Moving-mesh hydrodynamics with the AREPO code

Volker Springel[1,2]

[1]Heidelberg Institute for Theoretical Studies, Schloss-Wolfsbrunnenweg 35,
69118 Heidelberg, Germany
[2]Zentrum für Astronomie, Universität Heidelberg, Mönchhofstr. 12-14,
69120 Heidelberg, Germany
email: `volker.springel@h-its.org`

Abstract. At present, hydrodynamical simulations in computational star formation are either carried out with Eulerian mesh-based approaches or with the Lagrangian smoothed particle hydrodynamics (SPH) technique. Both methods differ in their strengths and weaknesses, as well as in their error properties. It would be highly desirable to find an intermediate discretization scheme that combines the accuracy advantage of mesh-based methods with the automatic adaptivity and Galilean invariance of SPH. Here we briefly describe the novel AREPO code which achieves these goals based on a moving unstructured mesh defined by the Voronoi tessellation of a set of discrete points. The mesh is used to solve the hyperbolic conservation laws of ideal hydrodynamics with a finite volume approach, based on a second-order unsplit Godunov scheme with an exact Riemann solver. A particularly powerful feature is that the mesh-generating points can in principle be moved arbitrarily. If they are given the velocity of the local flow, an accurate Lagrangian formulation of continuum hydrodynamics is obtained that features a very low numerical diffusivity and is free of mesh distortion problems. If the points are kept fixed, the scheme is equivalent to a Eulerian code on a structured mesh. The new AREPO code appears especially well suited for problems such as gravitational fragmentation or compressible turbulence.

Keywords. hydrodynamics, methods: numerical, radiative transfer

---

1. Introduction

In astrophysics, a variety of fundamentally quite different numerical methods for hydrodynamic simulations are in use, the most prominent ones are SPH (e.g. Monaghan 1992) and Eulerian mesh-based hydrodynamics (e.g. Stone *et al.* 2008) with (optional) adaptive mesh refinement (AMR). However, it has become clear over recent years that both SPH and AMR suffer from fundamental problems that make them inaccurate in certain regimes. Indeed, these methods sometimes yield conflicting results even for basic calculations that only consider non-radiative hydrodynamics (e.g. Frenk *et al.* 1999, Agertz *et al.* 2007, Tasker *et al.* 2008, Mitchell et al. 2009). SPH codes have comparatively poor shock resolution, and offer only low-order accuracy for the treatment of contact discontinuities. Worse, they appear to suppress fluid instabilities under certain conditions, as a result of a spurious surface tension and inaccurate gradient estimates across density jumps. On the other hand, Eulerian codes are not free of problems either. They do not produce Galilean-invariant results, which can make the results sensitive to the presence of large bulk velocities. Another concern lies in the comparatively strong mixing inherent in multi-dimensional Eulerian hydrodynamics. This provides for an implicit source of entropy, with unclear consequences (e.g. Wadsley, Veeravalli & Couchman 2008).

There is hence substantial motivation to search for new hydrodynamical methods that improve on these weaknesses. We would like to retain the accuracy of mesh-based

203

hydrodynamical methods while at the same time outfit them with the Galilean-invariance, geometric flexibility, and automatic adaptivity that is characteristic of SPH. The principal idea for achieving such a synthesis is to allow the mesh to move with the flow itself. This is an obvious and old idea, but one fraught with many practical difficulties that have so far prevented widespread use of any of the few past attempts to introduce moving-mesh methods in astrophysics (e.g. Gnedin 1995, Pen 1998).

2. The AREPO code

In our new AREPO code (see Springel 2010 for a detailed description), we introduce a new formulation of continuum hydrodynamics based on an unstructured mesh. The mesh is defined as the Voronoi tessellation of a set of discrete mesh-generating points, which are in principle allowed to move freely. For the given set of points, the Voronoi tessellation of space consists of non-overlapping cells around each of the sites such that each cell contains the region of space closer to it than any of the other sites. In practice, we construct the Voronoi mesh in terms of the Delaunay tessellation, which is the topological dual of the Voronoi diagram and can be more easily calculated with fast geometric algorithms.

The Voronoi cells can be used as control volumes for a finite-volume formulation of hydrodynamics, using the same principal ideas for reconstruction, evolution and averaging (REA) steps that are commonly employed in many Eulerian techniques. However, it is possible to consistently include the mesh motion in the formulation of the numerical steps, allowing the REA-scheme to become Galilean-invariant. Even more importantly, due to the mathematical properties of the Voronoi tessellation, the mesh continuously deforms and changes its topology as a result of the point motion, without ever leading to the dreaded mesh-tangling effects that are the curse of traditional moving mesh methods.

