O(N) unconditionally stable methods through kernel based Successive Convolution

Andrew Christlieb

Michigan State University
Department of Computational Mathematics, Science and Engineering

ICERM workshop
Holistic Design of Time-Dependent PDE Discretizations

Jan 14, 2022

Team

William Sands (Student)

Hyoseon Yang (Former Post Doc at Kyung Hee University)

Yan Jiang (Former Post Doc now at USTC)

Wei Gou (Former Post Doc now at Texas Tech)

Thank you to: AFOSR, ONR, NSF and DoE

Rothe's Method

- Erich Rothe: 1930, "Zweidimensionale parabolische randwertaufgaben als grenzfall eindimensionaler randwertaufgaben", *Mathematische Annalen*,
- Analysis and Numerics Summary Papers
  - Jozef Kacur: 1978, "Method of Rothe and nonlinear parabolic boundary value problems of arbitrary order", *Czechoslovak Mathematical Journal*,
  - Karel Rektorys: 1982, "The method of discretization in time and partial differential equations", *Equadiff 5*
  - Fritz John: 1985, "On integration of parabolic equations by difference methods", *Fritz John Collected Papers*,
- Transverse Method of Lines (Show that treating all time at once lades to constant high order methods in time)
  - Pierluigi Amodio and Francesca Mazzia: 1995, "A boundary value approach to the numerical solution of initial value problems by multistep methos", *Journal of Difference Equations and Applications*,
  - Annamaria Mazzia and Francesca Mazzia: 1997, "High-order transverse schemes for the numerical solution of PDEs", *Journal of computational and applied mathematics*

Rothe's Method

- Parallel in Time (All of time, but distributed computing)
  - Jacques-Louis Lions, Yvon Maday, and Gabriel Turinici: 2001, "Résolution d'EDP par un schéma en temps «pararéel»", *Comptes Rendus de l'Académie des Sciences-Series I-Mathematics*,
  - Robert D Falgout, Stephanie Friedhoff, Tz V Kolev, Scott P MacLachlan, and Jacob B Schroder: 2014, "Parallel time integration with multigrid", *SIAM Journal on Scientific Computing*

- Method of Lines Transpose (Use of Greens functions to address BVP)
  - Jun Jia and Jingfang Huang: 2008, "Krylov deferred correction accelerated method of lines transpose for parabolic problems", *Journal of Computational Physics*,
  - Matthew Causley, Andrew J Christlieb, Benjamin Ong, and Lee Van Groningen: 2014 "Method of lines transpose: An implicit solution to the wave equation", *Mathematics of Computation*
  - Matthew Causley, Hana Cho, Andrew J Christlieb, David C Seal: 2016, "Method of Lines Transpose: High Order L-Stable $\mathcal{O}(N)$ Schemes for Parabolic Equations Using Successive Convolution", *SIAM Journal on Numerical Analysis*

Rothe's Method

- Successive Convolution (Expanding the spatial operator in continues well behaved convergent expansions)
  - Andrew J Christlieb, Wei Guo, Yan Jiang and Hyoseon Yang: 2020, "Kernel based high order 'explicit' unconditionally stable scheme for nonlinear degenerate advection-diffusion equations", *Journal of Scientific Computing*,
  - Andrew J Christlieb, William A Sands and Hyoseon Yang: 2020, "A Kernel-Based explicit unconditionally stable scheme for Hamilton-Jacobi equations on nonuniform meshe", *Journal of Scientific Computing*,
  - Andrew J Christlieb, Pierson T Guthrey, and William A Sands, and Mathialakan Thavappiragasm: 2021, "Parallel Algorithms for Successive Convolution", *Journal of Scientific Computing*.

What are we talking about?

Two big themes:

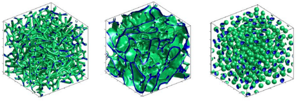
- First - New spatial discretization that makes explicit time stepping methods be unconditionally stable.
  Method designed for multi-core computing,
  Avoids iteration and all to one communication of implicit solves.

- Second - The method has been developed to solve PDEs with several boundary conditions.

We will go back and forth between the two as we put the story together.

