

Notes on semi-implicit time stepping, blended models, etc.

T. Benacchio, MetOffice, Exeter, UK

R. Klein, Mathematics & Informatics, Freie Universität Berlin

January 28, 2018

ToDo

January 19, 2018 (additions of January 25, 2018):

- get compressible version going as the long-wave hydrostatic GW test might involve a compressibility influence
(Set aside as long as I have no psinc version really running.)
- re-implement Runge-Kutta time integrator for the explicit part
(Set aside.)
- Move 1st projection to the half-time level of the second-order RK-scheme
(Will do so now, but keep the split/MUSCL scheme for now.)
- Can I invent a scheme that explicitly advects $\tilde{\chi}$ instead of full theta and thus allows me to control up-down symmetry better?
(Part of my next move.)
- Tested initial pressures with hydrostatic balance relative to background θ and relative to $\tilde{\theta}$: The latter gives slightly favorable results – less (though anyway small) wiggles, slightly higher wave amplitudes. But the solves become substantially more expensive.

January 25, 2018: New scheme

The key difficulty with my approach so far seems to be that the time semi-implicit integration is not consistently carried through the scheme. I am using the ingredients of the implicit midpoint rule in formulating and stabilizing the advective fluxes, yet I am so far not fully evaluating the implicit terms at the mid-time level. This would entail solving the pressure equation right there and accounting for the pressure gradients in correcting the advective fluxes also. This I am going to fix with the following strategy:

1. Split-step advection forward cycle, including separated treatment of $\tilde{\chi}$, with the contribution from the background stratification taken out and shifted to the flux correction part. Density transport is carried along as part of the compressible advection scheme. (See “Options” below.)
2. Advective flux correction including a correction that takes the split-step advection fluxes out and replaces them with fluxes computed afresh from the data obtained after step 1. Only now will the advection of the background be handled and added to $\tilde{\chi}$.

3. All fluxes, in particular the density flux, should be kept from the split-step prediction and then fully corrected to reflect the mid time level state. The density flux, in particular, has to be composed of the background and perturbation contributions associated with $\bar{\chi}$ and $\tilde{\chi}$.
4. Implicit momentum correction corresponding to “second projection” in our previous work. (See “Options”.)
5. Concluding advection split step.

Options:

- Options for compressible advection:
 1. Standard procedure based on $(\rho, \rho \mathbf{v}, P)$. The split into an additional transport scheme for $\tilde{\rho} = (P\tilde{\chi})$ may introduce a breach of mass continuity if ρ is replaced with the result of the decomposed step after the fact. It might be sufficient, though, to just use the split-version as an auxiliary means of formulating balanced fluxes.
 2. Switch to formulating advection in terms of $(P\tilde{\chi}, P\mathbf{v}, P)$. In this approach, we likely don’t even have to invoke an additional variable.
 3. Find a way to post-correct the density flux after solving the cell-centered correction equation so as to build a meaningful full density flux (hopefully without having to redo the upwinding).
- Options for gravity predictor
 1. Keep well-balanced explicit gravity computation as is prior to this development.
 2. Evaluate gravity from $\tilde{\chi}$.
 3. Take gravity out of advection entirely in split-fashion, moving it to after advection is completed. (EULAG variant)
- Options for second projection / balanced momentum update
 1. Treat implicit advection of background stratification as auxiliary calculation.
 2. Use “single projection only” approach developed earlier and implemented under “NODAL_PROJECTION_ONLY” in RKLM code variant.

