John von Neumann Institute for Computing

Massively Parallel Simulations of Solar Flares and Plasma Turbulence

Lukas Arnold, Christoph Beetz, Jürgen Dreher,
Holger Homann, Christian Schwarz, Rainer Grauer

published in

Parallel Computing: Architectures, Algorithms and Applications ,
C. Bischof, M. Bücker, P. Gibbon, G.R. Joubert, T. Lippert, B. Mohr,
F. Peters (Eds.),
John von Neumann Institute for Computing, Jülich,
NIC Series, Vol. **38**, ISBN 978-3-9810843-4-4, pp. 467-474, 2007.
Reprinted in: Advances in Parallel Computing, Volume **15**,
ISSN 0927-5452, ISBN 978-1-58603-796-3 (IOS Press), 2008.

Massively Parallel Simulations of Solar Flares and Plasma Turbulence

**Lukas Arnold, Christoph Beetz, Jürgen Dreher,
Holger Homann, Christian Schwarz, and Rainer Grauer**

Institute for Theoretical Physics I
Ruhr-University Bochum, Germany
E-mail: grauer@tp1.rub.de

Some of the outstanding problems in space- and astrophysical plasmasystems include solar flares and hydro- or magnetohydrodynamic turbulence (e.g. in the interstellar medium). Both fields demand for high resolution and thus numerical simulations need an efficient parallel implementation. We will describe the physics behind these problems and present the numerical frameworks for solving these problems on massive parallel computers.

1 Introduction

In this paper, we will describe numerical simulations of fundamental plasma phenomena like the evolution of solar flares and Lagrangian statistics of compressible and incompressible turbulent flows. In the following, we will first describe the framework *racoon* which is an adaptive mesh refinement framework for hyperbolic conservation laws. Simulations of solar flares and compressible turbulence are performed utilizing this framework. The incompressible turbulence simulations are based on the spectral solver LATU. After describing this solver and discussing performing issues for both *racoon* and LATU, we will present physical results obtained from the simulations.

2 The Framework *racoon*

All simulations using finite volume and finite differences are performed using the framework *racoon*[1]. *racoon* is a computational framework for the parallel, mesh-adaptive solution[2] of systems of hyperbolic conservation laws like the time-dependent Euler equations in compressible gas dynamics or Magneto-Hydrodynamics (MHD) and similar models in plasma physics. Local mesh refinement is realized by the recursive bisection of grid blocks along each spatial dimension, implemented numerical schemes include standard finite-differences as well as shock-capturing central schemes[3], both in connection with Runge-Kutta type integrators. Parallel execution is achieved through a configurable hybrid of POSIX-multithreading and MPI distribution with dynamic load balancing based on space-filling Hilbert curves[4] (see Fig. 1).

racoon also has the ability to advect tracer particles with the flow using the same parallelisation strategy as for the blocks. The main numerical work is spent in the interpolation routines from cell values to the actual particle positions.

Benchmarks on IBM p690 machines with 32 CPUs show that the hybrid concept in fact results in performance gain over a pure MPI parallelization, which, however requires a careful optimization of the multi-threaded implementation.

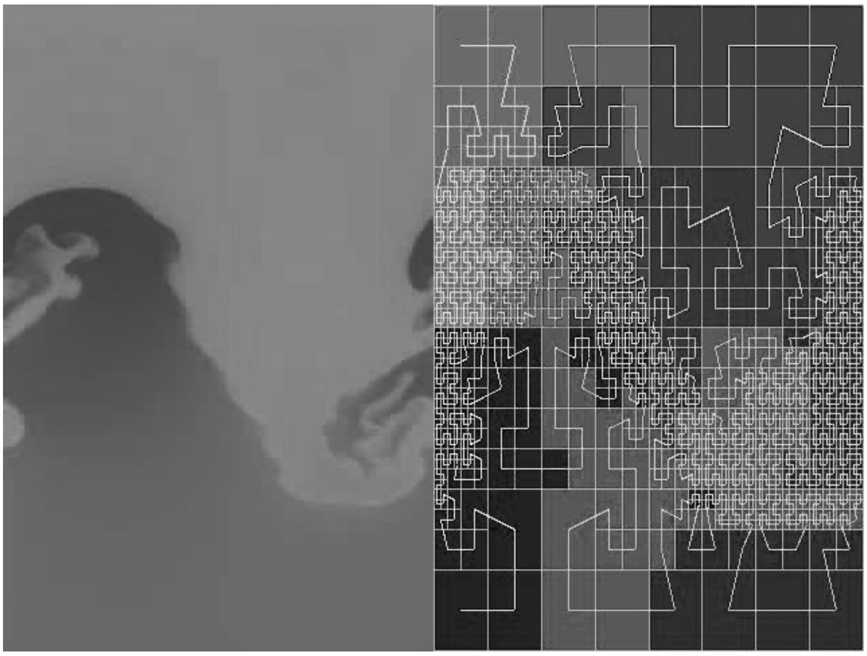

Figure 1. Simulation of a Rayleigh-Taylor instability. Load balancing is based on a Hilbert curve distribution (right side)

