Implicit-Explicit Time Integration Methods for Non-hydrostatic Atmospheric Models

David J. Gardner, LLNL

Jorge E. Guerra\textsuperscript{1}, Francois P. Hamon\textsuperscript{2}, Daniel R. Reynolds\textsuperscript{3}, Paul A. Ullrich\textsuperscript{1}, Carol S. Woodward\textsuperscript{4}

\textsuperscript{1}UC Davis, \textsuperscript{2}LBNL, \textsuperscript{3}SMU, \textsuperscript{4}LLNL

NCAR Workshop on Multiscale Geosicence Numerics

May 16, 2017
Outline

- Motivation
- Tempest non-hydrostatic formulation
- Additive Runge-Kutta methods, IMEX splittings, and solvers
- Gravity wave test and results
- Baroclinic wave test and results
- Conclusions and Future Work
Non-hydrostatic Atmospheric Models

- The Accelerated Climate Modeling for Energy (ACME) Project is designed to perform coupled climate simulations at high resolution on current and next generation DOE leadership class supercomputers.

- To accurately simulate atmospheric phenomena at very high resolution requires moving from a hydrostatic to a non-hydrostatic model.

- Non-hydrostatic dynamical cores build on the fully compressible Navier Stokes equations support flows containing acoustic (sound) waves.

- Acoustic waves are of little significance in climate modeling and impose an overly restrictive limit on time step sizes.

- To overcome this limitation, non-hydrostatic models utilize split-explicit, implicit-explicit, or fully implicit time integration.
Non-hydrostatic Formulation

- Five governing equations in an arbitrary coordinate system:
  \[
  \frac{\partial u_\alpha}{\partial t} = - \frac{\partial}{\partial \alpha} (K + \Phi) - \theta \frac{\partial \Pi}{\partial \alpha} + (\eta \times u)_\alpha
  \]
  \[
  \frac{\partial u_\beta}{\partial t} = - \frac{\partial}{\partial \beta} (K + \Phi) - \theta \frac{\partial \Pi}{\partial \beta} + (\eta \times u)_\beta
  \]
  \[
  \frac{\partial u_\xi}{\partial t} = - \frac{\partial}{\partial \xi} (K + \Phi) - \theta \frac{\partial \Pi}{\partial \xi} + (\eta \times u)_\xi
  \]

- Thermodynamics:
  \[
  \frac{\partial \theta}{\partial t} = -u_\alpha \frac{\partial \theta}{\partial \alpha} - u_\beta \frac{\partial \theta}{\partial \beta} - u_\xi \frac{\partial \theta}{\partial \xi}
  \]

- Continuity:
  \[
  \frac{\partial \rho}{\partial t} = - \frac{1}{J} \frac{\partial}{\partial \alpha} (J \rho u_\alpha) - \frac{1}{J} \frac{\partial}{\partial \beta} (J \rho u_\beta) - \frac{1}{J} \frac{\partial}{\partial \xi} (J \rho u_\xi)
  \]

- The spatial terms are discretized horizontally using 4th order spectral elements and vertically with the staggered nodal finite element method.
Additive Runge-Kutta Methods

- Consider the initial value problem \( \frac{dy}{dt} = f_E(t, y) + f_I(t, y) \quad y(t_0) = y_0 \)

- ARK methods combine an explicit RK method for the non-stiff dynamics with a diagonally implicit RK method for the stiff dynamics:

  \[
  z_i = y_{n-1} + h_n \sum_{j=1}^{i-1} a_{i,j}^E f_E(t_{n,j}, z_j) + h_n \sum_{j=1}^{i} a_{i,j}^I f_I(t_{n,j}, z_j), \quad i = 1, \ldots, s,
  \]

  \[
  y_n = y_{n-1} + h_n \sum_{i=1}^{s} \left( b_{i}^E f_E(t_{n,i}, z_i) + b_{i}^I f_I(t_{n,i}, z_i) \right),
  \]

  \[
  \tilde{y}_n = y_{n-1} + h_n \sum_{i=1}^{s} \left( \tilde{b}_{i}^E f_E(t_{n,i}, z_i) + \tilde{b}_{i}^I f_I(t_{n,i}, z_i) \right).
  \]

- Stage solutions are computed by solving a nonlinear residual equation

  \[
  G(z_i) \equiv z_i - h_n A_{i,i}^I f_I(t_{n,i}, z_i) - a_i = 0 \quad a_i \equiv y_{n-1} + h_n \sum_{j=1}^{i-1} A_{i,j}^E f_E(t_{n,j}, z_j) + A_{i,j}^I f_I(t_{n,j}, z_j)
  \]

