Characteristic-Based Flux Splitting for Implicit-Explicit Time Integration of Low-Mach Number Flows

Debojyoti Ghosh **Emil M. Constantinescu**

Mathematics & Computer Science
Argonne National Laboratory

13th U. S. National Congress on Computational Mechanics (USNCCM13)
San Diego, CA, July 26 – 30, 2015

Motivation & Objectives

Numerical simulation of atmospheric flows

Governing equations: 2D Euler equations with gravitational forces (conservation of mass, momentum and energy)

$$\frac{\partial}{\partial t}\begin{bmatrix}\rho\\ \rho u\\ \rho v\\ e\end{bmatrix}+\frac{\partial}{\partial x}\begin{bmatrix}\rho u\\ \rho u^2+p\\ \rho uv\\ (e+p)u\end{bmatrix}+\frac{\partial}{\partial y}\begin{bmatrix}\rho v\\ \rho uv\\ \rho v^2+p\\ (e+p)v\end{bmatrix}=\begin{bmatrix}0\\ \rho \mathbf{g}\cdot\hat{\mathbf{i}}\\ \rho \mathbf{g}\cdot\hat{\mathbf{j}}\\ \rho u\mathbf{g}\cdot\hat{\mathbf{i}}+\rho v\mathbf{g}\cdot\hat{\mathbf{j}}\end{bmatrix}$$

Time scales: entropy (u) << acoustic ($u \pm a$)

Time integration

- **Explicit time-integration** → time step size restricted by acoustic waves; but acoustic waves do not significantly impact any atmospheric phenomenon.
- **Implicit time-integration** → Unconditionally stable; but requires solutions to **non-linear system** or **linearized approximation**.
- **Implicit-Explicit (IMEX) time-integration** → Integrate **"fast" waves implicitly**, **"slow" waves explicitly**.
  - **Characteristic-based partitioning of the hyperbolic flux** (Acoustic waves integrated implicitly, entropy waves integrated explicitly)

Other (more popular) forms of the governing equations

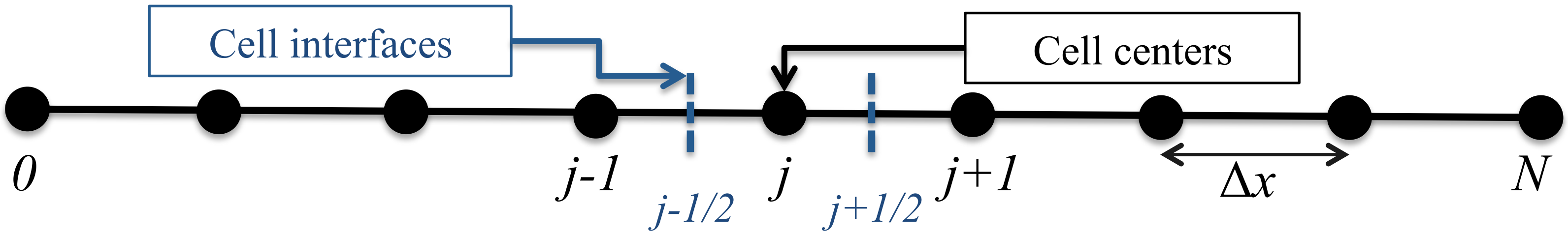
- **Exner pressure, velocity, potential temperature:** **COAMPS** (US Navy), **NMM** (NCEP), **MM5** (NCAR/PSU) .
- **Mass, momentum, potential temperature:** **WRF** (NCAR), **NUMA** (NPS).

Giraldo, Restelli, Laeuter, 2010: **Perturbation-based IMEX splitting** of the hyperbolic flux (first-order perturbations implicit, higher-order perturbations explicit)

Selective preconditioning of acoustic modes

- **Implicit Continuous Eulerian** (ICE) technique (***Harlow, Amsden, 1971***)
- Preconditioning applied to stiff modes (***Reynolds, Samtaney, Woodward, 2010***)

Spatial Discretization



Conservative finite-difference discretization of a hyperbolic conservation law

$$u_t + f(u)_x = 0;\ \ f'(u) \in \Re \quad \Rightarrow \quad \frac{du_j}{dt} + \frac{1}{\Delta x}\left[f(x_{j+1/2},t) - f(x_{j-1/2},t) \right] = 0$$

Weighted Essentially Non-Oscillatory (WENO) Schemes

- **Weights depend on the local smoothness of the solution**
- Optimal weights in smooth regions allow ***(2r-1)*****th order accuracy**
- Near-zero weights for stencils with discontinuities → **non-oscillatory behavior**
- **Compact-Reconstruction WENO (CRWENO)** → Higher spectral resolution and lower absolute errors for same order of convergence

$$f^{(\mathrm{WENO})}_{j+1/2} = \sum_{k=1}^{r} \omega_k f^{(r)}_{k,j+1/2}$$

$$\omega_k = \omega\left(IS_k\right)$$

Smoothness indicator

WENO5

$$\hat{f}^{(5)}_{j+1/2} = \frac{1}{30} f_{j-2} - \frac{13}{60} f_{j-1} + \frac{47}{60} f_j + \frac{27}{60} f_{j+1} - \frac{1}{20} f_{j+2}$$

CRWENO5 (Compact finite difference scheme)

$$\frac{3}{10} \hat{f}^{(5)}_{j-1/2} + \frac{6}{10} \hat{f}^{(5)}_{j+1/2} + \frac{1}{10} \hat{f}^{(5)}_{j+3/2} = \frac{1}{30} f_{j-1} + \frac{19}{30} f_j + \frac{1}{3} f_{j+1}$$