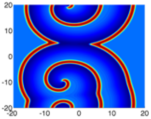
Photo gallery of applications of Kernel based methods

High order Phase Filed Models

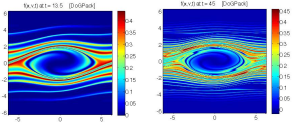


Pore network   Lamella network   Inverted micelles

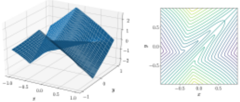
2D FitzHugh-Nagumo



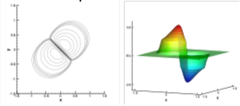
Vlasov Models



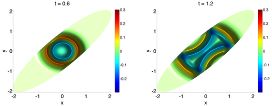
Hamilton-Jacobi



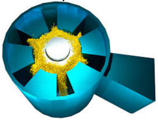
Nonlinear degenerate advection-diffusion equation



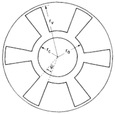
Acoustics in Cavities



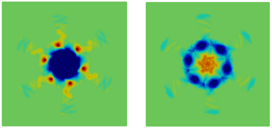
Maxwell's Equations and Particle Simulations



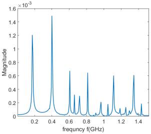
TechX: A6



2D A6



4th order time 4th order space



A6 2D Ping Test

Matthew F Causley and Andrew J Christlieb: 2014, "Higher order A-stable schemes for the wave equation using a successive convolution approach", *SIAM Journal on Numerical Analysis*.

Non-ideal MHD ($\nabla \cdot \mathbf{B} = 0$, $B = \nabla \times \mathbf{A}$, and $J = \nabla \times \mathbf{B}$)

$$\partial_t \mathbf{A} + (\nabla \times \mathbf{A}) \times \mathbf{u} = -\left(\eta \mathbf{J} + \frac{1}{ne} \mathbf{J} \times \nabla \times \mathbf{A} + \frac{m_e}{ne^2} \partial_t \mathbf{J}, \right)$$

$$\partial_t \mathbf{J} = \partial_t \nabla \times \nabla \times \mathbf{A}$$

Hamilton-Jacobi equations - Constrained Transport for MHD

Example

Consider 1D Hamilton-Jacobi equations

$$\phi_t + H(\phi_x) = 0.$$

We construct the following semi-discrete scheme

$$\frac{d}{dt}\phi_i(t) + \hat{H}(\phi_{x,i}^-, \phi_{x,i}^+) = 0,$$

where $\phi_{x,i}^-$ and $\phi_{x,i}^+$ are the approximations to ϕ_x at x_i obtained by left-biased and right-biased methods, respectively.

Applications in diverse fields:

- Optimal control, seismic waves, crystal growth, robotic navigation, image processing, and calculus of variations.
- Burgers equation, magnetic scalar/vector potential equations for MHD, Navier-Stokes with Maxwell equation can also be casted as H-J equations.

Reference:

- A.J. Christlieb, W. Guo, and Y. Jiang, *A kernel based high order "explicit" unconditionally stable scheme for time dependent Hamilton-Jacobi equations*, J. Comp. Phys. (2020).

What to look for in this talk:

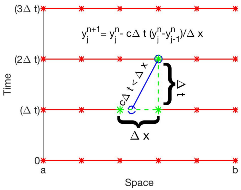
Consider $\partial_t y = c\partial_x f(y)$.

- Explicit time stepping, for example, forward Euler

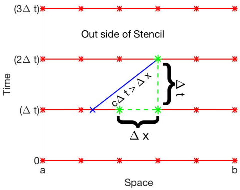
$$\frac{y^{n+1} - y^n}{\Delta t} = c\partial_x f(y)$$

$$\frac{y_j^{n+1} - y_j^n}{\Delta t} = c\frac{f(y_j^n) - f(y_{j-1}^n)}{\Delta x}$$

- CFL $\Delta t < \frac{\Delta x}{c}$, finite propagation $\rightarrow$ stable.



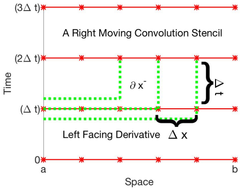
(a) Stable



(b) Unstable

Kernel based approximation

- Terms like ∂_x and $\partial_x(n(x)\partial_x)$ are made up of linear derivatives " ∂_x".
- Big idea: $\partial_x \sim$ global instead of local gives ($\Delta t > \frac{\Delta x}{c} \rightarrow$ stable).
- Approximate ∂_x via a fast $O(N)$ convolution integral with a kernel that gives an $O(\Delta t^k)$ approximation.



(c) Stable for all Δt

- For linear PDEs, provably unconditionally stable even when using explicit time stepping.

Kernel Based Expansion of ∂_x,

$\sim \partial_x(\cdot)$ with $\Sigma_{i=0}^{p}\left(\int_a^x(\cdot)\right)^i dy$ in $\mathcal{O}(N)$.