- ARK methods are implemented by interfacing with the SUNDIALS ARKode package
SUNDIALS

- **SU**ite of **Nonlinear and DIfferential ALgebraic Equation Solvers**
- **CVODE / CVODES** – Variable order, variable step, multistep methods for ODEs
- **ARKode** – Variable step, explicit, implicit, and IMEX multistage methods for ODEs
- **IDA / IDAS** – Variable order, variable step BDF methods for DAEs
- **KINSOL** – Nonlinear solvers (Newton and fixed point)
- Written in C with interfaces to Fortran
- Designed to be incorporated into existing codes and can work with user supplied data structures using a generic vector API
- Supplied with serial, MPI, OpenMP, pThreads, PETSc, and hypre optional vectors
- New features in development as part of the Exascale Computing Project
- Freely available under BSD license: [computation.llnl.gov/projects/sundials](http://computation.llnl.gov/projects/sundials)
HEVI: Horizontally Explicit Vertically Implicit

- **4 HEVI formulations:**
  - **HEVI-A:** all vertical dynamics are implicit (except vertical advection of $u_{\alpha}$ and $u_{\beta}$)
  - **HEVI-B:** explicit vertical velocity advection
  - **HEVI-C:** explicit thermodynamic advection
  - **HEVI-D:** explicit vertical velocity and thermodynamic advection

\[
\begin{align*}
\frac{\partial u_{\alpha}}{\partial t} &= -\frac{\partial}{\partial \alpha} (K + \Phi) - \theta \frac{\partial \Pi}{\partial \alpha} + (\eta \times u)_{\alpha} \\
\frac{\partial u_{\beta}}{\partial t} &= -\frac{\partial}{\partial \beta} (K + \Phi) - \theta \frac{\partial \Pi}{\partial \beta} + (\eta \times u)_{\beta} \\
\frac{\partial u_{\xi}}{\partial t} &= \frac{\partial K}{\partial \xi} + u_{\alpha} \frac{\partial u_{\alpha}}{\partial \xi} + u_{\beta} \frac{\partial u_{\beta}}{\partial \xi} - \frac{\partial \Phi}{\partial \xi} - \theta \frac{\partial \Pi}{\partial \xi} - u_{\alpha} \frac{\partial u_{\xi}}{\partial \alpha} - u_{\beta} \frac{\partial u_{\xi}}{\partial \beta} \\
\frac{\partial \theta}{\partial t} &= -u_{\alpha} \frac{\partial \theta}{\partial \alpha} - u_{\beta} \frac{\partial \theta}{\partial \beta} - u_{\xi} \frac{\partial \theta}{\partial \xi} \\
\frac{\partial \rho}{\partial t} &= -\frac{1}{J} \frac{\partial}{\partial \alpha} (J \rho u^\alpha) - \frac{1}{J} \frac{\partial}{\partial \beta} (J \rho u^\beta) - \frac{1}{J} \frac{\partial}{\partial \xi} (J \rho u^\xi)
\end{align*}
\]
IMEX: HEVI + Implicit Horizontal Dynamics

- 2 IMEX formulations (HEVI-A (red) + blue terms):
  - **IMEX-A**: Fully implicit density equation
  - **IMEX-B**: Fully implicit density and thermodynamics and implicit Exner pressure

\[ \frac{\partial u_\alpha}{\partial t} = -\frac{\partial}{\partial \alpha} (K + \Phi) - \theta \frac{\partial \Pi}{\partial \alpha} + (\eta \times u)_\alpha \]

\[ \frac{\partial u_\beta}{\partial t} = -\frac{\partial}{\partial \beta} (K + \Phi) - \theta \frac{\partial \Pi}{\partial \beta} + (\eta \times u)_\beta \]

\[ \frac{\partial u_\xi}{\partial t} = -\frac{\partial K}{\partial \xi} + u_\alpha \frac{\partial u_\alpha}{\partial \xi} + u_\beta \frac{\partial u_\beta}{\partial \xi} - \theta \frac{\partial \Pi}{\partial \xi} - u_\alpha \frac{\partial u_\xi}{\partial \alpha} - u_\beta \frac{\partial u_\xi}{\partial \beta} \]

\[ \frac{\partial \theta}{\partial t} = -u_\alpha \frac{\partial \theta}{\partial \alpha} - u_\beta \frac{\partial \theta}{\partial \beta} - u_\xi \frac{\partial \theta}{\partial \xi} \]

\[ \frac{\partial \rho}{\partial t} = -\frac{1}{J} \frac{\partial}{\partial \alpha} (J \rho u^\alpha) - \frac{1}{J} \frac{\partial}{\partial \beta} (J \rho u^\beta) - \frac{1}{J} \frac{\partial}{\partial \xi} (J \rho u^\xi) \]
ARK Methods

