SF2A 2022
J. Richard, A. Siebert, E. Lagadec, N. Lagarde, O. Venot, J. Malzac, J.-B. Marquette, M. N'Diaye, D. Briot (eds)

NOVEL NUMERICAL METHODS FOR SOLAR CONVECTION: THE *DYABLO WHOLE-SUN* ADAPTATIVE MESH REFINEMENT CODE

M. Delorme[1], A. Durocher[2], A. S. Brun[1] and A. Strugarek[1]

Abstract. We present a new solar simulation code named Dyablo Whole-Sun (DWS) and the first steps of its validation. DWS is a novel portable high-performance code aiming at making the first holistic simulations of the Sun, from the radiative interior to the corona. We discuss the validation of the development of the code using a solar convection benchmark in Cartesian geometry.

Keywords: Software, hydrodynamics, benchmark, solar physics

1 Introduction

Convection is a central process in the dynamics of the Sun and solar-like stars. Although Solar convection has been studied and simulated for decades from the seminal works of Gilman & Glatzmaier (1981), Nordlund (1982) and Hurlburt et al. (1984) until the more recent advances of Brun & Toomre (2002), Miesch et al. (2008) and Hotta & Kusano (2021), this process and its consequences on the solar dynamics is still under active scrutiny. One classical problem with the simulation of convection is that no code is currently capable of generating rising and emerging sunspots from first principles. Sunspots in simulations are often introduced by hand in the setups, while global dynamo simulations do not have realistic enough atmospheres allowing for flux emergence. One way to make progress on these topics is to take a holistic approach, linking the global interior of the Sun with the atmosphere and the corona (See for instance arguments developed in Nelson et al. (2014), Warnecke et al. (2016) and Perri et al. (2021)). However, building a code able to approach the problem globally is extremely difficult. The variety of physical regimes, as well as the sheer number of points required to resolve convection to a sufficient level is huge. For instance, an ideal resolution at the surface of the Sun would require meshes 10 km high and 30 km wide on a sphere of 700 Mm radius with the need to adapt along the radius both the horizontal and vertical resolutions. A gross estimate indicates that more than 100 billions cells are needed, so this requires extreme computing resources. Thus, developing a new solar global code requires high-performance computing expertise and modern hardware. Such a high-level of technology translates into complex codes which in turn necessitate regular tests and benchmarks in order to demonstrate the validity of the solutions retained.

In this work, we introduce the new code Dyablo-Whole Sun as well as a co-developed validation benchmark. This new code aims at the first global simulations of the Sun from the radiative interior to the solar corona, and is built with high-performance portability in mind. The benchmark is based on well-known setups in the community to validate each step of the code development. In the next sections, we will describe in section 2 the general benchmark used for the evaluation of hydrodynamical convection. We will then briefly present our new code in section 3. Finally we will present the results of the benchmark and conclude in sections 4 and 6.

[1] Département d'Astrophysique/AIM, CEA/IRFU, CNRS/INSU, Univ. Paris-Saclay & Univ. de Paris, 91191 Gif-sur-Yvette, France
[2] Département d'Électronique, des Détecteurs et d'Informatique pour la Physique, CEA/IRFU, 91191 Gif-sur-Yvette, France

2 Benchmark

2.1 General equations

In this work, we solve the compressible Navier-Stokes equations with radiative transfer in the radiative diffusion approximation (ie using a thermal conduction term) :

$$\begin{aligned} \partial_t \rho + \nabla \cdot (\rho \boldsymbol{u}) &= 0 \\ \partial_t (\rho \boldsymbol{u}) + \nabla \cdot (\rho \boldsymbol{u}^2 + \boldsymbol{I} p - \boldsymbol{\tau}) &= \rho \boldsymbol{g} \\ \partial_t E + \nabla \cdot ([E+p]\boldsymbol{u} + \tau \boldsymbol{u} + \kappa \nabla T) &= \rho \boldsymbol{u} \cdot \boldsymbol{g} \end{aligned} \tag{2.1}$$

The quantities ρ, $\boldsymbol{u}$, E, p, T, $\boldsymbol{g}$, κ are respectively the density, velocity, total energy, pressure, temperature, gravitational acceleration and the thermal conductivity coefficient. The viscosity tensor $\boldsymbol{\tau}$ is defined as

$$\boldsymbol{\tau} := \mu \left[\nabla \boldsymbol{u} + (\nabla \boldsymbol{u})^T - \frac{2}{3} (\nabla \cdot \boldsymbol{u}) \boldsymbol{I} \right] \tag{2.2}$$

Where μ is the dynamic viscosity coefficient. The total energy E is defined as $E = \rho(|\boldsymbol{u}|^2/2 + e)$ with e the internal energy. Finally, the system is closed using a perfect gas equation of state $e = p/[\rho(\gamma - 1)]$ with γ the adiabatic index. We consider a mono-atomic gas such that $\gamma = 5/3$. For simplicity, we use an adimensional system, where the gas constant is set to $R = 1$, and which implies that the temperature of the system can be obtained directly as $T = p/\rho$.