A key issue here is the efficient use of the CPU cache, which in the first place can be naturally obtained in AMR by the use of small grid block sizes that fit well into the cache. In addition, each thread in the current implementation creates its own effective data subspace consisting of a fixed subset of grid blocks and the block connectivity information, all allocated in the thread itself in order to achieve small CPU-memory distances in NUMA architectures. Block assignment to threads is based on the same space-filling curve algorithm that determines the distribution among processes, and which thereby tends to minimize not only inter-process communication but also inter-thread memory accesses with potential cache conflicts. To finally achieve the desired gain from multi-threading, the affinity between tasks and CPUs must be enforced manually by binding the working threads to individual CPUs.

For the MPI part of the communication, it turned out that the creation of fewer messages of moderate size (1 MB and below) by collecting the small inter-block messages which are addressed to the same target processor is favourable compared to mapping the typically small messages (one to few kB) between blocks directly to MPI messages, despite the fact that all MPI traffic is channelled through one thread in the message collection method. Here, the (de-) serialization of compound MPI messages can occur concurrently by many threads. These results indicate that the concurrent access to the MPI layer for the completion of many small-sized messages, even with multi-thread abilities, should be used carefully with respect to the overall performance. In the end, the hybrid concept proved to work satisfactory and resulted in floating point performances in the range of 7-10% of

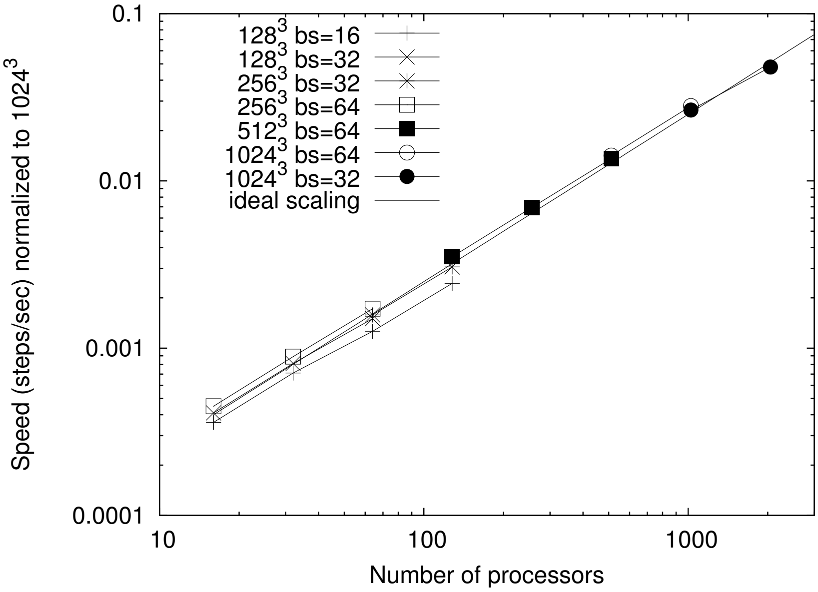

Figure 2. Strong scaling for different blocksizes (bs) varying from 16^3 to 64^3.

the theoretical peak performance on 64 processors for the described application. Naturally, there is still some room for further improvement, for example in connection with automatic estimates for the size of MPI compound messages. For practical use, further development of high-level interfaces for the control of task and memory affinity on high performance computers would be helpful, as the method of explicit CPU binding that was chosen here has the potential to conflict with the job dispatcher and load distribution algorithms in larger settings. One interesting initiative for IBM's platform is the VSRAC interface project (www.redbooks.ibm.com/redpapers/pdfs/redp3932.pdf), that might be extended in the near future to allow a thread-level control in addition to its current process-level control (see also R. Rabenseifer[5] for hybrid parallel programming).

On very massive parallel machines like the IBM BlueGene, a MPI only version is used. Scaling tests on the BlueGene JUBL at the FZ Jülich reveal linear scaling up to 2048 processors (see Fig. 2).

