}} \cdot (\mathbf{x} - \boldsymbol{\xi}) = 0 \quad \text{on } \Gamma^S_{\boldsymbol{\xi}} \tag{2}$$

The weak formulation of the above PDE problem is discretized with linear finite elements. For each space direction, the above equations are uncoupled and nonlinear through the monitor function $\omega(\mathbf{x})$ built from a scalar fluid flow solution $\rho(\mathbf{x})$ as

$$\omega(\mathbf{x}) = \sqrt{1 + \alpha ||\boldsymbol{\nabla}_{\boldsymbol{\xi}} \rho(\mathbf{x})||^2_{\gamma_\alpha} + \beta ||\mathbf{H}_{\boldsymbol{\xi}}(\rho)(\mathbf{x})||^2_{\gamma_\beta} + \tau ||\rho||^2_{\gamma_\tau}} \tag{3}$$

where $\boldsymbol{\nabla}_{\boldsymbol{\xi}}$ and $\mathbf{H}_{\boldsymbol{\xi}}$ denote the gradient and Hessian computed on the reference domain $\Omega_{\boldsymbol{\xi}}$; the $||\cdot||_\gamma$ norm is defined as a Frobenius norm, normalized by γ-times its maximum and upper-limited at 1. The nonlinear system of equations resulting from the above model is re-expressed in incremental form and solved through Jacobi iterations, thus producing a sequence of nodes displacements $\boldsymbol{\delta}_i^{[k+1]} = \mathbf{x}_i^{[k+1]} - \boldsymbol{\xi}_i$.

2.2 Enforcing mesh validity through nodewise relaxation

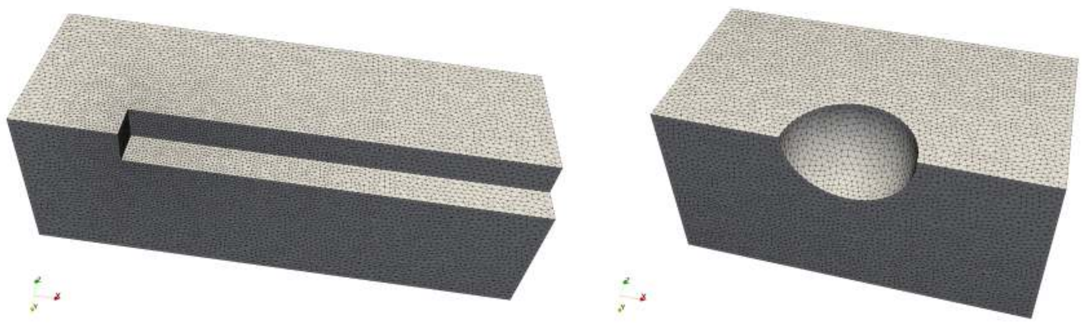
The Laplacian model in the reference domain does not guarantee that the Jacobian of the mapping is strictly positive everywhere, thus leading to the occurrence of tangled (invalid) mesh elements. While experience shows that in two dimensions it is possible to avoid invalid elements by carefully tuning the monitor function $\omega(\mathbf{x})$, this is generally not the case in three dimensions, where tangling is much easier. To this aim, at each iteration the displacement of each node is relaxed by a factor $\mu_i^{[k+1]}$ before updating one-by-one each node position as

$$\mathbf{x}_i^{[k+1]} = \boldsymbol{\xi}_i + \mu_i^{[k+1]} \boldsymbol{\delta}_i^{[k+1]} \tag{4}$$

The relaxation factor is chosen in order to guarantee a minimum volume for each virtual tetrahedron produced by the displacement of node i in its ball $\mathcal{B}_i^{[k,k+1]}$, whose coordinates are evaluated at the previous mesh configuration k for nodes not already updated (indices in $[i+1, n]$), and at the next mesh configuration $k+1$ for nodes already updated (indices in $[1, i-1]$). This check allows to enforce the validity of every intermediate mesh configuration, effectively preventing the occurrence of invalid elements.

