Lawrence Berkeley National Laboratory
LBL Publications

Title

Practical Effects of Integrating Temperature with Strang Split Reactions

Permalink

https://escholarship.org/uc/item/4mv9d420

Authors

Zingale, M
Katz, MP
Willcox, DE
et al.

Publication Date

2021-03-23

Peer reviewed

eScholarship.org Powered by the California Digital Library
University of California

arXiv:2103.13193v1 [astro-ph.IM] 23 Mar 2021

Practical Effects of Integrating Temperature with Strang Split Reactions

M. ZINGALE,[1] M. P. KATZ,[2] D. E. WILLCOX,[3] AND A. HARPOLE[1]

[1] *Dept. of Physics and Astronomy, Stony Brook University, Stony Brook, NY 11794-3800*
[2] *NVIDIA Corporation*
[3] *Lawrence Berkeley National Laboratory, Berkeley, CA*

ABSTRACT

For astrophysical reacting flows, operator splitting is commonly used to couple hydrodynamics and reactions. Each process operates independent of one another, but by staggering the updates in a symmetric fashion (via Strang splitting) second order accuracy in time can be achieved. However, approximations are often made to the reacting system, including the choice of whether or not to integrate temperature with the species. Here we demonstrate through a simple convergence test that integrating an energy equation together with reactions achieves the best convergence when modeling reactive flows with Strang splitting. Additionally, second order convergence cannot be achieved without integrating an energy or temperature equation.

Keywords: hydrodynamics—methods: numerical

1. INTRODUCTION

Simulations of stellar flows require solving the equations of hydrodynamics coupled with a nuclear reaction network. The equations of hydrodynamics with reacting sources are:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{U}) = 0 \tag{1}$$

$$\frac{\partial (\rho X_k)}{\partial t} + \nabla \cdot (\rho X_k \mathbf{U}) = \rho \dot{\omega}_k \tag{2}$$

$$\frac{\partial (\rho \mathbf{U})}{\partial t} + \nabla \cdot (\rho \mathbf{U} \mathbf{U}) + \nabla p = 0 \tag{3}$$

$$\frac{\partial (\rho E)}{\partial t} + \nabla \cdot [(\rho E + p) \mathbf{U}] = \rho \dot{S} \tag{4}$$

where ρ is the density, $\mathbf{U}$ is the velocity vector, X_k are the species mass fractions with creation rates $\dot{\omega}_k$, p is the pressure, E is the specific total energy, and $\dot{S}$ is the nuclear energy generation rate.

When we are reacting, we can look at internal energy

$$\rho \frac{De}{Dt} + p \nabla \cdot \mathbf{U} = \rho \dot{S}, \tag{5}$$

where e is the specific internal energy or alternately, we can evolve the temperature, T,

$$\rho c_v \frac{DT}{Dt} = \rho \left(\frac{p}{\rho^2} - \left.\frac{\partial e}{\partial \rho}\right|_T \right) \frac{D\rho}{Dt} + \rho \dot{S}, \tag{6}$$

where $c_v = \partial e/\partial T|_\rho$ (this form neglects composition changes, see Almgren et al. 2008).

2. NUMERICAL METHODOLOGY

We use the freely-available Castro simulation code (Almgren et al. 2010; Almgren et al. 2020) to solve the equations of hydrodynamics, using an unsplit piecewise parabolic method coupled with reactions. Castro uses either Strang splitting or spectral deferred corrections (SDC) to couple the hydrodynamics and reactions (Zingale et al. 2019). Here we focus on the Strang splitting.

In a Strang split evolution (Strang 1968), we update the full hydrodynamics state, $\boldsymbol{\mathcal{U}}$, as:

$$\boldsymbol{\mathcal{U}}^{n+1} = \mathbf{R}_{\Delta t/2}\mathbf{A}_{\Delta t}\mathbf{R}_{\Delta t/2}\boldsymbol{\mathcal{U}}^{n} \tag{7}$$

where $\mathbf{R}_{\Delta t/2}$ is the reaction update through a timestep $\Delta t/2$ and $\mathbf{A}_{\Delta t}$ is the advective update through Δt. We see with this splitting, each process operates on the state left behind by the previous operation, and the staggering of the physics is done to give second order accuracy in time.

During reactions, we neglect the hydrodynamics terms, so the reactive system updates according to:

$$\frac{D\rho}{Dt} = 0 \tag{8}$$

$$\frac{DX_k}{Dt} = \dot{\omega}_k \tag{9}$$

with either

$$\frac{DT}{Dt} = \frac{\dot{S}}{c_v} \tag{10}$$

or

$$\frac{De}{Dt} = \dot{S} \tag{11}$$

There are several different approaches taken in the literature to this operator-split reacting system, including some approximations that make integrating the reaction system computationally less expensive:

- Evolve (X_k) only. This neglects temperature evolution completely in the burn, only evolving Eq. 9. This is the method used in Fryxell et al. (2000).