3 LATU: An Incompressible MHD Spectral Solver with Passive Tracer Particles

The numerical simulations of incompressible turbulence are performed using a pseudo-spectral code. The underlying equations are treated in Fourier-space, while convolutions arising from non-linear terms are computed in real space. A Fast-Fourier-Transformation (FFT) is used to switch between this two spaces. This scheme is very accurate and produces negligible numerical dissipation. The code is written in C++ and parallelizes efficiently. The time scheme is a Runge-Kutta third order. The inter-process communication uses the

Message Passing Interface (MPI) and the FFT are performed using the portable FFTW library or the San Diego P3DFFT routines using the IBM ESSL library. Simulations with 1024^3 collocation points were performed using up to 512 CPUs on the IBM p690 machine of the John von Neumann-Institut in Jülich. Results of preliminary scaling tests on BlueGene using up to 16384 processors and the P3DFFT routines are depicted in Fig. 3.

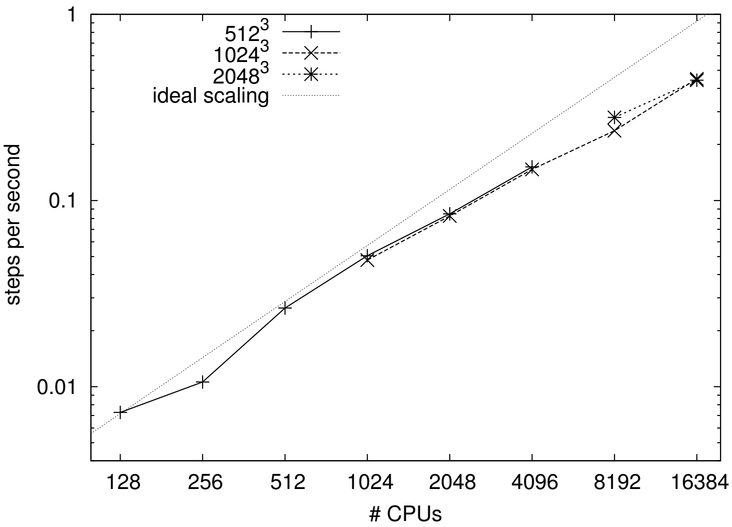

Figure 3. Mixture of strong and weak scaling of the LaTu code on BlueGene.

The implementation of the passive tracer particles is also done in a parallel way. This is necessary, because a large number of particles has to be integrated in order to sample the considered volume homogeneously and obtain reliable statistical results. We performed simulations with up to 10^7 particles on the IBM p690 machine.

The crucial point is the interpolation scheme needed in order to obtain the velocity field at the particle positions from the numerical grid. The code uses a tri-cubic interpolation scheme which on the one hand provides a high degree of accuracy and on the other hand parallelizes efficiently. Comparisons of the numerical results to simulations with a trilinear scheme are reported in[6].

4 FlareLab and Solar Flares

Observations of various phenomena at the solar surface and atmosphere – the so called solar transition region and solar corona – exist since a long time. Although successful work has been done in this time, mostly to explain and understand a special phenomenon, as yet there is no fundamental comprehension. These are for example the solar flares, the coronal mass ejections or the long living filaments and prominences. The connecting part in all these structures is the magnetic field. In these regions the magnetic forces and energies dominate the gravitational and thermal ones. This puts the investigation focus on magnetic structures and its evolution.

The *FlareLab* project is intended to simulate solar flares. It splits in two parts, an experimental and a numerical. The experiment is basically a plasma arc created by a gas discharge in an arcade like magnetic field, whereas the setup is orientated on the previous work by[7,8]. The evolution of this current arc and the structure formation are the key aspects of the experiment. Accompanying numerical simulations on the one hand are performed to analyze the experiment and to create a link to solar conditions. Numerical experiments on the other hand are able to approximate the influence of plasma parameter and magnetic topology - as well as its geometry - before the experiment is modified.

The plasma in these regions is well described by the magneto-hydrodynamical equations. This set of partial differential equations gives the coupled temporal evolution of the plasma fluid and the magnetic field. They are solved by using finite differences for the spatial derivations and a third order Runge-Kutta method for the temporal integration.

The fundamental property of a magnetic field is solenoidality. This property is in general not conserved in numerical simulations. One way to ensure a physical magnetic field is to extend the MHD equations. The corresponding extension was proposed by[9]. In the framework *racoon* the divergence of the magnetic field is transported to outer regions and damped on its way. This works only for localized problems, here it is an arc shaped plasma in a huge empty domain.

The localization in this problem offers the possibility for the use of an adaptive mesh refinement. Here the grid is adapted to the electric current density, so that the current arc is well resolved during its evolution without resolving the other, plasma and current free, regions.