ToDo-List: (not done yet: ○; done: ●)

- Introduce advection of density perturbation in `rhoX`-Variables.
- Make advective fluxes available inside the split step advection routines.
 - Access them everywhere to represent the advecting velocities.
 - In the reconstruction.
 - In the flux function.
 - Switch off gravity source term entirely in the advection routines.
 - Introduce entire gravity source term as part of the pair of projections.
 - apply half time step cell center gravity

- recompute fluxes
 - correct fluxes and post-correct all advective contributions
 - runs second pressure solve towards half time level
 - apply resulting source term another time to get to new time level
- Move projections in between the advection split steps.
 - Make sure second split sequence works with advective fluxes determined by the first projection.
 - Make sure second split sequence includes vertical advection of $\bar{\chi}$.
 - Make sure the second split sequence does not advect $\bar{\chi}$ for the $\tilde{\chi}$ variable.
 - Compare results of full density advection with that of perturbation density / $\tilde{\chi}$ advection
 - In the second half step, which quantities am I to advect? Those fully updated to the new time level or those strictly computed for the half time level? In fact, depending on how the details are done, I can recover Piotr's time integration strategy:
 - Euler forward advection over $\Delta t/2$
 - Euler implicit pressure/gravity for the advective fluxes over $\Delta t/2$ (advective fluxes done)
 - Euler explicit pressure/gravity for centered data from old time level over $\Delta t/2$
 - Second order advection of resulting fields over Δt based on previously estimated advective fluxes
 - Euler implicit pressure/gravity for centered data from the result over $\Delta t/2$
 - Check influence of how advecting velocity for MUSCL-update is computed from advecting fluxes in `recovery_gravity()`. Is division by the local `Sol->rhoY[i]` OK?

Other clean-up operations

- Move output-related memory flipping into the output routine.

1 Introduction

Previous attempts at implementing the semi-implicit time discretization strategy as explicated in T. Benacchio's thesis have not allowed us to go to time steps larger than $N\Delta t$, where N is the Brunt-Väisälä frequency and Δt the time step. It seems that it is necessary to split advection of potential temperature, θ , into that of its large-scale horizontal mean, $\bar{\theta}$, and deviations, $\tilde{\theta}$, from it. Advection of the former becomes part of the (quasi-)linearized implicit system and that of the latter remains part of the explicit advection step. This corresponds to the usual practice of subtracting out a background profile of θ at the differential equation level. A dynamically evolving and horizontally non-homogeneous background can be thought of in terms of a multiscale/multigrid strategy, but we leave this for later.

The time stepping strategy to be described here follows closely that described by P. Smolarkiewicz in a series of papers, see, e.g., (Smolarkiewicz et al. 2014, Smolarkiewicz & Margolin 1998).

2 Getting rid of the first (MAC) projection

2.1 Node- and cell-centered divergences

Currently, our scheme invokes two sequential projections / Helmholtz-solves to realize consistent divergence of the advective fluxes $P\mathbf{v}$. The first fixes the actual flux divergence employed in discretizing the advection terms, the second makes sure that the cell centered velocities are compatible with the respective constraint as well.

Two elliptic solves are expensive, so the question naturally arises as to whether there is a way to eliminate one of them. Here is a suggestion for how to achieve this without losing second-order accuracy. This suggestion is partially inspired by the time stepping and spatial discretization strategy of Piotr Smolarkiewicz' EULAG code (see, e.g., Smolarkiewicz & Margolin 1998).

The key observation can be explained on the basis of Fig. 1. The “second projection” in the

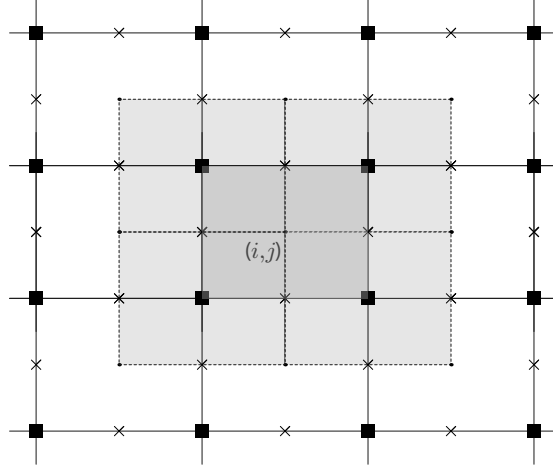


Figure 1: Arrangement of primary and dual cells surrounding a reference cell (i, j) on a cartesian 2D mesh.