3 Results

For the application to unsteady compressible flow simulation, we employ the `Flowmesh` [5] solver for the Euler equations for inviscid flows in Arbitrary Lagrangian-Eulerian formulations. We select our test cases in order to exhibit unsteady flows with shocks on singular and curved geometries (figure 1).

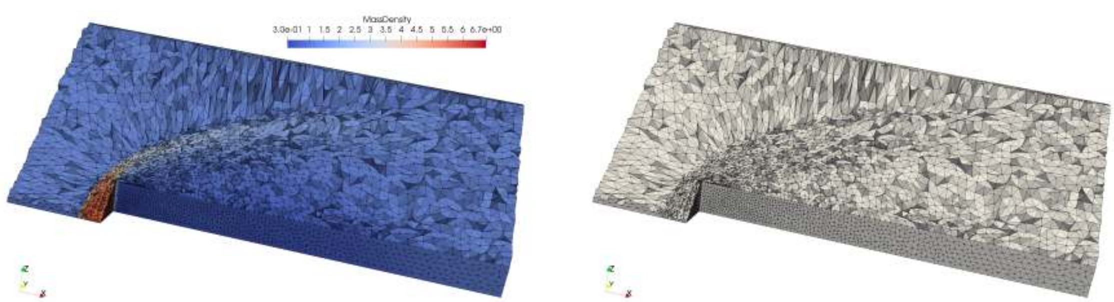


(a) Forward facing step case. (b) Sphere case.

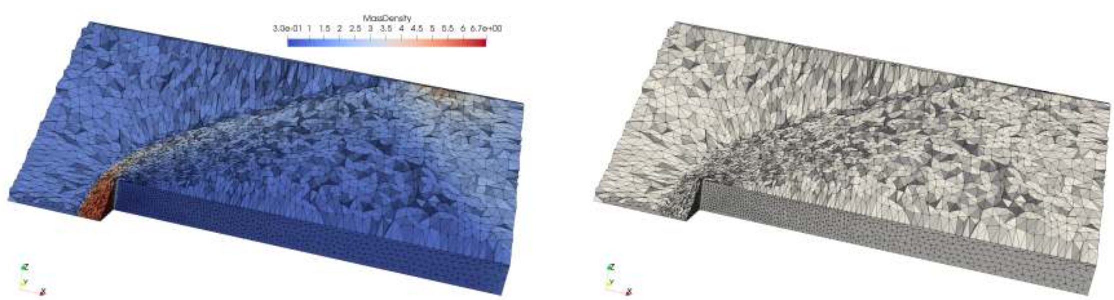
Figure 1: Initial meshes.

3.1 Three-dimensional forward facing step

We propose a three-dimensional extension of the classical supersonic forward facing step. The impulsive start of a Mach 3 flow in a 1 length unit wide and 3 length units long wind tunnel, with a 0.2 length unit high step located at 0.6 length units from the inlet. Adaptation is performed on the mass density (figure 2).



(a) Volumic cut and mass density at $t = 70$. (b) Volumic cut at $t = 70$.