Characteristic-based Flux Splitting (1)

Separation of **acoustic** and **entropy** modes in the flux for implicit-explicit time integration

1D Euler equations

$$\frac{\partial \mathbf{u}}{\partial t} + \frac{\partial \mathbf{F}(\mathbf{u})}{\partial x} = 0$$

Spatial discretization

Semi-discrete ODE in time

$$\frac{\partial \mathbf{u}}{\partial t} = \hat{\mathbf{F}}(\mathbf{u}) = [\mathcal{D} \otimes \mathcal{A}(u)]\, \mathbf{u}$$

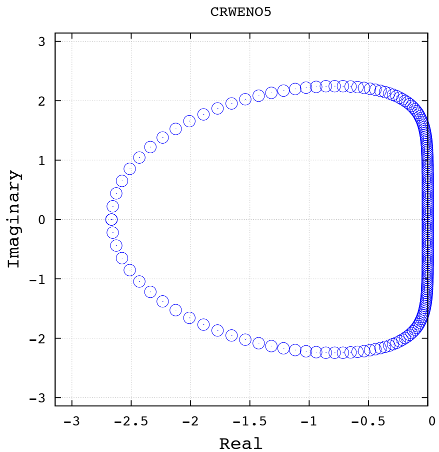
Discretization operator (e.g.:WENO5, CRWENO5)

Flux Jacobian

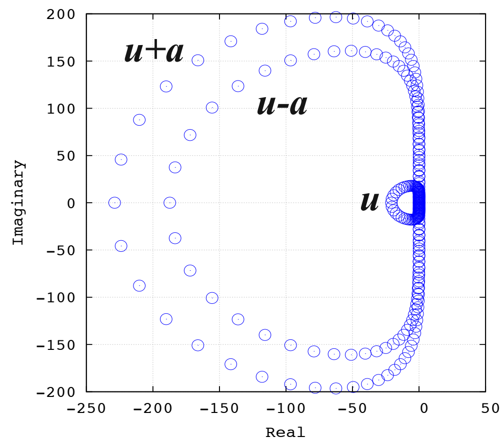
Example: Periodic density sine wave on a unit domain discretized by N=80 points.

$$\mathrm{eig}\left[\frac{\partial \hat{\mathbf{F}}}{\partial \mathbf{u}}\right] = \mathrm{eig}\,[\mathcal{D}] \times \mathrm{eig}\,[\mathcal{A}(\mathbf{u})]$$

Time step size limit for linear stability



Eigenvalues of the CRWENO5 discretization



Eigenvalues of the right-hand-side operator (u=0.1, a=1.0, dx=0.0125)

Eigenvalues of the right-hand-side of the ODE are the **eigenvalues of the discretization operator** times the **characteristic speeds** of the physical system

Characteristic-based Flux Splitting (2)

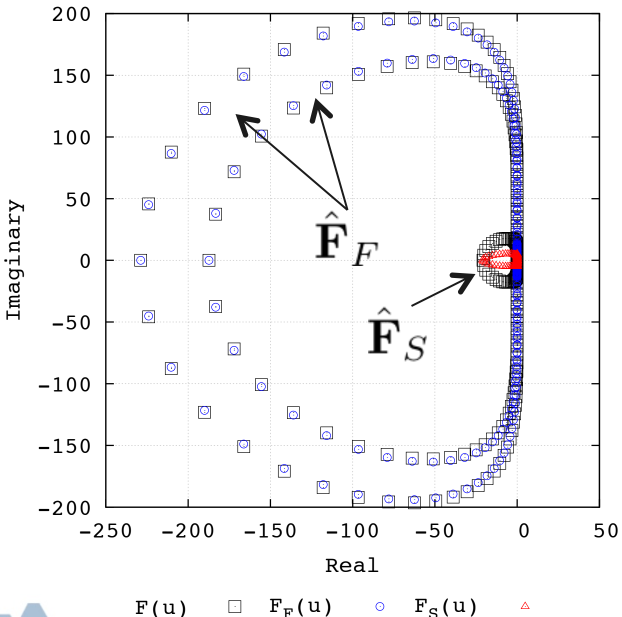
Splitting of the **flux Jacobian** based on its eigenvalues

$$\frac{\partial \mathbf{u}}{\partial t} = \hat{\mathbf{F}}(\mathbf{u}) = \left[\mathcal{D} \otimes \mathcal{A}(u)\right]\mathbf{u}$$
$$= \left[\mathcal{D} \otimes \mathcal{A}_S(u) + \mathcal{D} \otimes \mathcal{A}_F(u)\right]\mathbf{u}$$
$$= \hat{\mathbf{F}}_S(\mathbf{u}) + \hat{\mathbf{F}}_F(\mathbf{u})$$

"Slow" flux **"Fast" Flux**

$$\mathcal{A}(\mathbf{u}) = \mathcal{R}\Lambda\mathcal{L}$$
$$= \mathcal{R}\Lambda_S\mathcal{L} + \mathcal{R}\Lambda_F\mathcal{L}$$
$$= \mathcal{A}_{\mathcal{S}}(\mathbf{u}) + \mathcal{A}_{\mathcal{F}}(\mathbf{u})$$

$$\Lambda_S = \begin{bmatrix} u & & \\ & 0 & \\ & & 0 \end{bmatrix} \quad \Lambda_F = \begin{bmatrix} 0 & & \\ & u+a & \\ & & u-a \end{bmatrix}$$



Example: Periodic density sine wave on a unit domain discretized by N=80 points (CRWENO5).

$$\frac{\partial \mathbf{F}_{S,F}(\mathbf{u})}{\partial \mathbf{u}} \neq \left[\mathcal{A}_{S,F}\right]$$

Small difference between the eigenvalues of the complete operator and the split operator.