current setting involves a discrete divergence for each node (filled squares in Fig. 1), *i.e.*, for some vector field $\mathbf{v} = (u, v)^t$ we have

$$\widetilde{(\nabla \cdot \mathbf{v})}_{i+\frac{1}{2}, j+\frac{1}{2}} = \frac{(u_{i+1, j} + u_{i+1, j+1}) - (u_{i, j} + u_{i, j+1})}{2\Delta x} + \frac{(v_{i, j+1} + v_{i+1, j+1}) - (v_{i+1, j} + v_{i, j})}{2\Delta y}. \quad (1)$$

Now we observe that averaging these divergences over the four dual cells that overlap with cell (i, j) , we obtain a valid divergence discretization for that cell,

$$\frac{1}{4} \left(\widetilde{(\nabla \cdot \mathbf{v})}_{i+\frac{1}{2}, j+\frac{1}{2}} + \widetilde{(\nabla \cdot \mathbf{v})}_{i-\frac{1}{2}, j+\frac{1}{2}} + \widetilde{(\nabla \cdot \mathbf{v})}_{i-\frac{1}{2}, j-\frac{1}{2}} + \widetilde{(\nabla \cdot \mathbf{v})}_{i+\frac{1}{2}, j-\frac{1}{2}} \right) = \overline{(\nabla \cdot \mathbf{v})}_{i, j} \quad (2)$$

where

$$\overline{(\nabla \cdot \mathbf{v})}_{i, j} = \frac{\bar{u}_{i+\frac{1}{2}, j} - \bar{u}_{i-\frac{1}{2}, j}}{\Delta x} + \frac{\bar{v}_{i, j+\frac{1}{2}} - \bar{v}_{i, j-\frac{1}{2}}}{\Delta y} \quad (3)$$

with the averaged primary cell interface velocities

$$\begin{aligned}\bar{u}_{i+\frac{1}{2},j} &= \frac{1}{8} \left([u_{i+1,j+1} + 2u_{i+1,j} + u_{i+1,j-1}] + [u_{i,j+1} + 2u_{i,j} + u_{i,j-1}] \right) \\ \bar{v}_{i,j+\frac{1}{2}} &= \frac{1}{8} \left([v_{i+1,j+1} + 2v_{i,j+1} + v_{i-1,j+1}] + [v_{i+1,j} + 2v_{i,j} + v_{i-1,j}] \right).\end{aligned}\tag{4}$$

We conclude that from the dual cell discrete divergence we can readily construct a compatible primary cell discrete divergence in conseration form. This is the key in replacing the MAC-projection of the scheme.

2.2 Outline of modified divergence control

The advecting fluxes $(P\mathbf{v})^{n+\frac{1}{2}}$ should ideally be determined using the implicit trapezoidal rule so that

$$(P\mathbf{v})^{n+\frac{1}{2}} = \frac{1}{2} \left((P\mathbf{v})^n + (P\mathbf{v})^{n+1} \right).\tag{5}$$

1. Store the cellface advective fluxes at time level n based on (4) in two separate fields. The first is to keep this quantity, the second is to be modified in the course of a time step to bring it to the new time level. Let us denote these fields by $(P\mathbf{v})^n$ and $(P\mathbf{v})^{n+\xi}$
2. Run the explicit predictor as before with the only exception of how the advective fluxes are determined in the flux function. Instead of directly letting, e.g., in the first x -step,

$$(Pu)_{i+\frac{1}{2},j}^{n+\xi_1} = \frac{1}{2} \left((Pu)_{i+\frac{1}{2},j}^{n+\xi_1} + (Pu)_{[i+1]^{-},j}^{n+\xi_1} \right)\tag{6}$$

with $[\cdot]_{i+\frac{1}{2},j}$ denoting reconstructed states at the split-step half time level at the right edge of primary cell (i, j) , we let