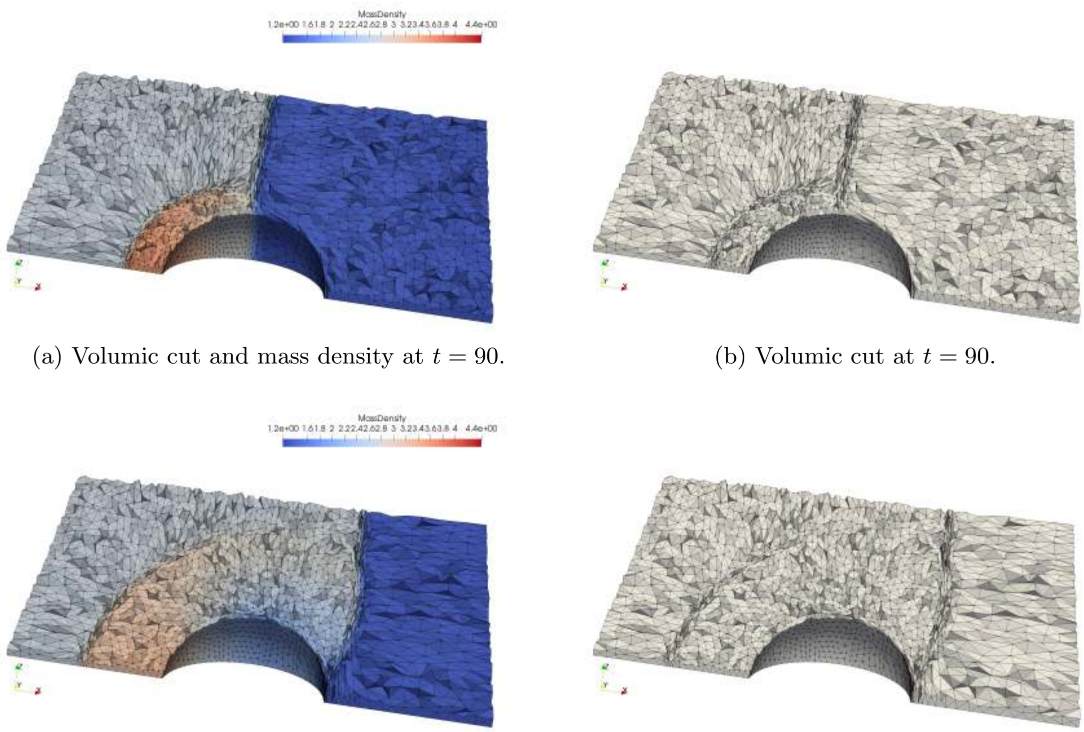


(c) Volumic cut and mass density at $t = 200$. (d) Volumic cut at $t = 200$.

Figure 2: Adapted meshes and solution for the 3D forward facing step case.

3.2 Shock–sphere interaction

A planar shock moving at $M_s = 1.5$ impinges on a sphere. Adaptation is performed on the mass density (figure 3).



(a) Volumic cut and mass density at $t = 90$.

(b) Volumic cut at $t = 90$.

(c) Volumic cut and mass density at $t = 180$.

(d) Volumic cut at $t = 180$.

Figure 3: Adapted meshes and solution for the shock-sphere interaction case.

4 Conclusions

The proposed approach produces sufficiently adapted meshes in as few as ten Jacobi iterations per time step. The nodewise relaxation effectively prevents the occurrence of invalid elements throughout the unsteady simulations, also with corners and curved surfaces, and its dependence from the node ordering doesn't appear to spoil the adaptation pattern in any of our tests. The Laplacian models excessively pulls nodes towards non-convex boundaries, and further work is in progress to mitigate this problem.

Acknowledgements

We would like to recognize the fundamental contribution Cécile Dobrzynski brought to this work.

REFERENCES

[1] Luca Arpaia and Mario Ricchiuto, *r-adaptation for shallow water flows: conservation, well balancedness, efficiency*, Research Report RR-8956, Inria Bordeaux Sud-Ouest, 2016.

[2] Hector D. Ceniceros and Thomas Y. Hou, *An efficient dynamically adaptive mesh for potentially singular solutions*, Journal of Computational Physics **172** (2001), no. 2, 609 – 639.

[3] Guoxian Chen, Huazhong Tang, and Pingwen Zhang, *Second-order accurate godunov scheme for multicomponent flows on moving triangular meshes*, Journal of Scientific Computing **34** (2008), no. 1, 64–86.

[4] C. Dapogny, C. Dobrzynski, and P. Frey, *Three-dimensional adaptive domain remeshing, implicit domain meshing, and applications to free and moving boundary problems*, Journal of Computational Physics **262** (2014), 358 – 378.

[5] B. Re, C. Dobrzynski, and A. Guardone, *An interpolation-free ale scheme for unsteady inviscid flows computations with large boundary displacements over three-dimensional adaptive grids*, Journal of Computational Physics **340** (2017), 26 – 54.