(Not an error)

$$\operatorname{eig}\left[\frac{\partial \hat{\mathbf{F}}_S}{\partial \mathbf{u}}\right] \approx u \times \operatorname{eig}\left[\mathcal{D}\right] \qquad \operatorname{eig}\left[\frac{\partial \hat{\mathbf{F}}_F}{\partial \mathbf{u}}\right] \approx \{u \pm a\} \times \operatorname{eig}\left[\mathcal{D}\right]$$

IMEX Time Integration with Characteristic-based Flux Splitting (1)

Apply **Implicit-Explicit Runge-Kutta** (PETSc - `TSARKIMEX`) time-integrators

$$\mathbf{U}^{(i)} = \mathbf{u}_n + \Delta t \sum_{j=1}^{i-1} a_{ij} \hat{\mathbf{F}}_S \left(\mathbf{U}^{(j)} \right) + \Delta t \sum_{j=1}^{i} \tilde{a}_{ij} \hat{\mathbf{F}}_F \left(\mathbf{U}^{(j)} \right)$$

Stage values

(s stages)

$$i = 1, \cdots, s$$

$$\mathbf{u}_{n+1} = \mathbf{u}_n + \Delta t \sum_{i=1}^{s} b_i \hat{\mathbf{F}}_S \left(\mathbf{U}^{(i)} \right) + \Delta t \sum_{i=1}^{s} \tilde{b}_i \hat{\mathbf{F}}_F \left(\mathbf{U}^{(i)} \right)$$

Step completion

⇒ **Non-linear system of equations**

$$\hat{\mathbf{F}}_F \left(\mathbf{u} \right) = \left[\mathcal{D} \left(\omega \right) \otimes \mathcal{A}_F \left(\mathbf{u} \right) \right] \mathbf{u}$$

$$\omega = \omega \left[\mathbf{F} \left(\mathbf{u} \right) \right]$$

Solution-dependent weights for the WENO5/CRWENO5 scheme

Linearized Formulation

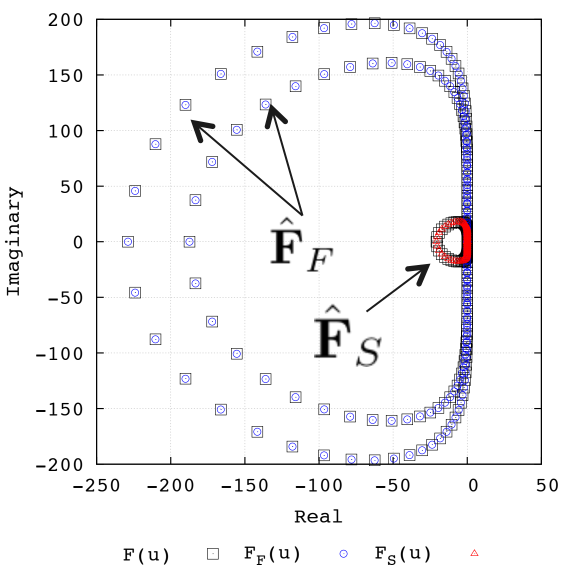
Redefine the splitting as

$$\mathbf{F}_F \left(\mathbf{u} \right) = \left[\mathcal{A}_F \left(\mathbf{u}_n \right) \right] \mathbf{u}$$
$$\mathbf{F}_S \left(\mathbf{u} \right) = \mathbf{F} \left(\mathbf{u} \right) - \mathbf{F}_F \left(\mathbf{u} \right)$$

Note: Introduces **no error** in the governing equation.

At the beginning of a time step:-

$$\text{eig} \left[\frac{\partial \hat{\mathbf{F}}_S}{\partial \mathbf{u}} \right] = u \times \text{eig} \left[\mathcal{D} \right], \quad \text{eig} \left[\frac{\partial \hat{\mathbf{F}}_F}{\partial \mathbf{u}} \right] = \left\{ u \pm a \right\} \times \text{eig} \left[\mathcal{D} \right]$$



Is F_F a good approximation at each stage?

IMEX Time Integration with Characteristic-based Flux Splitting (2)

Linearization of the WENO/CRWENO discretization: Within a stage, the non-linear coefficients are kept fixed.

Linear system of equations for implicit stages:

$$\left[\mathcal{I} - \Delta t \tilde{a}_{ii} \mathcal{D} \otimes \mathcal{A}_F\left(\mathbf{u_n}\right)\right] \mathbf{U}^{(i)} = \mathbf{u}_n + \Delta t \sum_{j=1}^{i-1} a_{ij} \hat{\mathbf{F}}_S\left(\mathbf{U}^{(j)}\right) + \Delta t \left[\mathcal{D} \otimes \mathcal{A}_F\left(\mathbf{u_n}\right)\right] \sum_{j=1}^{i-1} \tilde{a}_{ij} \mathbf{U}^{(j)},$$