$$(Pu)_{i+\frac{1}{2},j}^{n+\xi_{m+1}} = (Pu)_{i+\frac{1}{2},j}^{n+\xi_m} + \widetilde{\Delta(Pu)_{i+\frac{1}{2},j}^{n+\xi_{m+1}}}\tag{7}$$

with the increment obtained from the local MUSCL-type reconstruction as

$$\begin{aligned}\widetilde{\Delta(Pu)_{i+\frac{1}{2},j}^{n+\xi_{m+1}}} &= \frac{1}{4} \left((Pu)_{i+,j} + (Pu)_{i-,j} + (Pu)_{[i+1]^{-},j} + (Pu)_{[i+1]^{+},j} \right)^{n+\xi_{m+1}} \\ &\quad - \frac{1}{2} \left((Pu)_{i,j} + (Pu)_{[i+1],j} \right)^{n+\xi_m}\end{aligned}\tag{8}$$

3. Once the predictor is done the advective fluxes in the second auxiliary field have been updated provisionally using the explicit scheme to time level $t^{n+\frac{1}{2}}$, which are labelled

$$(Pu)_{i+\frac{1}{2},j}^{n+\frac{1}{2},\nu}, \quad (Pu)_{i,j+\frac{1}{2}}^{n+\frac{1}{2},\nu}.\tag{9}$$

To start the divergence correction iterations, let $\nu = 0$.

4. Next we want a divergence-controlled velocity field at t^{n+1} . To this end, run the second projection for the cell-centered fields. From the corrected $(P\mathbf{v})_{i,j}^{n+1,\nu}$ field, compute advecting fluxes at the cell faces using (4). This generates the fields

$$(Pu)_{i+\frac{1}{2},j}^{n+1,\nu}, \quad (Pu)_{i,j+\frac{1}{2}}^{n+1,\nu}.\tag{10}$$

5. From these, and the old time level advective fluxes we obtain the next iterate for the half-time advective fluxes

$$(P\mathbf{v})^{n+\frac{1}{2},\nu+1} = \frac{1}{2} ((P\mathbf{v})^{n+1,\nu} + (P\mathbf{v})^n) \quad (11)$$

6. The ν th correction cycle is completed by running the advection correction based on the advective flux increments

$$\Delta(P\mathbf{v})^{n+\frac{1}{2},\nu} = (P\mathbf{v})^{n+\frac{1}{2},\nu+1} - (P\mathbf{v})^{n+\frac{1}{2},\nu}. \quad (12)$$

7. Check convergence and either continue the iteration or stop.

8.

9. Run the scheme as before up until the first projection would be invoked, except that the advecting fluxes are now computed using the averaging formulae in (4). Also, the effective advective fluxes over the entire advection cycle (be it computed via OpSplit or RK) are to be monitored. This yields advective fluxes $(Pu)_{i+\frac{1}{2},j}^{\nu,n+\frac{1}{2}}$ and $(Pv)_{i,j+\frac{1}{2}}^{\nu,n+\frac{1}{2}}$ for $\nu = 0$.

10. Run the node-centered projection to obtain the next iterate of the divergence corrected cell-centered velocities. Compute first iterate target cell interface advective fluxes using (4), $(Pu)_{i+\frac{1}{2},j}^{\nu+1,n+\frac{1}{2}}$ and $(Pv)_{i,j+\frac{1}{2}}^{\nu+1,n+\frac{1}{2}}$.

11. Correct advection results based on the advecting flux differences $(Pu)_{i+\frac{1}{2},j}^{\nu+1,n+\frac{1}{2}} - (Pu)_{i+\frac{1}{2},j}^{\nu,n+\frac{1}{2}}$ and $(Pv)_{i,j+\frac{1}{2}}^{\nu+1,n+\frac{1}{2}} - (Pv)_{i,j+\frac{1}{2}}^{\nu,n+\frac{1}{2}}$.