Motivation $\rightarrow$ MOL^T

Prototypical linear PDEs:

- Linear advection equation: $(\partial_t - c\partial_x)u = 0$
- Diffusion equation: $(\partial_t - \nu\partial_{xx})u = 0$
- Wave equation: $(\partial_{tt} - c^2\partial_{xx})u = 0$

Semi-discretize the equations in time (could use a BDF method to get high order):

- Linear advection equation: $(\mathcal{I} - c\Delta t\partial_x)u^{n+1} = u^n$
- Diffusion equation: $(\mathcal{I} - \nu\Delta t\partial_{xx})u^{n+1} = u^n$
- Wave equation: $(\mathcal{I} - c^2\Delta t^2\partial_{xx})u^{n+1} = 2u^n - u^{n-1}$

with an identity operator $\mathcal{I}$.

Observation: the operators $\mathcal{I} \pm \frac{1}{\alpha}\partial_x$ arise in each of examples.

$$(\mathcal{I} - \frac{1}{\alpha^2}\partial_{xx}) = (\mathcal{I} - \frac{1}{\alpha}\partial_x)(\mathcal{I} + \frac{1}{\alpha}\partial_x)$$

We introduce the operators for simplicity:

$$\mathcal{L}_L := \mathcal{I} - \frac{1}{\alpha}\partial_x \quad \text{and} \quad \mathcal{L}_R := \mathcal{I} + \frac{1}{\alpha}\partial_x.$$

Integral solutions to MOL^T formulation

Consider the univariate Helmholtz operators

$$\mathcal{L}_L[u] = \left(\mathcal{I} - \frac{1}{\alpha}\partial_x\right)u(x) \quad \text{and} \quad \mathcal{L}_R[u] = \left(\mathcal{I} + \frac{1}{\alpha}\partial_x\right)u(x).$$

We define convolution with the Green's function by the integral operators

$$I_L[u](x) := \alpha \int_x^b e^{-\alpha(\tau - x)} u(\tau)\, d\tau \quad \text{and} \quad I_R[u](x) := \alpha \int_a^x e^{-\alpha(x-\tau)} u(\tau)\, d\tau$$

so that the Helmholtz operators are inverted as

$$\mathcal{L}_L^{-1} = Be^{-\alpha(b-x)} + I_L[u](x) \quad \text{and} \quad \mathcal{L}_R^{-1} = Ae^{-\alpha(x-a)} + I_R[u](x),$$

with boundary terms B and A in the Homogeneous solutions.

Kernel expansion of ∂_x^+

$$\mathcal{L}_L = (\mathcal{I} - \frac{1}{\alpha}\partial_x)(\cdot) + B.C. \ , \quad \mathcal{L}_L^{-1} = B^{n+1} e^{-\alpha(b-x)} + \alpha \int_x^b e^{-\alpha(\tau - x)}(\cdot) d\tau.$$

Rewriting the first equation:

$$-\frac{1}{\alpha}\partial_x = (\mathcal{L}_L - \mathcal{I}) + B.C.$$

Defining a new operator $\mathcal{D}_L = \mathcal{I} - \mathcal{L}_L^{-1}$, then this gives

$$\mathcal{L}_L = (\mathcal{I} - \mathcal{D}_L)^{-1}$$

and we expand $(\mathcal{I} - \mathcal{D}_L)^{-1}$ in a power series

$$-\frac{1}{\alpha}\partial_x^+ = (\mathcal{L}_L - \mathcal{I}) = \mathcal{L}_L(\mathcal{I} - \mathcal{L}_L^{-1}) = (\mathcal{I} - \mathcal{D}_L)^{-1}\mathcal{D}_L = \sum_{p=1}^{\infty} \mathcal{D}_L^p.$$

Kernel based approximation

A right traveling wave gives left facing operator (x moving from b to a) to give a right facing derivative:

$$\mathcal{L}_L = (1 - \frac{1}{\alpha}\partial_x)(\cdot) + B.C. \ , \quad \mathcal{L}_L^{-1} = B^{n+1} e^{-\alpha(b-x)} + \alpha \int_x^b e^{-\alpha(\tau - x)}(\cdot) d\tau$$

$$\partial_x^+ = -\alpha \sum_{p=1}^{\infty} \mathcal{D}_L^p$$

and a left traveling wave gives right facing operator (x moving from a to b) to give a left facing derivative:

$$\mathcal{L}_R = (1 + \frac{1}{\alpha}\partial_x)(\cdot) + B.C. \ , \quad \mathcal{L}_R^{-1} = A^{n+1} e^{-\alpha(x-a)} + \alpha \int_a^x e^{-\alpha(x - \tau)}(\cdot) d\tau$$

$$\partial_x^- = \alpha \sum_{p=1}^{\infty} \mathcal{D}_R^p$$

where

$$\mathcal{D}_L = \mathcal{I} - \mathcal{L}_L^{-1} \quad \text{and} \quad \mathcal{D}_R = \mathcal{I} - \mathcal{L}_R^{-1}.$$

Kernel Based Methods: Description and Analysis

Consistency

Multiple ways to establish consistency, easiest way is to observe the relations between our series and the resolvent expansion [Abadias et. al. (2017).].