$i = 1, \cdots, s$

Preconditioning (Preliminary attempts)

$$\mathcal{P} = \left[\mathcal{I} - \Delta t \tilde{a}_{ii} \mathcal{D}^{(1)} \otimes \mathcal{A}_F\left(\mathbf{u_n}\right)\right] \approx \left[\mathcal{I} - \Delta t \tilde{a}_{ii} \mathcal{D} \otimes \mathcal{A}_F\left(\mathbf{u_n}\right)\right]$$

First order upwind discretization

Periodic boundaries ignored

Block n-diagonal matrices

- Block tri-diagonal (1D)
- Block penta-diagonal (2D)
- Block septa-diagonal (3D)

- **Jacobian-free approach** → Linear Jacobian defined as a function describing its action on a vector (`MatShell`)
- **Preconditioning matrix** → Stored as a sparse matrix (`MatAIJ`)

ARK Methods (PETSc)

ARKIMEX 2c

- 2nd order accurate
- 3 stage (1 explicit, 2 implicit)
- L-Stable implicit part
- Large real stability of explicit part

ARKIMEX 2e

- 2nd order accurate
- 3 stage (1 explicit, 2 implicit)
- L-Stable implicit part

ARKIMEX 3

- 3rd order accurate
- 4 stage (1 explicit, 3 implicit)
- L-Stable implicit part

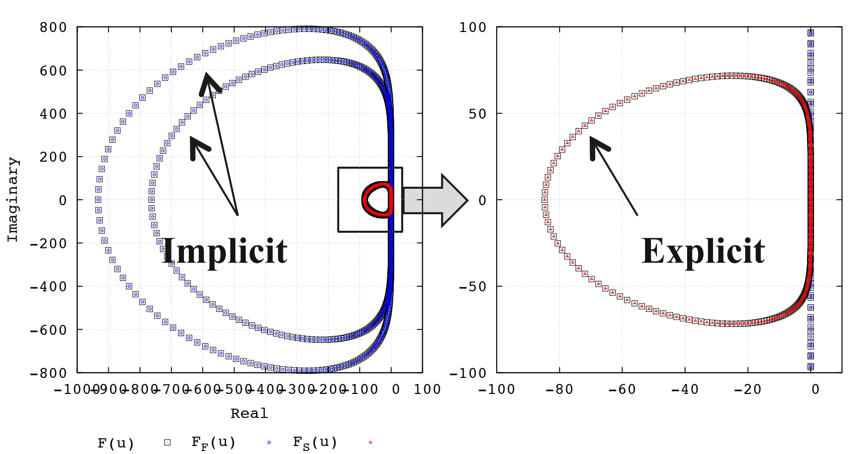
ARKIMEX 4

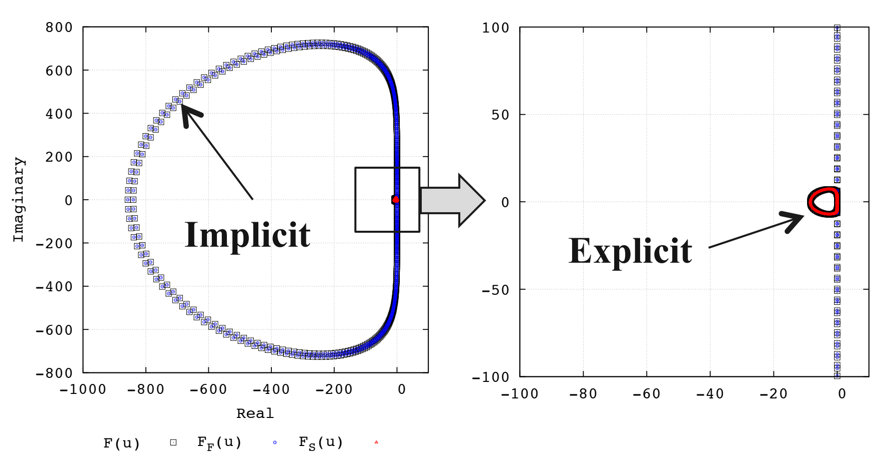
- 4th order accurate
- 5 stage (1 explicit, 4 implicit)
- L-Stable implicit part

Example: 1D Density Wave Advection

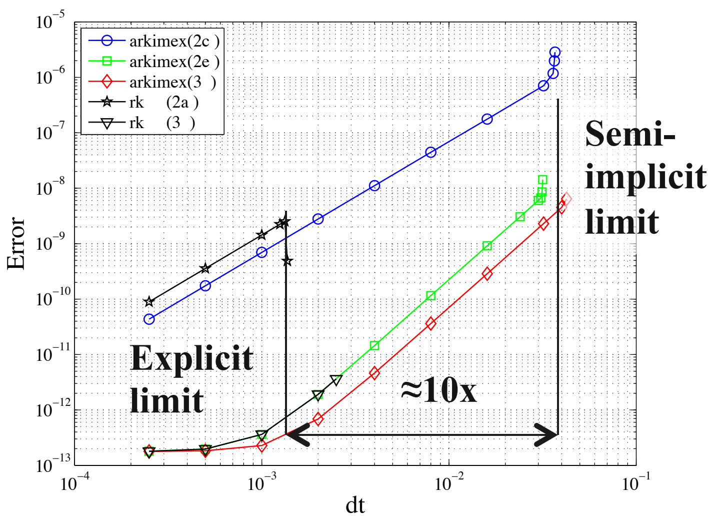
Initial solution $\rho = \rho_\infty + \hat{\rho}\sin\left(2\pi x\right), u = u_\infty, p = p_\infty;\; 0 \le x \le 1$