A first comparison of the experiment FlareLab and the adaptive mesh simulations is shown in Fig. 4.

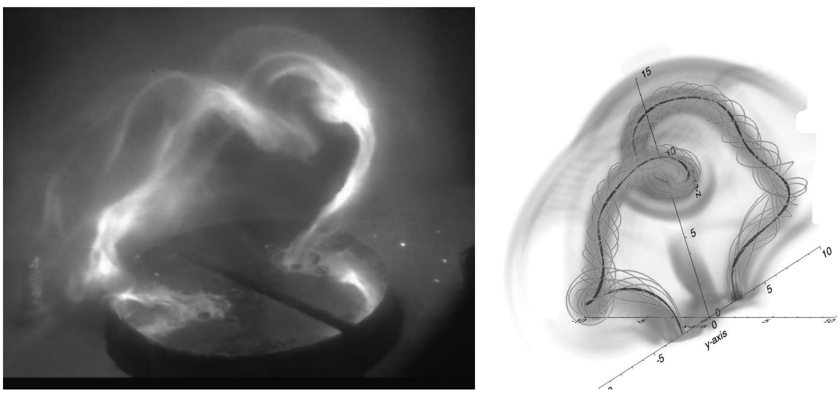

Figure 4. Comparison of the experiment FlareLab with simulations. Shown are high temperature regions (left) and current density and magnetic field lines (right).

5 Turbulence

Turbulence is an important and wide spread matter in today's research. From gas in molecular clouds to blood streaming through a heart valve one has to deal with turbulent flows and its properties. Although their generation is based on different forces and although they are enclosed by specific boundaries, there are features which all turbulent flows have in common.

The forces and boundaries naturally act on the large scales of the motion. From these scales turbulence generates a whole range of structures of different sizes down to the smallest scales where the dissipation transforms the kinetic energy into heat. The universality sets in at scales smaller than the boundary or forcing scale down to scales larger than the dissipation scales. Here the information of the geometry and the specific dissipation mechanism of the flow is lost and the motion is completely determined by the non-linear inertial interaction of the eddies. This range is called inertial range. Physical theories often deal with fundamental features such as scaling behaviour and intermittency of this range of scales. In order to analyze its properties numerically it is necessary to provide a large amount of scales. Numerical simulations of Euler- and Navier-Stokes-turbulence revealed that a resolution of at least 512^3 collocation points is needed to obtain an inertial range of scales within a turbulent flow.

Numerical investigations of different flows have to focus on different concerns. Incompressible flows are conveniently solved using pseudo-spectral codes. The formation of

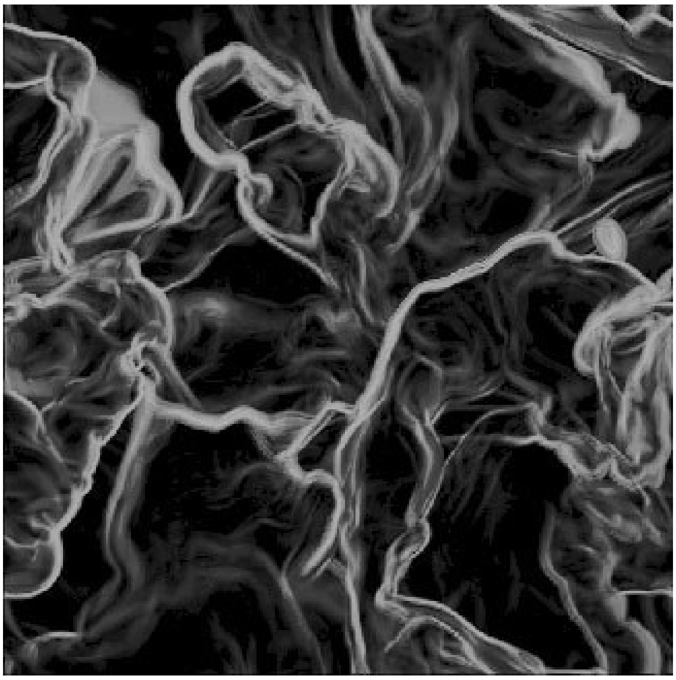

Figure 5. Volume rendering of vorticity of compressible turbulence

strong shocks in compressible gas dynamics needs a shock capturing central scheme and allows the application of adaptive mesh refining techniques (AMR)[10], so this problem is predestined to the *racoon* framework (see below). With this we examined isothermal Euler turbulence up to an effective resolution of 1024^3 cells. Fig. 5 shows the vorticity of a high Mach number compressible simulation.