2.2 Benchmark setup

The setup of the benchmark is adapted from the work of Cattaneo et al. (1991) (C91). We consider a cartesian slab representing a portion of the convective zone of the Sun close to the surface. Considering the z axis to be the vertical direction, and gravity pointing towards the positive z-axis. The slab is initialized using a polytropic model with $\rho(z) = (1 + \theta z)^m$ and $P(z) = (1 + \theta z)^{m+1}$. Where m is the polytropic index and θ the temperature gradient over the domain. In our reference frame of coordinates, $z = 0$ is the top of the domain while $z = 1$ is the bottom. Hence, gravity is pointing upwards. The model is initialized in hydrostatic equilibrium, which leads to a gravitational acceleration along z equal to $g_z = \theta(m + 1)$. Finally, the instability is triggered by adding a small perturbation (of the order 10^{-3}) to the initial pressure.

The slab is setup with a horizontal to vertical aspect ratio of 4. Horizontal boundaries are periodic while vertical boundaries are stress-free impenetrable walls : $\partial_z u_x = \partial_z u_y = \rho u_z = 0$. At the top, the temperature is constant $T(z = 0) = 1$ while at the bottom ($z = 1$), the initial temperature gradient is imposed : $\partial_z T = \theta$.

The setup is characterized by the free parameters θ, m, μ, and κ. For all the runs, we keep a constant thermal conduction coefficient $\kappa = 0.07$ as well as a polytropic index $m = 1$. We introduce the Prandtl number defined as $\sigma := \mu c_p/\kappa$. Each run is thus parametrized by two constants : θ and σ.

2.3 Runs and evaluation

This setup is run four times, with the combinations of $\theta = (10, 20)$ and $\sigma = (1.0, 0.3)$ implying different degrees of vertical stratification and turbulence. Both setups with $\theta = 10$ are similar to setups 1 and 2 of C91. Setups 3 and 4 double the temperature gradient and the density stratification, multiplying thus by 10 the effective Rayleigh and leading to more turbulent convection states. To make sure each run has relaxed, we run them all for 200 time-units.

Each run is then evaluated according to three main criteria. First the temporal evolution of kinetic and total energy in the slab. This metric has three purposes : 1- Checking that the triggering of the instability happens roughly at the same times across all runs; 2- Checking that the resting energy after thermal relaxation is the same and 3- Checking that the relaxation time is roughly the same for each code. Second, we check the vertical profiles of the enthalpy flux (F_e), the kinetic flux (F_k), the acoustic flux (F_p) and the buoyancy work (W_b). These fluxes are defined as

$$F_e = \frac{\gamma}{\gamma - 1} T' \rho u_z \qquad F_k = \frac{1}{2} \rho u_z |\boldsymbol{u}|^2 \qquad F_p = u_z p' \qquad W_b = \theta(m+1) u_z \rho' \tag{2.3}$$

where the primed variables are defined as variation to the horizontal average (eg. $T'(x,y,z) = T(x,y,z) - \langle T \rangle_{x,y}$). All these fluxes are computed for each computational element of the domain, then averaged horizontally to obtain a vertical profile. These profiles are then averaged over the course of a few convective turnover times to obtain robust statistics. Finally, we also plot vertical profiles for the energy ratio (r_e) and enstrophy ratio (r_ω). These quantities are defined as

$$r_e(z) = \frac{\langle u_z^2 \rangle_{x,y}}{\langle u_x^2 \rangle_{x,y} + \langle u_y^2 \rangle_{x,y}} \qquad r_\omega(z) = \frac{\langle \omega_x^2 \rangle_{x,y} + \langle \omega_y^2 \rangle_{x,y}}{\langle \omega_z^2 \rangle_{x,y}}, \tag{2.4}$$

with $\boldsymbol{\omega} := \nabla \times \boldsymbol{u}$ the vorticity. These three quantities allow us to compare that all the codes included in the benchmark are having the same behaviour in terms of temporal evolution, and final statistical behaviour.