$M_\infty = 0.1$ **Eigenvalues** $M_\infty = 0.01$

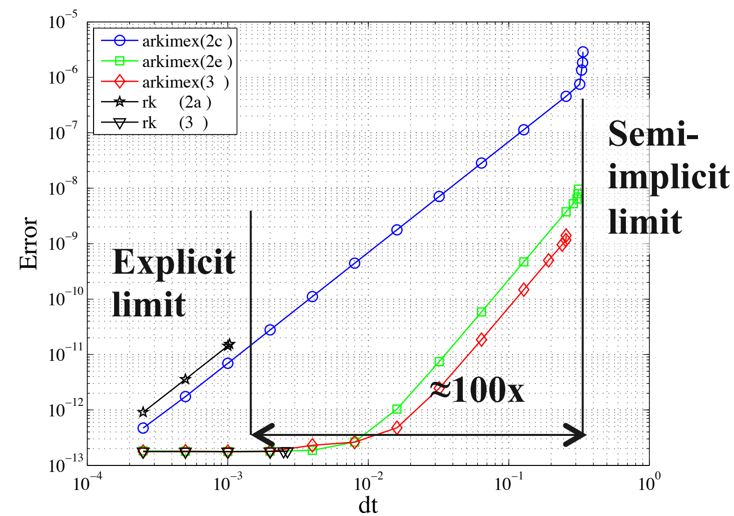




CRWENO5, 320 grid points



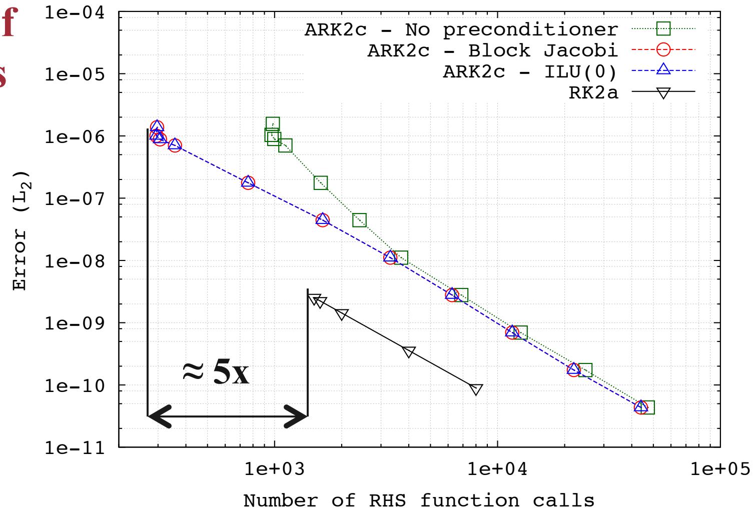
Semi-implicit **time step size limit determined** by the **flow velocity**

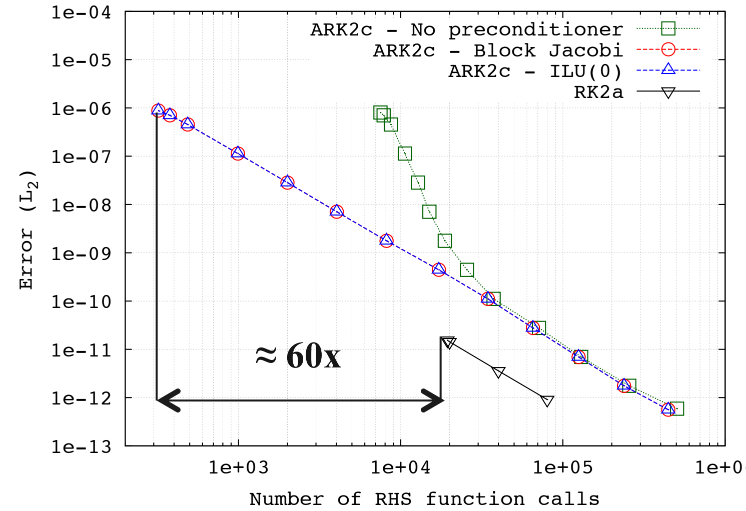


Example: 1D Density Wave Advection (Computational Cost)