In our approach, each computational cell carries as conserved fluid variables the mass, momentum and total energy, while the primitive variables are determined through the Voronoi volume of a cell. We then estimate spatial gradients for the unstructured mesh, apply slope limiting techniques to them, and finally define a piece-wise linear reconstruction for all primitive fluid variables. Appropriate spatial and temporal extrapolation steps are applied to estimate the fluid states at both sides of each mesh face. These are fed to a Riemann solver, yielding numerical fluxes that update in a detailed balance the conserved quantities of the cells. The overall method is a second-order accurate scheme in space and time, and closely corresponds to a MUSCL approach.

Self-gravity is calculated in the code with a hierarchical multipole expansion (a tree algorithm), which we have ported from the GADGET-3 code. Also, a TreePM extension for a more efficient long-range gravity calculation is available. The AREPO code is fully parallelized for distributed memory machines, allows for individual and adaptive timesteps, and can also treat additional collisionless particle components.

3. Example calculations

In Springel (2010), a number of test problems are discussed that demonstrate that the new scheme achieves high accuracy in the treatment of shocks, shear waves, and fluid instabilities, making it an attractive alternative to currently employed SPH and AMR codes. To illustrate a few principal features of the code, we here briefly discuss the Rayleigh-Taylor (RT) instability as a representative example for the behavior of the method in multi-dimensional flow.

The RT instability can arise in stratified layers of gas in an external gravitational field. If higher density gas lies on top of low-density gas, the stratification is unstable to
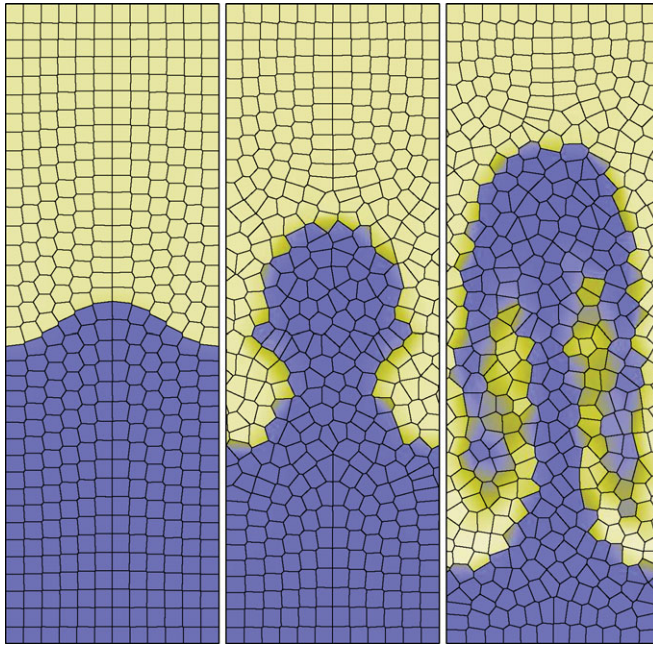
Figure 1. Rayleigh-Taylor instability calculated at low resolution with the moving-mesh approach. A denser fluid lies above a less dense fluid in an external gravitational field. The hydrostatic equilibrium of the initial state is buoyantly instable. The three frames show the time evolution of the density field of the system at times $t = 5.0$, 10.0, and 15.0, after a single mode has been perturbed to trigger the stability, as described in the text.

buoyancy forces, and characteristic finger-like perturbations grow that will mix the fluids with time. To illustrate the motion of the mesh in our new code, we show in Figure 1 the evolution of a single Rayleigh-Taylor mode, calculated at the low resolution of 12×36 cells. The simulation domain is two-dimensional (extension $[0.5, 1.5]$), and has periodic boundaries on the left and right and solid walls at the bottom and top. There is an external gravitational field with acceleration $g = -0.1$, and the bottom and top halves of the box are filled with gas of density $\rho = 1$ and $\rho = 2$, respectively. The initial hydrostatic pressure profile is of the form $P(y) = P_0 + (y - 0.75)\,g\,\rho(y)$ with $P_0 = 2.5$ and $\gamma = 1.4$. To seed the perturbation, one mode is excited with a small velocity perturbation of the form $v_y(x, y) = w_0[1 - \cos(4\pi x)][1 - \cos(4\pi y/3)]$, where $w_0 = 0.0025$.