- Evolve (X_k, T). This is used in Pakmor et al. (2012) and García-Senz et al. (2013). To avoid expensive equation of state calls in getting the specific heat, we can optionally "freeze" the value of c_v at the start of integration. This was discussed in Bell et al. (2004) and until recently was the default method in Castro.

- Evolve (X_k, e), and get T from e using the equation of state. This was discussed in Fryxell et al. (1989) and is the current default method in Castro.

Raskin et al. (2010) also propose a hybrid system where the first approach is used in most cases, switching to the second approach only near NSE. For all of these cases, density is constant during the reaction operation. Depending on how vigorous the burning is and how much the temperature changes during a hydrodynamic timestep, one or more of these methods may be reasonable. For explosive reactions, we expect that evolving the full system will be required. The goal of this note is to try to quantify the convergence of a reacting hydrodynamics code with these different approximations.

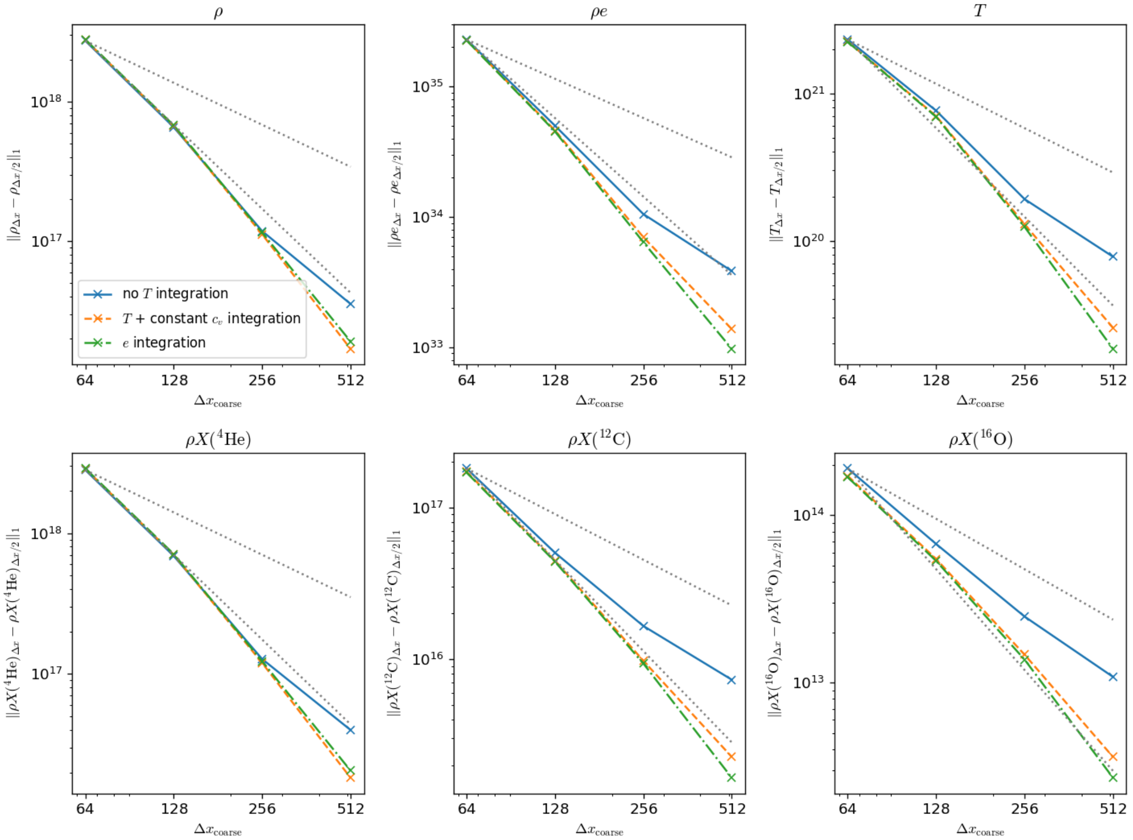

Figure 1. Convergence of fluid quantities as a function of resolution for 3 different Strang equation systems: just evolving X_k, evolving (X_k, T) with c_v held fixed, and evolving (X_k, e). The dotted lines show ideal first and second order convergence.

In Zingale et al. (2019), we introduced a test problem where we could measure the convergence of a reacting hydrodynamics problem via Richardson extrapolation—this was an acoustic pulse with helium burning via 3-α and $^{12}\mathrm{C}(\alpha, \gamma)^{16}\mathrm{O}$. Initially, the domain is pure ^{4}He, but both ^{12}C and ^{16}O are created as time evolves. The published tests showed that we can get overall 4th order in space and time convergence with SDC coupling. Here we run the same test with Strang coupling, looking at the various approaches at incorporating a temperature / energy equation in the reactive portion of the update.