- Ascher, Ruuth, and Spiteri (1997):
  - 2nd order: ARS232, ARS222, ARS233
  - 3rd order: ARS343, ARS443

- Kennedy and Carpenter (2003):
  - 3rd order: ARK324
  - 4th order: ARK436
  - 5th order: ARK548

- Giraldo et al. (2013):
  - 2nd order: ARK232

- Pareschi and Russo (2005):
  - 2nd order: SSP2(222), SSP2(332)a, SSP3(332)
  - 3rd order: SSP3(433)

  - 2nd order: SSP2(332)b, lpm1, lpm2, lpum, lspum
  - 3rd order: SSP3(333)

- A-Stable: 18 methods
- L-Stable: 17 methods
- B-Stable: 2 methods
- Stiffly accurate: 10 methods
- Second order implicit stage accuracy: 4 methods
- Strong Stability Preserving:
  - S – explicit stability region contains part of Im. axis
  - P – positive amplification factor
  - U – uniform convergence
  - M – non trivial region of absolute monotonicity
Nonlinear and Linear Solvers

- ARK methods require solving multiple nonlinear systems in each time step
  \[ G(z_i) \equiv z_i - h_n A_{i,i}^I f_I(t_n^I, z_i) - a_i = 0 \]

- HEVI splittings leverage the 2D parallel decomposition for highly efficient solves
  - Since each processor owns entire columns of the domain no inter-processor communication is necessary when computing stage solutions

- IMEX splittings require solving a globally coupled systems of equations
  - Utilize a Newton-Krylov approach using the HEVI-A column solver as a preconditioner

<table>
<thead>
<tr>
<th></th>
<th>HEVI</th>
<th>IMEX</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nonlinear Solve</td>
<td>Linearly Implicit or Newton</td>
<td>Linearly Implicit or Newton</td>
</tr>
<tr>
<td>Linear Solve</td>
<td>Column-wise direct Solve</td>
<td>GMRES</td>
</tr>
<tr>
<td>Preconditioner</td>
<td>--</td>
<td>Column-wise direct solve</td>
</tr>
</tbody>
</table>

- Linearly implicit solves (one Newton iteration) with a column-wise direct linear solver do not require setting any solver tolerances
A initially balanced state on a reduced size Earth (1/125) is given a potential temperature perturbation giving rise to gravity waves

Test setup:
- 1 hour simulation
- Horizontal resolution: NE20 (2,400 elements), 4th order
- Vertical resolution: 10 levels, 1st order
- Tolerances: Relative = 1e-4, Absolute = 1e-4
- 96 MPI tasks (6 nodes on Cab Linux cluster at LLNL)
HEVI Results

- The RMS error in the solution after 1 hour is computed using a fully explicit reference solution with a time step of 0.001s

- Results for HEVI splittings differ due to the treatment of vertical velocity advection
  - HEVI-A/C (implicit vertical velocity) vs HEVI-B/D (explicit vertical velocity advection)

- Multiple Newton iterations do not improve accuracy or stability in this case and increase the runtime by 20% - 50% with most splittings/methods

- Second order methods fall into two groups based on accuracy and have similar efficiency for all splittings
  - Optimized SSP methods of Higuera (2014), represented by SSP2(332)Ispum
  - All other second order methods, represented by ARS232

- Most third order methods generally give similar levels of accuracy across the splittings options
  - Less accuracy ARS233, typical accuracy ARS343, higher accuracy SSP3(433)
HEVI-A: Convergence (left) and Efficiency (right)

2\textsuperscript{nd}, 3\textsuperscript{rd} and 4\textsuperscript{th} order methods achieve theoretical orders of convergence.
Higher order methods are more accurate and generally more efficient but do not enable larger step sizes.
Errors plateau at $\sim 2.7 \times 10^{-3}$ for steps size of 0.1s or less and the methods do not achieve the expected convergence rates.
HEVI-B: Convergence (left) and Efficiency (right)