As can be clearly seen in the time evolution shown in Figure 1, the Rayleigh-Taylor instability is captured well by the moving-mesh method even at this low resolution. Note that the sharp boundary between the phases can be maintained for relatively long time during the early evolution of the instability, simply because the contact discontinuity is not smeared out as it bends, thanks to the mesh's ability to follow this motion in an approximately Lagrangian fashion. A stationary mesh on the other hand would automatically wash out the boundary due to advection errors.

This important improvement becomes clearer in Figure 2, where we compare a high-resolution version of the RT instability between the moving-mesh approach and the same calculation carried out with a stationary Cartesian mesh. Here 1024×1024 cells have been used in the unit domain, and the instability was triggered by adding small random noise to the velocity field everywhere. While the instability shows a similar overall growth rate in both cases, there are also striking differences. The calculation with the fixed mesh produces a lot of intermediate density values due to the strong mixing of the phases
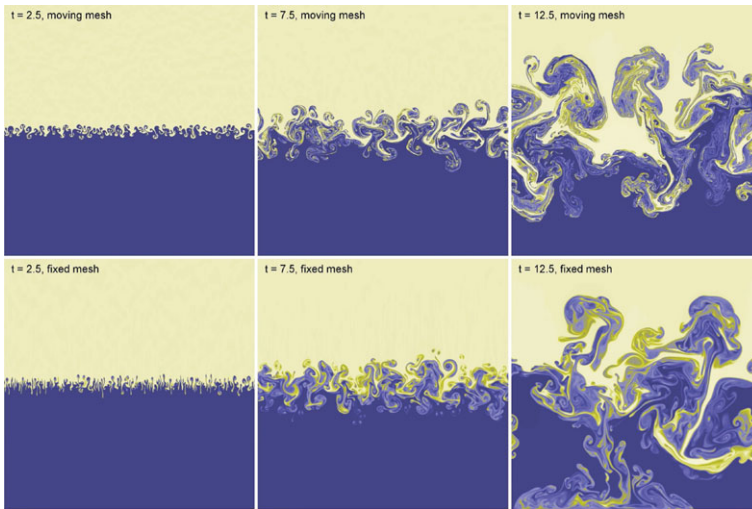
Figure 2. Rayleigh-Taylor instability calculated at high resolution with 1024×1024 points in the unit domain. The instability is here seeded by small random noise added to the velocity field. The top and bottom rows compare the time evolution for calculations with a moving and a stationary mesh, respectively.

on small scales, whereas the moving mesh approach maintains finely stratified regions where different layers of the fluid phases have been folded over each other. The contact discontinuities between these layers can be kept sharp by the code even when they are moving relative to the rest-frame of the box. In the early phase of the growth, it also appears as if small-scale RT fingers grow somewhat too quickly in the Eulerian case as a result of grid alignment effects.

4. Outlook

With the addition of an extensive chemical cooling network, the AREPO code is presently already employed to study formation of the first stars (Greif *et al.*, in preparation). The new code should be especially well suited for problems of gravitational fragmentation and for hydrodynamical turbulence. We presently also work on a radiative transfer solver that is directly operating on the Voronoi mesh (Petkova *et al.*, in preparation), and think about a potential implementation of magnetohydrodynamics in AREPO. With these additions, the code should become even more attractive for problems of computational star formation.

References

Agertz, O., Moore, B., Stadel, J., Potter, D., & Miniati, F., 2007, *MNRAS*, 380, 963
Frenk, C. S., *et al.*, 1999, *ApJ*, 525, 554
Gnedin, N. Y., 1995, *ApJS*, 97, 231
Mitchell, N. L., *et al.*, 2009, *MNRAS*, 395, 180
Monaghan, J. J., 1992, *ARAA*, 30, 543
Pen, U. L., 1998, *ApJS*, 115, 19
Springel, V., 2010, *MNRAS*, 401, 791
Stone, J. M., Gardiner, T. A., Teuben, P., Hawley, J. F., & Simon J. B., 2008, *ApJS*, 178, 137
Tasker, E. J., etal., 2008, *MNRAS*, 390, 1267
Wadsley, J. W., Veeravalli, G., & Couchman, H. M. P., 2008, *MNRAS*, 387, 427