While it is natural to perform incompressible MHD simulations with a pseudo-spectral code, compressible MHD simulations in real space appear to be more difficult. The proper way seems to be the constraint transport method[11] on a staggered grid in combination with divergence-free reconstruction[12] for the AMR.

Lagrangian statistics of turbulent fluid and magnetohydrodynamic flows has undergone a rapid development in the last 6 years to enormous progress in experimental techniques measuring particle trajectories[13]. Lagrangian statistics is not only interesting for obtaining a deeper understanding of the influence of typical coherent or nearly-singular structures in the flow but also of fundamental importance for understanding mixing, clustering and diffusion properties of turbulent astrophysical fluid and plasma flows. Tracer particles are employed in the incompressible as well as in the compressible code.

Concerning the incompressible case we computed the Lagrangian statistics of MHD- and neutral turbulence. The comparison revealed the intriguing and differing influence of the flow structures on the Eulerian and Lagrangian statistics[14] (see Fig. 6). The issue of intermittency was addressed by the computation of probability density functions (PDFs) and structure functions and comparison to a multifractal model[15].

Figure 6. Particle trajectories near singular events (left: Navier-Stokes, right: MHD)

In compressible turbulence we found an explanation for the PDF of the spatial particle distribution as a counterpart of the mass density field[16].

References

1. J. Dreher and R. Grauer, *Racoon: A parallel mesh-adaptive framework for hyperbolic conservation laws*, Parallel Comp. **31**, 913–932, (2005).
2. M. J. Berger and P. Colella, *Local adaptive mesh refinement for shock hydrodynamics*, J. Comp. Phys. **82**, 64–84, (1989).
3. A. Kurganov and E. Tadmor, *New high-resolution central schemes for nonlinear conservation laws and convection-diffusion equations*, J. Comp. Phys. **160**, 241–282, (2000).
4. G.W. Zumbusch, *On the quality of space-filling curve induced partitions*, Z. Angew. Math. Mech. **81**, 25–28 (2001).
5. R. Rabenseifer, *Hybrid Parallel Programming on HPC Platforms*, Fifth European Workshop on OpenMP, Aachen, (2003).
6. H. Homann and R. Grauer, *Impact of the floating-point precision and interpolation scheme on the results of DNS of turbulence by pseudo-spectral codes*, Comp. Phys. Comm. **177**, 560–565, (2007).
7. J. Hansen and P. Bellan, *Laboratory simulations of solar prominences*, APS Meeting Abstracts, 1089, (2001).
8. J. Hansen, S. Tripathi and P. Bellan, *Co- and counter-helicity interaction between two adjacent laboratory prominences*, Physics of Plasmas **11**, 3177–3185, (2004).
9. A. Dedner, F. Kemm, D. Kröner, C.-D. Munz, T. Schnitzer and M. Wesenberg, *Hyperbolic Divergence Cleaning for the MHD Equations*, J. Comp. Phys. **175**, 645–673, (2002).
10. A. Kritsuk, M. L. Norman and P. Padoan, *Adaptive mesh refinement for supersonic molecular cloud turbulence*, The Astrophysical Journal **638**, L25–L28, (2006).
11. U. Ziegler, *A central-constrained transport scheme for ideal magnetohydrodynamics*, J. Computat. Phys. **196**, 393–416, (2004).
12. D. S. Balsara, *Second-Order-accurate Schemes for Magnetohydrodynamics with Divergence-free Reconstruction*, The Astrophysical Journal Supplement Series **151**, 149–184, (2004).
13. A. La Porta, G. A. Voth, A. M. Crawford, J. Alexander and E. Bodenschatz, *Fluid particle accelerations in fully developed turbulence*, Nature **409**, 1017–1019, (2001).
14. H. Homann, R. Grauer, A. Busse and W. C. Müller, *Lagrangian Statistics of Navier-Stokes- and MHD-Turbulenz*, to appear in J. Plasma Phys, doi:10.1017/S0022377807006575, (Published online 31 May 2007).
15. L. Biferale, G. Boffetta, A. Celani, B. J. Devenish, A. Lanotte and F. Toschi, *Multifractal Statistics of Lagrangian Velocity and Acceleration in Turbulence*, Phys. Rev. Lett. **93**, 064502, (2004).
16. Ch. Beetz, Ch. Schwartz, J. Dreher and R. Grauer, *Density-PDFs and Lagrangian Statistics of highly compressible Turbulence*, arXiv:0707.1798v1 [physics.flu-dyn]