*SF2A 2021*
*A. Siebert, K. Baillié, E. Lagadec, N. Lagarde, J. Malzac, J.-B. Marquette, M. N'Diaye, J. Richard, O. Venot (eds)*

# GINEA AND DYABLO

D. Aubert[1] and A. Durocher[2]

**Abstract.** GINEA (Groupe d'investigation numérique pour l'exascale en astrophysique) is an initiative of numerical astrophysicists dedicated to the development of future astrophysics simulation codes. During the Exascale era, the next generation of supercomputers, massively parallel and hybrid, will provide significant challenges to the current generation of simulation codes. Prototypes and code evolutions are being investigated and discussed worldwide to prepare for the advent of these machines. Within GINEA such investigations are conducted with the new Dyablo AMR hydrodynamics code, developed at the CEA. Thanks to the hardware agnostic library Kokkos, Dyablo is currently able to run on multi-GPU architectures with promising performances and parallel scaling. The features of Dyablo are presented as well as the objectives of GINEA.

Keywords: numerical simulations, high-performance computing

## 1 Introduction

The near future of High Performance Computing (HPC) is often referred to as the 'Exascale Era' due to the upcoming generation of super-computers that aim at being able to reach a computing power of one *Exaflop* , i.e. $10^{18}$ floating point operations per second. Corresponding to a few times the performance of the current most powerful machines, this threshold should be passed in the next few years. For example, *Frontier*, the next flagship of U.S. supercomputing is expected to reach this level of computing power and will be delivered before the end of 2021. Likewise, Euro-HPC *, a European Union joint initiative, is expected to install several computers with Exascale abilities in the next few years. Of course, numerical astrophysics, and notably fields that are driven by numerical simulations, should benefit from such an evolution. Many codes display high level of parallelization that are able to harness the processing power provided by thousands, if not tens or hundreds of thousands, cores, to simulate astrophysical processes highly resolved in time and space with large dynamical ranges.

However, the advent of the Exascale era relies on a few paradigms that come with their share of challenges such as :

- an increase in the number of CPUs being made available. Load balancing can be more difficult to optimize leading to parallelized codes with performances unable to scale with the number of cores. The available memory per core has also a tendency to decrease, simply forbidding some applications to be done. Finally, a large number of computing units often imply significant I/Os, putting high stresses on storage systems.

- hybrid computing. Supercomputers rely more and more on co-processing devices such as graphics processing units (GPUs). These devices can provide significant boosts in computing power, but rely on parallelization or memory handling paradigms that can significantly differ from the ones applied to standard CPUs. In practice, some current codes are found to be unable, by design, to benefit from them.

[1] Observatoire Astronomique de Strasbourg, Université de Strasbourg, CNRS UMR 7550, 11 rue de l'Université, 67000 Strasbourg, France
[2] IRFU, CEA, Université Paris-Saclay, 91191 Gif-sur-Yvette, France

*https://eurohpc-ju.europa.eu/

- new architectures. For example, the last few years have seen the emergence of ARM architectures for CPUs, disputing the previous dominance of x86 processors and it is expected that the future flaships of the Euro-HPC Joint initiative will rely on such technology. It's not clear yet how existing codes will behave on such chips and more generally it illustrates how changes in architectures and/or devices is an obstacle to the long-term development and support of scientific applications that uses HPC installations.

It should be noted that this ever changing landscape is also nurtured by the rise of new actors such as the field of artificial intelligence or High Performance Data Analysis (HPDA). Their specifications can differ from the ones required by the more traditional actors of supercomputing (such as numerical simulations) and a convergence of the requirements, techniques and methods of these different field has to be expected. Such convergence will inevitably lead to changes in the current standard solutions for numerical astrophysics.

## 2 GINEA : Groupe d'Investigation Numerique pour l'Exascale en Astrophysique

GINEA is group of scientists from French laboratories that aims at defining, implementing and prototyping future astrophysical simulation codes tailored for the Exascale Era. These codes are expected to be massively parallel, hybrid and compatible with multiple architectures. More importantly such codes should be maintainable with a relative ease and capable of long-term evolution, as it is difficult to anticipate today what will be the standards (methods, hardware,...) of the Exascale era and beyond. In addition, GINEA also aims at pushing for a collaborative development between GINEA numerical astrophysicists, computational scientists and software engineers. This strategy should provide solid foundations, professional standards for code engineering, thus ensuring long-term support and code evolutions.

These future exascale codes should also answer the challenges specific to astrophysical situations. Among the desirable features, one can list the ability to deal with :

- the coupling of multiple physics such as gravity, hydrodynamics, radiative transfer, thermo-chemistry, MHD, etc..
- heterogeneous data with often combined eulerian (grid-based) and lagrangian (particle-based) descriptions
- great dynamics of length and time scales, which inevitably lead to load-balancing and memory issues
- production of large amount of data. For example, a large simulation project can produce petabytes of data, i.e. equivalent to several years of state-of-the art observational surveys.

