Application of Exponential Time Integration Methods in Numerical Weather Prediction Models

Stéphane Gaudreault

Recherche en prévision numérique atmosphérique
The Evolution of High Performance Computing (HPC)

- HPC is evolving rapidly: the best numerical algorithms on today’s supercomputer could be suboptimal in the future

- Are we ready for such architectures?
  - GEM will be improved
    - Better parallelism with OpenMPI
    - Improved semi-Lagrangian advection
    - Improved elliptic solvers

- Exponential time integrators
  - Promising approach to overcome accuracy drawbacks related to the large time-step choice while still correctly simulating all relevant wave dispersion relations. (Clancy and Pudykiewicz, 2013).
Variation of constants formula

• Consider a large scale stiff system of ODEs of the form

\[
\frac{d}{dt} u(t) = F(u(t)),
\]
\[
u(t_0) = u_0, \quad u, F(u) \in \mathbb{R}^N.
\]

• Solution given by the variation of constants formula

\[
u(t_n + \Delta t) = u_n + (e^{J\Delta t} - I) J^{-1} F_n + \int_{t_n}^{t_n+\Delta t} e^{J(t_n+\Delta t-t)} R(u(t)) \, dt
\]

\[
R(u(t)) = F(u(t)) - F(u(t_n)) - J(u(t) - u(t_n))
\]

\[
J = \frac{dF}{du}(u_n)
\]
Introduction of the $\phi$-functions

- For convenience, we rewrite the variation of constants formula in terms of the $\phi$-functions:

$$u(t_n + \Delta t) = u_n + \phi_1(J \Delta t)F_n \Delta t + \int_{t_n}^{t_n+\Delta t} e^{J(t_n+\Delta t-t)} R(u(t)) dt$$

where

$$\phi_0(A)v = e^A v, \quad \phi_p(A)v = \sum_{n=0}^{\infty} \frac{1}{(n + p)!} A^n v$$

- This is the starting point for numerical methods
Exponential Propagation Iterative (EPI) methods

\[ u(t_n + \Delta t) = u_n + \varphi_1(J \Delta t) F_n \Delta t + \int_{t_n}^{t_n+\Delta t} e^{J(t_n+\Delta t-t)} R(u(t)) \, dt \]

- EPI2

\[ u_{n+1} = u_n + \varphi_1(J_n \Delta t) F_n \Delta t \]

- EPI3

\[ u_{n+1} = u_n + \varphi_1(J_n \Delta t) F_n \Delta t + \frac{2}{3} \varphi_2(J_n \Delta t) \Delta t R_{n-1} \]

\[ R_{n-1} = F(u_{n-1}) - F(u_n) - J_n (u_{n-1} - u_n) \]

- EPI4 and EPI5 have also been developed.
  - Collaboration with Mayya Tokman, Valentin Dallerit, Tommaso Buvoli (UC Merced) and Dominik L. Michels (KAUST).
  - Siggraph paper in preparation.
Computation of the $\phi$-functions

- Key to efficiency
- Difficult problem
- Naïves methods are not practical
- Still the topic of a considerable research
- Our contribution: Krylov with Incomplete Orthogonalization Procedure Solver (KIOPS)

KIOPS in a nutshell

• We seek for the solution in a smaller space: the m-dimensional Krylov subspace

\[ K_m(A, v) = \text{span}\{v, Av, \ldots, A^{m-1}v\} \]

• Incomplete orthogonalization (IOM) method produce matrices \(H_m\) and \(V_m\) such that

\[ e^{\tau \tilde{A}} v \approx \|v\| \ V_m \ e^{\tau H_m} \ e_1 \]

• A new adaptive procedure has been developed.

\[ e^{\tau \tilde{A}} v = e^{(\tau_1+\tau_2+\ldots+\tau_k)\tilde{A}} v = e^{\tau_k \tilde{A}} (\ldots (e^{\tau_2 \tilde{A}} (e^{\tau_1 \tilde{A}} v))) \]

• The algorithm is completely matrix-free. Only a matvec (Jacobian*vector) routine is needed.
Comparison with the current state-of-the-art adaptive Krylov algorithm phiPM.

Phipm: http://www1.maths.leeds.ac.uk/~jitse/software.html
Shallow Water Model on the Yin-Yang Grid

\[
\frac{dh}{dt} = - \frac{\overline{u}}{a \cos \phi} \delta^a_{\lambda} h^* - \frac{\overline{v}}{a} \delta^a_{\phi} h^* - \frac{h^*}{a \cos \phi} \left( \delta^e_{\lambda} u + \delta^e_{\phi} v \cos \phi \right) \quad (1)
\]

\[
\frac{du}{dt} = - \frac{u}{a \cos \phi} \delta^a_{\lambda} u - \frac{\overline{v}}{a} \delta^a_{\phi} u + \left( f + u \frac{\tan \phi}{a} \right) \overline{v} - \frac{g}{a \cos \phi} \delta^e_{\lambda} h \quad (2)
\]

\[
\frac{dv}{dt} = - \frac{\overline{u}}{a \cos \phi} \delta^a_{\lambda} v - \frac{v}{a} \delta^a_{\phi} v - \left( f + \overline{u} \frac{\tan \phi}{a} \right) \overline{u} - \frac{g}{a} \delta^e_{\phi} h \quad (3)
\]

✔ Yin-Yang grid, Arakawa C, 5th-order finite difference
✔ No filter, no limiter, no diffusion
✔ MPI implementation in GEM
Zonal Flow Over an Isolated Mountain
15 Days Simulation at 1 degree resolution

spectral model
1080 × 540 lin. Gaussian grid (T539)
Timestep : 240 s
(Qaddouri et al. 2012)
The Non-hydrostatic Euler Equations

- Vertical coordinate based on height (cf presentations by Syed Z. Husain and Abdessamad Qaddouri)

- 5 dynamical variables:
  - Momentum \( \frac{\partial u^i}{\partial t} + u^j u^i_{,j} + 2\Gamma^i_{j0} u^j + \Gamma^i_{jk} u^j u^k = -h^{ij} \left( \theta_v \hat{\Pi}_{,j} + \psi_{,j} \right) \)
  - Continuity \( \frac{\partial \hat{\Pi}}{\partial t} + u^j \hat{\Pi}_{,j} = -\frac{R_d \hat{\Pi}}{c_v \sqrt{g}} (\sqrt{g} u^j)_{,j} \)
  - Thermodynamic \( \frac{\partial \theta_v}{\partial t} + u^j (\theta_v)_{,j} = 0 \)

(Charron et al. 2014)
Dry Bubble Convection Problem

The story so far ...

- Stable
- Precise
- High arithmetic intensity (local)
- Fast (work in progress)
- Performance degrades for very stiff problems
  - Large difference in grid-spacings in the horizontal and vertical directions.
Current work

- Development of partitioned schemes (collaboration with Mayya Tokman, UC Merced).

\[ \frac{du}{dt} = F_{\text{exponential}}(u) + F_{\text{implicit}}(u) + F_{\text{explicit}}(u) \]

- Comming up next!
Thanks!

- Code availability
  - GEM: https://gitlab.com/gem-ec/goas
  - KIOPS
    - MATLAB: https://gitlab.com/stephane.gaudreault/kiops
    - Julia: https://github.com/JuliaDiffEq/ExponentialUtilities.jl
    - Fortran 2008: available upon request