Explicit velocity advection is like solving Burger’s equation with Dirichlet boundary conditions which admits a discontinuity.
IMEX-A: Convergence (left) and Efficiency (right)

Treating density fully implicitly does not increase the max step size over HEVI formulations.
An imbalance in the coupling maybe causing the reduction in convergence rate for higher order methods.
Adding more implicitness increases the maximum stable step size from 2s to 8s for all methods.
Expected orders of convergence are restored but using multiple Newton iterations requires smaller step sizes for convergence.
Baroclinic Wave Test

- A balanced initial condition is perturbed to trigger the development of a wave over approximately 10 days

Test setup
- 30 day simulation
- Horizontal resolution: 2,400 elements, 4th order
- Vertical resolution: 30 levels, 1st order
- Tolerances: Relative = 1e-4, Absolute = 1e-4
- 96 MPI tasks (6 nodes on Cab Linux cluster at LLNL)
### Max Stable Step Size (Linearly Implicit / Newton)

<table>
<thead>
<tr>
<th>Method</th>
<th>HEVI - A / B</th>
<th>HEVI – C / D</th>
<th>IMEX - A</th>
<th>IMEX - B</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARS 232</td>
<td>200</td>
<td>120</td>
<td>450</td>
<td>550</td>
</tr>
<tr>
<td>SSP2 332 Ispum</td>
<td>300</td>
<td>100</td>
<td>450</td>
<td>550</td>
</tr>
<tr>
<td>ARS 343</td>
<td>450</td>
<td>384</td>
<td>300</td>
<td>550</td>
</tr>
<tr>
<td>SSP3 433</td>
<td>200</td>
<td>120</td>
<td>450</td>
<td>550</td>
</tr>
</tbody>
</table>

- Like in the gravity wave test, the maximum stable step size does not differ when the system is treated as linearly implicit or when iterating Newton to convergence.

- However in this test taking multiple Newton iterations can produce a better solution than a linearly implicit approach (following slides).

- Treating the thermodynamic equation fully explicitly reduces the largest stable step size for most methods but some are unaffected.

- Generally adding horizontally implicit terms increases stability and for some methods greatly improves solution quality (following slides).
For all HEVI splittings results using the largest possible step size produce a wide range of maximum vertical velocity values.
HEVI-A: Max Vertical Velocity at Max Step Size

For step sizes above 300s, iterating the nonlinear stage solutions to convergence can give lower velocity values (green solid vs green dashed)
SSP methods have vertical velocities 2.5x to 5x larger than other methods with either nonlinear solver approach.
Treating density fully implicitly greatly reduces the range of velocity values by bringing the SSP results more inline with those of other methods.
The additional implicitness also increases the maximum step size for most methods and the difference between a linearly implicit and fully implicit solve are more important.
Adding more implicit terms further increases the max step size to 550s for almost all methods.
Due to the larger step sizes, the range of velocity values increases and, like with the gravity wave test, the Newton solver is unable to converge to the given tolerance at the largest time step sizes.
To determine an acceptably accurate solution we define a range of velocity values using the results from fully explicit simulations with initial conditions perturbed by normally distributed random values.
Determining an Acceptable Solution

The 99% confidence interval for the mean value is given by the light red area.

The light purple area is 10% of the maximum deviation in the light red area and used to identify overly large initial transients in the solution.
Example of an Acceptable Solution with ARS343

While ARS343 is stable with a step size of 450s using the linearly implicit approach the velocity values are well outside of the confidence interval of the explicit solutions.
Iterating the stage systems to convergence gives an acceptable solution at 450s while the linearly implicit approach requires a reduced step size of 300s.
Acceptable Step Size (Linearly Implicit / Newton)

- Using the explicit solution confidence interval we find the approximate maximum step size that produces an acceptably accurate result

<table>
<thead>
<tr>
<th>Method</th>
<th>HEVI - A / B</th>
<th>HEVI – C / D</th>
<th>IMEX - A</th>
<th>IMEX - B</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARS 232</td>
<td>200</td>
<td>120</td>
<td>300</td>
<td></td>
</tr>
<tr>
<td>SSP2 332 Lspum</td>
<td>--</td>
<td>--</td>
<td>300</td>
<td></td>
</tr>
<tr>
<td>ARS 343</td>
<td>300</td>
<td></td>
<td>450</td>
<td>300</td>
</tr>
<tr>
<td>SSP3 433</td>
<td>--</td>
<td>--</td>
<td>450</td>
<td>432</td>
</tr>
</tbody>
</table>

- The linearly implicit approach generally requires a reduced step size of about 300s
- Iterating Newton to convergence gives good results at the largest stable step sizes
- Nearly all SSP methods did not have acceptable solutions with step sizes of 100s
- Treating more dynamics implicitly increases stability but are expensive (next slide)
Runtimes (Linearly Implicit / Newton)

- Runtimes for the Baroclinic wave test normalized by the fastest result, HEVI-C ARS343 with a linearly implicit solve and a step size of 320s.