The GINEA initiative started in November 2019, at the Maison de la Simulation and this first meeting led to the declared intent of looking at ways to have prototypes and platforms for new simulation codes with the aforementioned capabilities. As of today, GINEA gathers about 20 computer scientists and numerical astrophysicists from CEA, CNRS and Universities, as well as collaborators in foreign institutions, with a majority coming from the RAMSES code (Teyssier 2002) community. It benefits from support by 4 National Programs (PNCG, PCMI, PNPS, PNHE), reflecting the scope of our fields of research in astrophysics. In practice, (virtual) meetings are regularly organised on a monthly basis to discuss about numerical techniques and strategical choices about code design. Most importantly such meetings are also dedicated to updates about the developments made at the CEA on the adaptive mesh refinment code DYABLO, described in the following section.

## 3 DYABLO

DYABLO is an adaptive mesh refinement (AMR) hydrodynamics simulation code, initially designed by P. Kestener at the CEA, and currently developed by A. Durocher as well as M. Delorme on a different fork dedicated to physics of the Sun. DYABLO is a 2D/3D parallel code, capable of being deployed on many cores in distributed memory (MPI) and shared memory systems (OpenMP). The code is also able to benefit from GPUs parallel devices (Nvidia/CUDA) and it can be run in hybrid configurations, e.g. on multiple CPU nodes equipped with GPUs. This versatility in terms of parallelisation is made possible by relying on the C++ KoKKos[†] library (Edwards et al. 2014), which provides means of high-level development for multiple parallel

[†]https://github.com/kokkos/kokkos

architectures and in an agnostic and optimized way. Kokkos therefore provides portability to Dyablo, even for architectures that may arise in the future as long as they are supported by the library.

The AMR structure is currently managed by the PABLO CPU-only library[‡] but the development of a portable (CPU/GPU) custom AMR library is currently near completion. The Dyablo AMR strategy can be either cell-based or block-based, both having their own pros and cons in terms of e.g. parallel performance and memory consumption on GPUs. In terms of physics, the code currently implements a MUSCL-Hancock hydrodynamics, as well as a static gravity. The inclusion of self-gravity and particles (e.g. for collisionless dark matter, stars, tracers) is currently under study. In terms of design, the code, written in C++, aims at being modular to ease e.g. the addition of future astrophysics modules. Finally, it should be noted that beyond astrophysics, Dyablo is written as a generic platform for parallel AMR simulation, hence the emphasis on modularity and high-level templates.

Since the code is still in development, performance assessments should be taken with caution. Nevertheless, preliminary benchmarks on 3D Sedov Blast waves ($256^3$ + 2 refinement levels, equiv. $1024^3$) already show an acceleration factor of a few using a Jean-Zay (IDRIS) V100 GPU for Dyablo compared to Ramses deployed on 40 cores on the same Jean-Zay nodes. Likewise weak scaling appears satisfactory up to 256 GPUs on the same test-case. Obviously, these promising results will have to be revisited on more demanding astrophysical test-cases. Only then full confidence would be claimed about the potential of Dyablo as a generic and performant astrophysics simulation code for exascale machines.

## 4 Conclusions and Prospects

The AMR code Dyablo is currently in heavy development and while initial benches are very promising, a long way remains until a fully functional code for state-of-the art astrophysical productions. In particular, fundamental choices about code design (esp. regarding the AMR data structure) are being currently discussed to implement functionalities related to particles and to self-gravity. For example, the introduction of particles or the use of a multigrid solver for self-gravity may require significant changes in the way Dyablo handles the AMR tree to achieve good performance, especially on GPU accelerators. The GINEA group is thus a place to discuss the priorities in terms of features and performances, to decide for the best route to be taken for a future simulation framework useful for state-of-the exascale astrophysics. Beyond that, once this framework (Dyablo or an evolution from Dyablo) has been settled, numerical astrophysicists will have to adopt it and expand it by adding the astrophysical modules required for their specific science. That will be the task of the GINEA initiative for the months and years to come.

This work has been supported by the Programmes Nationaux Cosmologie et Galaxies (PNCG), Hautes Energies (PNHE), Physique Stellaire (PNPS) and Milieu Interstellaire (PCMI).

## References

Edwards, H. C., Trott, C. R., & Sunderland, D. 2014, Journal of Parallel and Distributed Computing, 74, 3202 , domain-Specific Languages and High-Level Frameworks for High-Performance Computing

Teyssier, R. 2002, A&A, 385, 337

[‡]https://optimad.github.io/PABLO/doxy/html/index.html