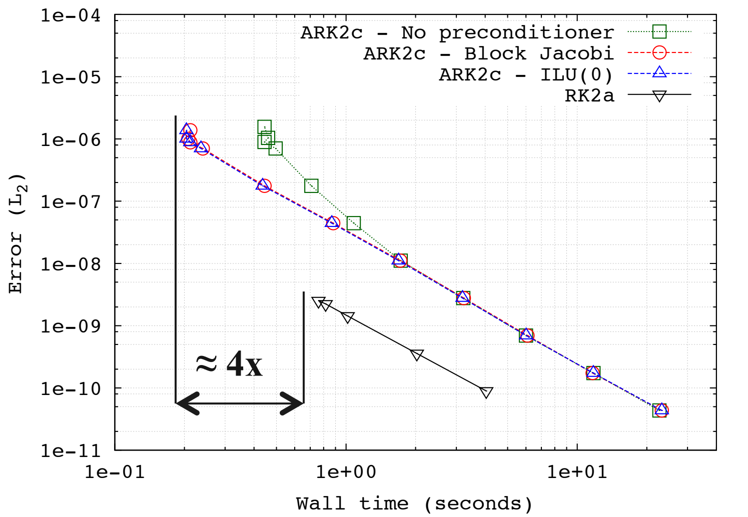
$M_\infty = 0.1$ | $M_\infty = 0.01$

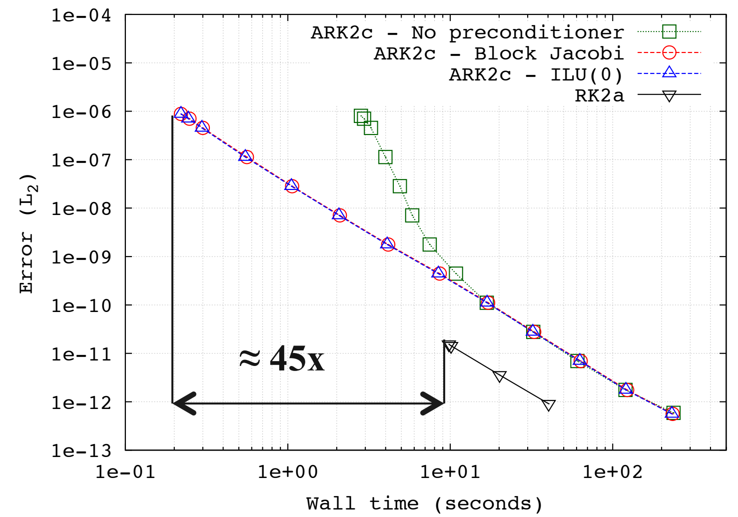
Number of function calls





Wall time





Example: Low Mach Isentropic Vortex Convection

Freestream flow

$$\left.\begin{aligned}\rho_\infty &= 1\\ p_\infty &= 1\\ u_\infty &= 0.1\\ v_\infty &= 0\end{aligned}\right\} M_\infty \approx 0.08$$

Vortex (Strength $b = 0.5$)

$$\rho = \left[1.0 - \frac{(\gamma-1)\,b^2}{8\gamma\pi^2}\exp\left(1-r^2\right)\right]^{\frac{1}{\gamma-1}}$$

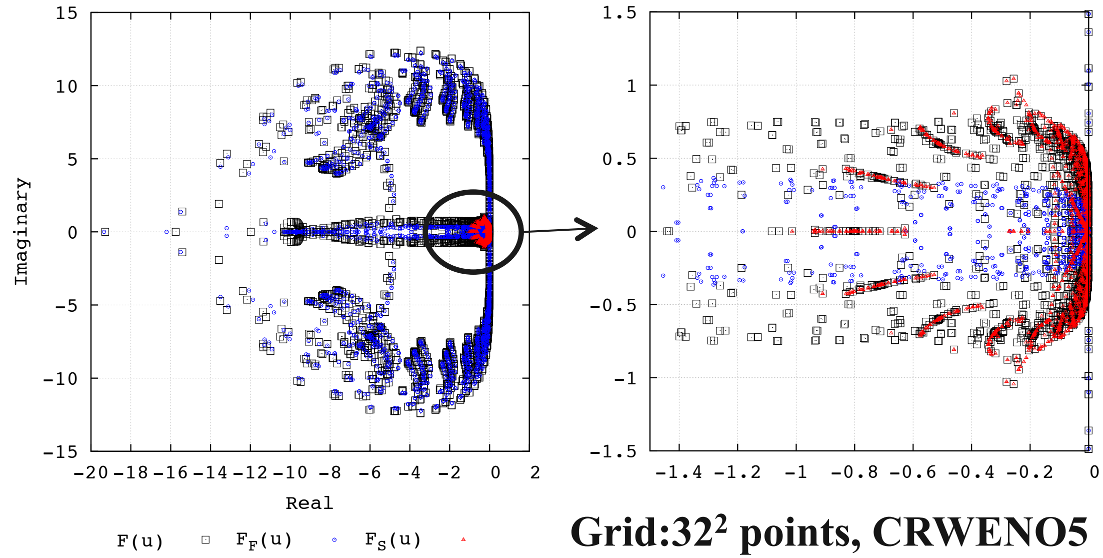
$$p = \left[1.0 - \frac{(\gamma-1)\,b^2}{8\gamma\pi^2}\exp\left(1-r^2\right)\right]^{\frac{\gamma}{\gamma-1}}$$

$$u = u_\infty - \frac{b}{2\pi}\exp\left(\frac{1-r^2}{2}\right)(y-y_c)$$

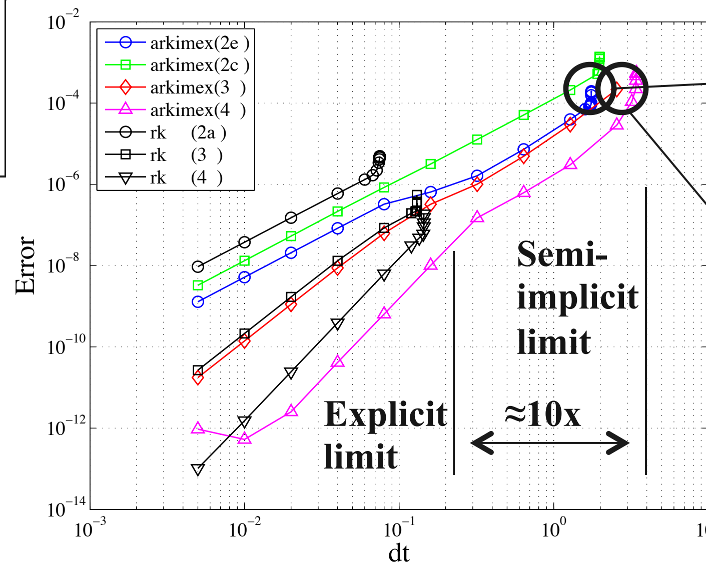
$$v = v_\infty + \frac{b}{2\pi}\exp\left(\frac{1-r^2}{2}\right)(x-x_c)$$