3 Dyablo-Whole Sun

The benchmark is designed to be tested on many codes. For the moment, all runs in this work have been tested and validated with respect to C91 using the Dyablo-Whole Sun (DWS) code. Dyablo is a high-performance adaptive mesh refinement (AMR) (Berger & Colella 1989) framework developed with performance portability and exascale computing in mind. DWS is an applications built on top of this framework and is specifically aimed at the simulation of the Sun and Solar-like stars. DWS is developed in modern C++. Distributed parallelism is handled by MPI while shared-memory parallelism is managed by the portable performance library Kokkos (Trott et al. 2022). DWS solves the Navier-Stokes equations using a finite-volume approach with first and second order schemes (Euler, Hancock or RK2) and a piecewise linear reconstruction with a Minmod slope limiter. In this work, all simulations use the RK2 solver. Parabolic terms (viscosity and thermal conduction) are integrated explicitely using a diffusive flux approach.

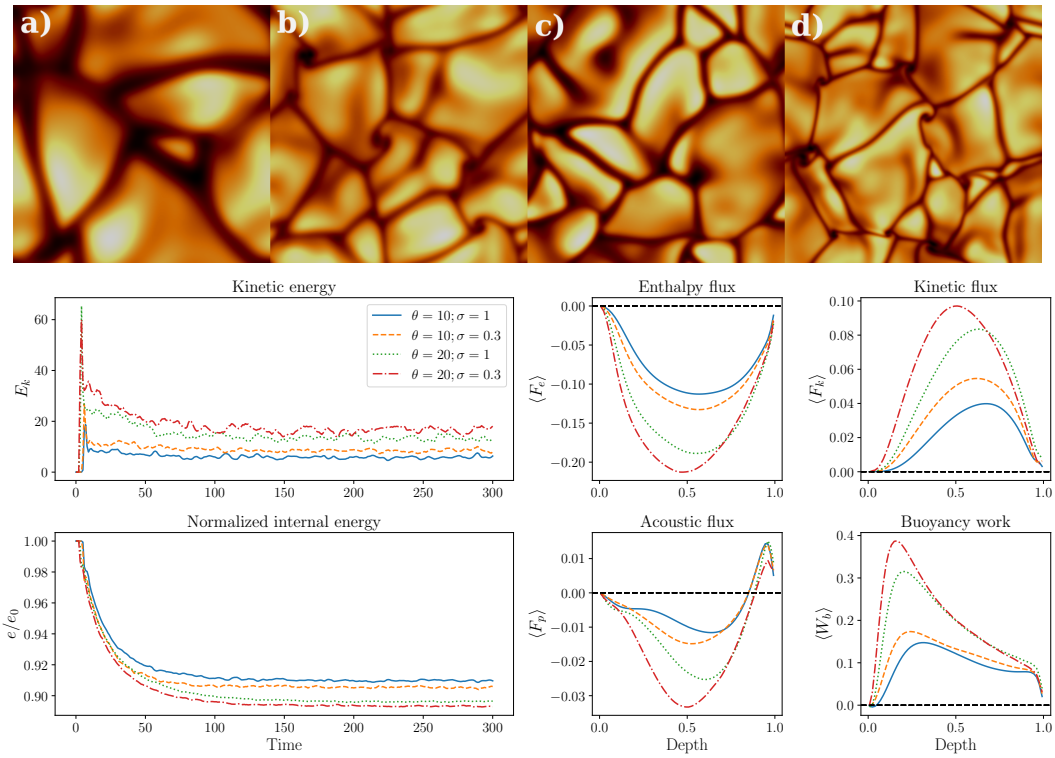

Fig. 1. Comparison of the four different setups in fixed grid. **Top:** The slices of temperature are extracted near the surface ($z = 0.1$). Black is cold, yellow is hot. **a**: $\theta = 10$ and $\sigma = 1$; **b** : $\theta = 10$ and $\sigma = 0.3$; **c**: $\theta = 20$ and $\sigma = 1$; **d**: $\theta = 20$ and $\sigma = 0.3$. **Bottom left**: Evolution of the kinetic energy and the normalized internal energy through time. By the end, all simulations have reached thermal relaxation. **Bottom right**: Horizontally averaged flux profiles over a few turnover-times. We show in turn Enthalpy, Kinetic and Acoustic fluxes and Buoyancy work.