12. If $\| (P\mathbf{v})^{\nu+1,n+\frac{1}{2}} - (P\mathbf{v})^{\nu,n+\frac{1}{2}} \| \geq \text{tol}$ return to step 10, else we are done.

3 The time stepping strategy

Let U denote our vector of unknowns, which are generally conserved quantities in the standard sense, except maybe for the energy variable. Here we use $P = \rho\theta$ for energy as in (Benacchio et al. 2014, Klein 2009), where ρ is the density. Thus,

$$U = \begin{pmatrix} \rho \\ \rho\mathbf{v} \\ P \end{pmatrix} \quad (13)$$

where \mathbf{v} is the velocity. These unknowns depend on time t , horizontal position $\mathbf{x} = (x, y)$ and vertical position z , *i.e.*, we have the signature $U(t, \mathbf{x}, z)$.

We cover compressible and pseudo-incompressible flow following (Durran 1989) through the blended evolution equation (Benacchio et al. 2014)

$$D_\alpha U_t = \mathcal{A}[U] + \mathcal{R}[U] \quad (14)$$

with

$$D_\alpha = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \text{Id} & 0 \\ 0 & 0 & \alpha \end{pmatrix}, \quad \mathcal{A}[\rho\psi] = -\nabla \cdot (P\mathbf{v}[\psi/\theta]), \quad \mathcal{R}[U] = \begin{pmatrix} 0 \\ \nabla p' - g\mathbf{k} \left(\rho + \frac{1-\alpha}{c_0^2} p' \right) \\ \rho S(\theta) \end{pmatrix}. \quad (15)$$

Here $\alpha \in [0, 1]$ and the system describes fully compressible flow for $\alpha = 1$ and pseudo-incompressible flow for $\alpha = 0$. The quantities \bar{c}_0 and p' are defined once a horizontally homogeneous background stratification $(P_0, \rho_0, \theta_0)(z)$ of the thermodynamic variables has been defined. To this end we pick, e.g., a background potential temperature distribution $\theta_0(z)$ and then obtain P_0 and ρ_0 assuming hydrostatic balance, *i.e.*,

$$\frac{dp_0}{dz} = -\rho_0 g, \quad p_0 = P_0^{1/\gamma}, \quad \rho_0 = \frac{P_0}{\theta_0}, \quad P_0(0) = 1, \quad (16)$$

where γ is the isentropic exponent which is assumed to be constant here. Then we have

$$c_0^2 = \gamma \frac{p_0}{\rho_0}, \quad \text{and} \quad p' = P^{1/\gamma} - P_0^{1/\gamma} \quad (\alpha \neq 0). \quad (17)$$

For the limiting pseudo-incompressible case, $\alpha = 0$, p' cannot be obtained from the primary thermodynamic variables but satisfies an elliptic equation that guarantees compliance of the velocity field with the pseudo-incompressible divergence constraint $\nabla \cdot (P_0 \mathbf{v}) = \rho S(\theta)$.

then a second-order time stepping strategy reads

$$D_\alpha U^{n+1} = \tilde{\mathcal{A}} \left[D_\alpha U^n + \frac{\Delta t}{2} \mathcal{R}[U^n] \right] + \mathcal{R}[U^{n+1}]. \quad (18)$$

4 Semi-implicit gravity

To be solved in a (linearized) implicit, backward Euler step over $\Delta t/2$,

$$w_t = -\frac{\theta}{\Gamma} (\pi_z + \Gamma g \chi) \quad (19)$$

$$\chi_t = -w \frac{d\bar{\chi}}{dz} \quad (20)$$

where $\theta = 1/\chi$ is frozen in the linearization. The current state is w^*, χ^*, π^* .