<table>
<thead>
<tr>
<th>Method</th>
<th>HEVI - A / B</th>
<th>HEVI - C / D</th>
<th>IMEX - A</th>
<th>IMEX - B</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARS 232</td>
<td>1.20</td>
<td></td>
<td>1.62</td>
<td>1.95</td>
</tr>
<tr>
<td>SSP2 332 Ispum</td>
<td>--</td>
<td>--</td>
<td>2.43</td>
<td></td>
</tr>
<tr>
<td>ARS 343</td>
<td>1.08</td>
<td></td>
<td>1.02</td>
<td><strong>1.00</strong></td>
</tr>
<tr>
<td>SSP3 433</td>
<td>--</td>
<td>--</td>
<td>2.33</td>
<td></td>
</tr>
</tbody>
</table>

- If the difference in acceptable step size is sufficiently large then the Newton solver is a faster approach otherwise a linearly implicit solve is sufficient.

- Even with the minimal horizontal implicitness and effective preconditioning, IMEX-A is slower than a HEVI approach in most cases (note ars232 linearly implicit).

- Treating more horizontal terms implicitly does improve step sizes but the additional solver cost with less effective preconditioning outweighs the gains.
Baroclinic Wave Results

- Third order methods (ARS 343 and ARK 324) produce the fastest results
- If the difference in acceptable step size is sufficiently large, iterating Newton’s method to convergence can be more efficient
- SSP methods over estimate the vertical velocity at step sizes of 100s or more and require additional implicit dynamics for accurate results with large step sizes
- L-stability is an important property in a successful method
  — The only two methods that were unstable in all splitting were not L-stable

<table>
<thead>
<tr>
<th>Method</th>
<th>HEVI – Linearly Implicit</th>
<th>HEVI - Newton</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARS 343</td>
<td>300s</td>
<td></td>
</tr>
<tr>
<td>ARK 324</td>
<td>320s</td>
<td></td>
</tr>
<tr>
<td>ARS 232</td>
<td>200s</td>
<td></td>
</tr>
<tr>
<td>ARK 232</td>
<td>200s</td>
<td></td>
</tr>
</tbody>
</table>
Summary and Conclusions

- HEVI-A shows the best performance in the two test cases
  - There were no clear benefits, in terms of accuracy or efficiency, from treating some vertical dynamics explicitly

- Adding horizontally implicit terms increases stability and improves results with SSP methods is less efficient due to the cost of globally coupled solves

- The gravity wave test case produced a very different set of optimal methods than identified in the baroclinic wave test case
  - This is likely due to the increased nonlinearity in the baroclinic wave test and the reduced domain size in the gravity wave test

- Third order methods, in particular ARS 343 and ARK 324, were the best performing methods overall.
  - The second order ARS 232 and ARK 232 methods also performed well

- The choice of nonlinear solver approach is important to achieving the maximum time step sizes possible however, a linearly implicit approach can be more efficient with a smaller step size for accuracy
Current and Future Work

- Interfacing ARKode with the HOMME-NH dycore
  - Evaluate the best methods from these tests in the HOMME model

- DCMIP 2016 Test cases
  - Evaluate the splittings and integration methods in tests with simplified physics

- Vertically Lagrangian formulation
  - Produces better climatology and tracer transport

- Adaptive time stepping
  - ARKode allows for ARK methods with embeddings to estimate the temporal error and adjust the time step size as necessary to meet a given accuracy

- Rosenbrock methods
  - Given the success of the linearly implicit approach, Rosenbrock methods may give more accurate results with little additional work
Acknowledgements

Support for this work was provided by the Department of Energy, Office of Science SciDAC project
A Non-hydrostatic Variable Resolution Atmospheric Model in ACME

github.com/paullric/tempestmodel

computation.llnl.gov/projects/sundials