4 Results

We ran the four 3D setups using DWS with a fixed-grid of resolution $256 \times 256 \times 64$. All setups were run on our own cluster Soulèu, on 8 nodes with AMD-EPYC-7452 processors. To demonstrate the portability of DWS, additional runs were made on the Nvidia V100 partition of the Irene supercomputer in France. We used Kokkos' OpenMP backend for the CPU runs on Souleu and the CUDA backend for the GPU runs. All runs were made using the RK2 solver.

After the run finished, we extracted the statistics described in section 2.3 for the fixed grid runs as well as temperatures slices near the top of the domain ($z = 0.1$). Results can be seen on Fig. 1. All results are consistent with the original results of C91 for the two similar cases although buoyancy driving seems to be less pronounced in our models near the surface. However, when comparing with other codes, we find that the other codes are also in agreement with our profiles (Delorme et al. 2023, in prep).

5 AMR Prospects

Runs with adaptive mesh refinement are currently being calculated and processed. Preliminary results on a slightly different 2D setup are shown on Fig. 2. Work will be carried-on to evaluate the effects of different refinement criteria as well as their free parameters on the simulations. The quality of the AMR runs will be evaluated on three criteria : 1- physical convergence to fiducial fixed-grid runs; 2- time to solution; 3- number of cells in simulation.

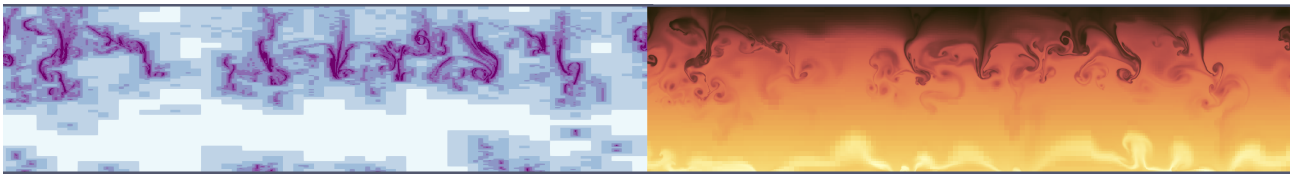

Fig. 2. 2D cartesian slab of convection in DWS with AMR activated. **Left**: Refinement level of the simulation. Light blue indicates low-resolution regions while violet indicate high-resolution cells. **Right**: Corresponding temperature slice. Yellow is hot while black is cold. Refinement is triggered for high horizontal gradients of temperature, effectively refining the descending convective plumes.

6 Conclusions

We have presented Dyablo-Whole Sun, our new code aimed at the first global simulations of the Sun. This code relies on modern libraries to ensure high-performance and portability on modern hardware. The benchmark presented is based on the seminal work C91 and provides a solid test-bed for the development of DWS. A number of other codes are being included in the benchmark to make sure that convergence within the community is guaranteed. The benchmark will be extended in the future to include more physics, for instance MHD to perform flux emergence experiments, to test for increasingly complex setups until we reach simulations of the Sun as a whole. Eventually, DWS will be rendered open-source and accessible to the community.

We acknowledge the financial support of the ERC Whole Sun Synergy grant #810218, INSU/PNST and CNES Solar Orbiter supports. We are thankful to GENCI via project 1623.

References

Berger, M. & Colella, P. 1989, Journal of Computational Physics, 82, 64

Brun, A. S. & Toomre, J. 2002, The Astrophysical Journal, 570, 865

Cattaneo, F., Brummell, N. H., Toomre, J., Malagoli, A., & Hurlburt, N. E. 1991, The Astrophysical Journal, 370, 282

Gilman, P. A. & Glatzmaier, G. A. 1981, The Astrophysical Journal Supplement Series, 45, 335

Hotta, H. & Kusano, K. 2021, Nature Astronomy, 5, 1100

Hurlburt, N. E., Toomre, J., & Massaguer, J. M. 1984, The Astrophysical Journal, 282, 557

Miesch, M. S., Brun, A. S., DeRosa, M. L., & Toomre, J. 2008, The Astrophysical Journal, 673, 557

Nelson, N. J., Brown, B. P., Brun, A. S., Miesch, M. S., & Toomre, J. 2014, Solar Physics, 289, 441
Nordlund, Å. 1982, Astronomy and Astrophysics, 107, 1
Perri, B., Brun, A. S., Strugarek, A., & Réville, V. 2021, The Astrophysical Journal, 910, 50
Trott, C. R., Lebrun-Grandie, D., Arndt, D., et al. 2022, IEEE Transact. on Parallel and Distributed Systems, 33, 805
Warnecke, J., Käpylä, P. J., Käpylä, M. J., & Brandenburg, A. 2016, A&A, 596, A115