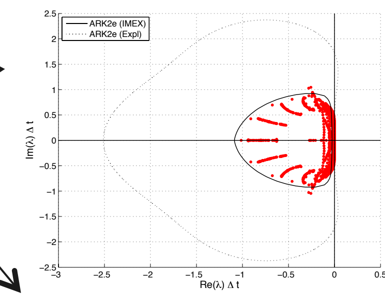
- **Optimal orders** of convergence observed for all methods
- **Time step size limited by the "slow" eigenvalues.**

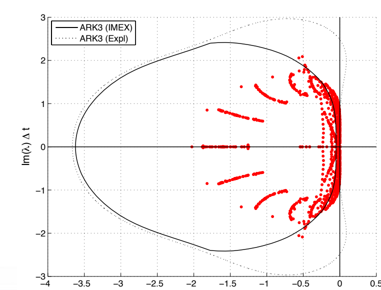
Eigenvalues of the right-hand-side operators



Grid:32^2 points, CRWENO5





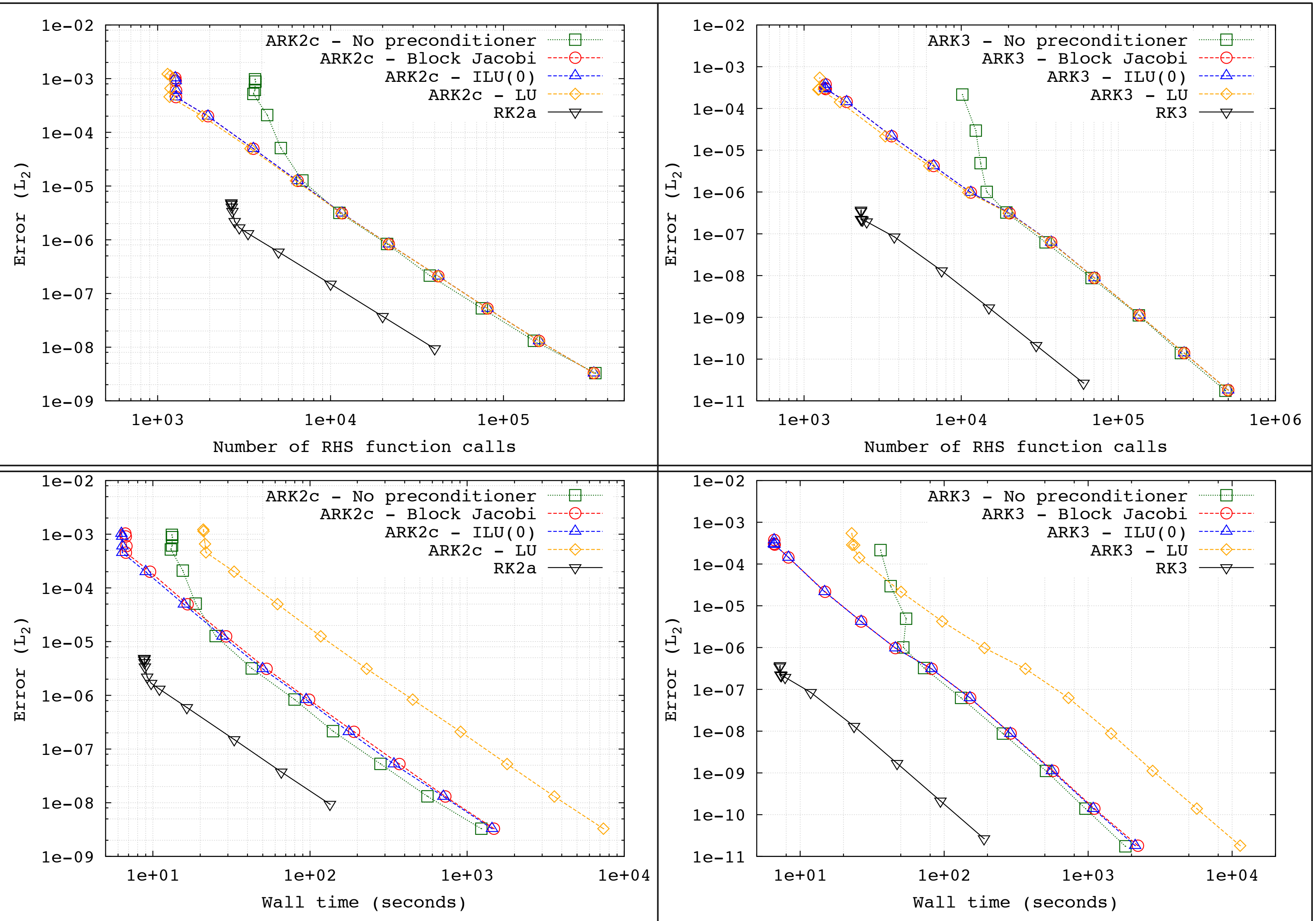


Example: Vortex Convection (Computational Cost)