Then, an implicit step over time step Δt reads

$$w^{n+1} - w^* = \delta w = -\Delta t \frac{\theta}{\Gamma} ([\pi^* + \delta\pi]_z + \Gamma g[\chi^* + \delta\chi]) \quad (21)$$

$$\chi^{n+1} - \chi^* = \delta\chi = -\Delta t (w^* + \delta w) \frac{d\bar{\chi}}{dz}. \quad (22)$$

Reorder explicit and implicit contributions, replace $\delta\chi$ in (21) and solve for δw ,

$$w^{n+1} - w^* = \delta w = -\Delta t \frac{\theta}{\Gamma} (\pi_z^* + \Gamma g \chi^*) - \Delta t \frac{\theta}{\Gamma} (\delta\pi_z + \Gamma g \delta\chi) \quad (23)$$

$$= -\Delta t \frac{\theta}{\Gamma} (\pi_z^* + \Gamma g \chi^*) - \Delta t \frac{\theta}{\Gamma} \left(\delta\pi_z - \Delta t \Gamma g (w^* + \delta w) \frac{d\bar{\chi}}{dz} \right) \quad (24)$$

$$\delta w \left(1 - (\Delta t)^2 g \theta \frac{d\bar{\chi}}{dz} \right) = -\Delta t \frac{\theta}{\Gamma} (\pi_z^* + \Gamma g \chi^*) + w^* (\Delta t)^2 g \theta \frac{d\bar{\chi}}{dz} - \Delta t \frac{\theta}{\Gamma} \delta\pi_z. \quad (25)$$

Letting

$$N_{sc}^2 = -(\Delta t)^2 g \theta \frac{d\bar{\chi}}{dz} \quad (26)$$

we have

$$\delta w = \frac{1}{1 + N_{sc}^2} \left(-\Delta t \frac{\theta}{\Gamma} (\pi_z^* + \Gamma g \chi^*) - w^* N_{sc}^2 \right) - \Delta t \frac{\theta/\Gamma}{1 + N_{sc}^2} \delta\pi_z \quad (27)$$

and the update formular for χ subsequently follows from (22).

In (28) we use the abbreviation $\Gamma g \chi^* = -\pi_z^{\text{hy}}$ to obtain

$$\delta w = \frac{1}{1 + N_{sc}^2} \left(-\Delta t \frac{\theta}{\Gamma} \left(\pi_z^* - \pi_z^{\text{hy}} \right) - w^* N_{sc}^2 \right) - \Delta t \frac{\theta/\Gamma}{1 + N_{sc}^2} \delta \pi_z \quad (28)$$

$$\delta w = \delta^{\text{expl}} w + \delta^\pi w \quad (29)$$

$$\delta \chi = -\Delta t (w^* + \delta^{\text{expl}} w) \frac{d\bar{\chi}}{dz} + \frac{1}{\Gamma g} \frac{1}{1 + N_{sc}^2} (\Delta t)^2 g \theta \frac{d\bar{\chi}}{dz} \delta \pi_z \quad (30)$$

$$\delta \chi = -\Delta t (w^* + \delta^{\text{expl}} w) \frac{d\bar{\chi}}{dz} + \frac{1}{\Gamma g} \frac{N_{sc}^2}{1 + N_{sc}^2} \delta \pi_z \quad (31)$$

It is clear how to implement this in a split fashion outside of the advection routines. It needs to be implemented also, however, in the flux calculations, and it is less clear how that could be achieved.

References

- Benacchio T, O'Neill W, Klein R. 2014. A blended soundproof-to-compressible model for atmospheric dynamics. *Mon. Wea. Rev.* 142:4416–4438
- Durran DR. 1989. Improving the anelastic approximation. *Journal of Atmosphere Sciences* 46:1453–1461
- Klein R. 2009. Asymptotics, Structure, and Integration of Sound-Proof Atmospheric flow Equations. *Theoretical and Computational Fluid Dynamics* 23:161–195
- Smolarkiewicz P, Kühnlein C, Wedi N. 2014. A consistent framework for discrete integrations of soundproof and compressible PDEs of atmospheric dynamics. *J. Comput. Phys.* 263:185–205
- Smolarkiewicz PK, Margolin LG. 1998. Mpdata: A finite-difference solver for geophysical flows. *J. Comp. Phys.* 140:459–480