For each method, we run the Castro `reacting_convergence` test problem at 5 resolutions: 64^2, 128^2, 256^2, 512^2, and 1024^2, with the timestep kept fixed in proportion to the grid resolution. We then compute four errors between adjacent resolutions by coarsening the finer resolution run, and computing the L_1 norm over all zones.

Figure 1 shows the norm of the error vs. the coarse run resolution. The slope of these lines is a measure of the convergence rate (Oberkampf & Roy 2010). We see that all methods converge at least second order for density, but for the thermodynamic quantities, (ρe) and T, the method where only X_k is evolved during reactions has larger errors and much worse convergence than the other methods. Looking at the trace nuclei generated in the burning, ^{12}C and ^{16}O, we see a large difference between the two methods that evolve some sort of energy and the one method where only X_k is evolved—the latter converging essentially first order at high resolution.

3. SUMMARY

Looking at global convergence of a reacting hydrodynamics problem we see that second order convergence is only realized when temperature or energy is evolved alongside the composition when using a Strang-split approach to reactions. This is just a single, rather simple problem, but this suggests that reactive hydrodynamics simulations should switch to integrating temperature or another energy equation together with reactions to yield better overall convergence and accuracy. This complements the work of Müller (1986) which showed that when evolving near nuclear statistical equilibrium, evolving entropy with the system is needed for stability. We expect that for problems with vigorous burning, such as detonations, directly coupling the composition and thermodynamic evolution will be especially important.

Castro is freely available at https://github.com/AMReX-Astro/Castro. The work at Stony Brook was supported by DOE/Office of Nuclear Physics grant DE-FG02-87ER40317. This material is based upon work supported by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research and Office of Nuclear Physics, Scientific Discovery through Advanced Computing (SciDAC) program under Award Number DE-SC0017955. This research was supported by the Exascale Computing Project (17-SC-20-SC), a collaborative effort of the U.S. Department of Energy Office of Science and the National Nuclear Security Administration.

Software: AMReX (Zhang et al. 2019), Castro (Almgren et al. 2020), matplotlib (Hunter 2007), NumPy (Oliphant 2007), VODE (Brown et al. 1989)

REFERENCES

Almgren, A., Sazo, M. B., Bell, J., et al. 2020, Journal of Open Source Software, 5, 2513, doi: 10.21105/joss.02513

Almgren, A. S., Bell, J. B., Nonaka, A., & Zingale, M. 2008, ApJ, 684, 449

Almgren, A. S., Beckner, V. E., Bell, J. B., et al. 2010, ApJ, 715, 1221, doi: 10.1088/0004-637X/715/2/1221

Bell, J. B., Day, M. S., Rendleman, C. A., Woosley, S. E., & Zingale, M. A. 2004, Journal of Computational Physics, 195, 677

Brown, P. N., Byrne, G. D., & Hindmarsh, A. C. 1989, SIAM J. Sci. Stat. Comput., 10, 1038

Fryxell, B., Müller, E., & Arnett, D. 1989, Max-Planck-Institut für Astrophysik Report 449

Fryxell, B., Olson, K., Ricker, P., et al. 2000, ApJS, 131, 273

García-Senz, D., Cabezón, R. M., Arcones, A., Relaño, A., & Thielemann, F. K. 2013, MNRAS, 436, 3413, doi: 10.1093/mnras/stt1821

Hunter, J. D. 2007, Computing in Science and Engg., 9, 90, doi: 10.1109/MCSE.2007.55

Müller, E. 1986, Astronomy and Astrophysics, 162, 103

Oberkampf, W. L., & Roy, C. J. 2010, Verification and Validation in Scientific Computing (Cambridge)

Oliphant, T. E. 2007, Computing in Science and Engg., 9, 10, doi: 10.1109/MCSE.2007.58

Pakmor, R., Edelmann, P., Röpke, F. K., & Hillebrandt, W. 2012, MNRAS, 424, 2222, doi: 10.1111/j.1365-2966.2012.21383.x

Raskin, C., Scannapieco, E., Rockefeller, G., et al. 2010, ApJ, 724, 111, doi: 10.1088/0004-637X/724/1/111

Strang, G. 1968, SIAM J. Numerical Analysis, 5, 506

Zhang, W., Almgren, A., Beckner, V., et al. 2019, Journal of Open Source Software, 4, 1370, doi: 10.21105/joss.01370

Zingale, M., Katz, M. P., Bell, J. B., et al. 2019, arXiv e-prints, arXiv:1908.03661. https://arxiv.org/abs/1908.03661