ARK 2c | ARK 3

Number of function calls

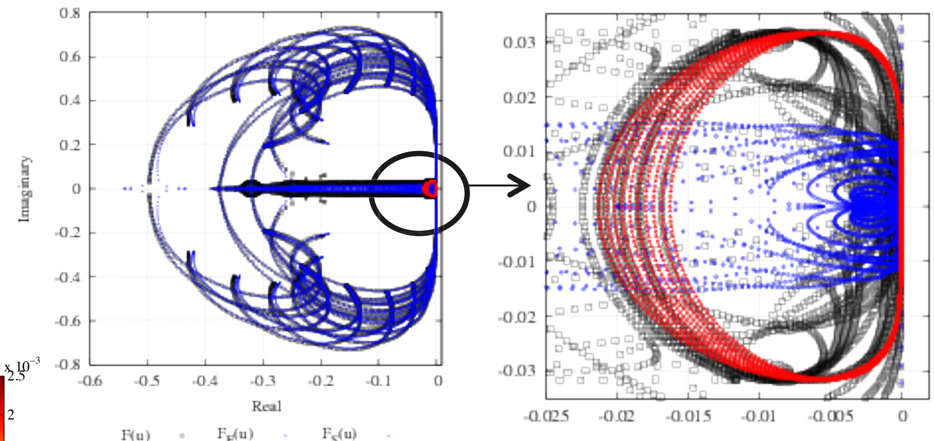
Wall time



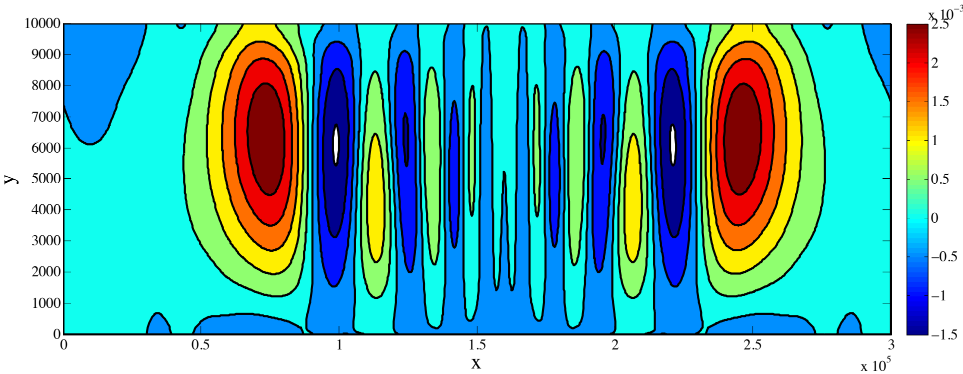
Example: Inertia – Gravity Wave

- **Periodic channel** – 300 km x 10 km
- **No-flux boundary conditions** at top and bottom boundaries
- **Mean horizontal velocity** of 20 m/s in a uniformly stratified atmosphere **($M_\infty \approx$ 0.06)**
- Initial solution – **Potential temperature perturbation**

Eigenvalues of the right-hand-side operators



Grid: 300x10 points, CRWENO5



Potential temperature perturbations at 3000 seconds (Solution obtained with **WENO5** and **ARKIMEX 2e**, 1200x50 grid points)

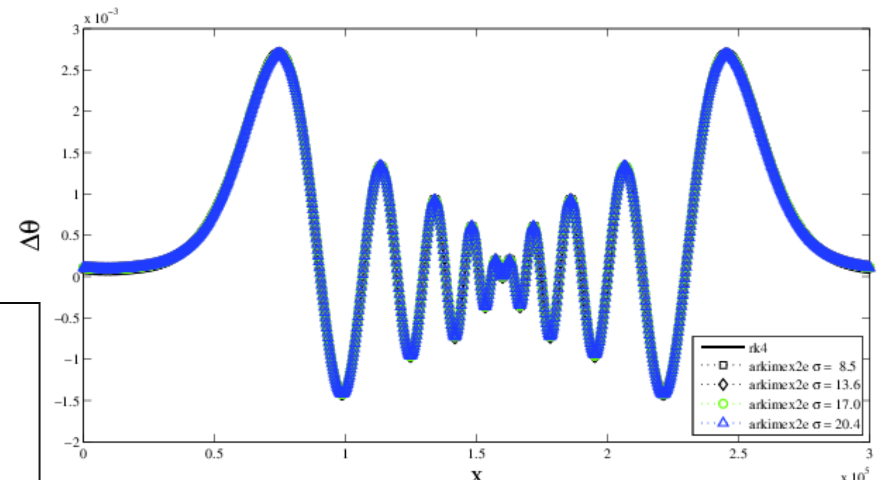
CFL	Wall time (s)	Function counts
8.5	6149	24800
13.6	**4118**	**17457**
17.0	**3492**	**14820**
20.4	**2934**	**12895**

RK4

CFL ~ **1.0**

Wall time: **5400 s**

Function counts: **24000**



Cross-sectional potential temperature perturbations at 3000 seconds (y = 5 km) at various CFL numbers **(0.2 – 13.6)**

Conclusions

Characteristic-based flux splitting (Work in progress):

- Partitioning of flux **separates the acoustic and entropy modes** → Allows **larger time step sizes** (determined by flow velocity, not speed of sound).
- **Comparison** to alternatives
  - **Vs. explicit time integration:** Larger time steps → More efficient algorithm
  - **Vs. implicit time integration:** Semi-implicit solves a linear system **without any approximations** to the overall governing equations (as opposed to: solve non-linear system of equations or linearize governing equations in a time step).

To do:

- **Improve efficiency** of the **linear solve**
  - Better preconditioning of the linear system
- Extend to **3D flow problems**

Thank you!

Acknowledgements

U.S. Department of Energy, Office of Science, Advanced Scientific Computing Research