Theorem (Consistency)

If $\phi \in C^{k+1}[a, b]$, then

$$\|\partial_x \phi(x) + \alpha \sum_{p=1}^{k} \mathcal{D}_L^p[\phi](x)\|_\infty = \mathcal{O}(\Delta t^k), \quad \|\partial_x \phi(x) - \alpha \sum_{p=1}^{k} \mathcal{D}_R^p[\phi](x)\|_\infty = \mathcal{O}(\Delta t^k).$$

Reference:

- A.J. Christlieb, W. Guo, and Y. Jiang, *A kernel based high order "explicit" unconditionally stable scheme for time dependent Hamilton-Jacobi equations*, J. Comp. Phys. (2019).
- L. Abadias and P.J. Miana, *C0-semigroups and resolvent operators approximated by Laguerre expansions*, Journal of Approximation Theory (2017).
- M. Causley, H. Cho, A.J. Christlieb and D. Seal, *Method of Lines Transpose: High Order L-Stable $\mathcal{O}(N)$ Schemes for Parabolic Equations Using Successive Convolution*, SIAM J. Numer. Anal. (2016)

Kernel based scheme 1 - The whole picture

$(\star)\ \partial_t y - c\partial_x y = 0 \quad + B.C.$

Single-step and Multi-stage method

Apply SSP-RK3 for the equation $(\star)$:

$$
\begin{aligned}
y^{(1)} &= y^n - \Delta t c \partial_x y^n, \\
y^{(2)} &= \frac{3}{4} y^n + \frac{1}{4}\left(y^{(1)} - \Delta t c \partial_x y^{(1)}\right), \\
y^{n+1} &= \frac{1}{3} y^n + \frac{2}{3}\left(y^{(2)} - \Delta t c \partial_x y^{(2)}\right).
\end{aligned}
$$

Then we use the approximation

$$
\partial_x^- = \alpha \sum_{p=1}^{3} \mathcal{D}_R^p + O(\Delta t^3).
$$

Reference:

- A.J. Christlieb, W. Guo, and Y. Jiang, *A kernel based high order "explicit" unconditionally stable scheme for time dependent Hamilton-Jacobi equations*, J. Comp. Phys. (2019).
- A.J. Christlieb, W. Guo, Y. Jiang and H. Yang, *A kernel based high order "explicit" unconditionally stable scheme for nonlinear degenerate advection-diffusion equation*, J. Sci. Comp. (2020).
- A.J. Christlieb, W. Sands and H. Yang, *A kernel based explicit unconditionally stable scheme for Hamilton-Jacobi equations on nonuniform meshes*, J. Comp. Phys., (2020).

Kernel based scheme 2 - The whole picture

$(\star)$ $\partial_t y - c\partial_x y = 0 \quad + B.C.$

Multi-step and Single-stage method

Integrating the equation $(\star)$:

$$\int_t^{t+\Delta t} \partial_\tau y(\tau, x) d\tau - c\partial_x \int_t^{t+\Delta t} y(\tau, x) d\tau = 0$$

gives the update

$$y^{n+1} = y^n - c\partial_x P[y^n](x)$$

with

$$P[y^n](x) := \int_{t^n}^{t^{n+1}} y(\tau, x) d\tau \approx \Delta t \Big(\frac{5}{12} y^{n-2}(x) - \frac{4}{3} y^{n-1}(x) + \frac{23}{12} y^n(x) \Big).$$

Then we use the approximation

$$\partial_x^- = \alpha \sum_{p=1}^{3} \mathcal{D}_R^p + O(\Delta t^3).$$

Reference:

- A.J. Christlieb and H. Yang, *High order single-stage multi-step schemes with MOL^T approach*, In progress.

Stability

Consider Hamilton-Jacobi equations on $[a, b]$ with periodic boundary conditions. We take the parameter $\alpha = \beta/(c\Delta t)$ with $c = \max|H'(\phi)|$. Here we address the operators $\mathcal{D}_L$ and $\mathcal{D}_R$ with periodic boundary treatments

$$\mathcal{D}_L^p[\phi](a) = \mathcal{D}_L^p[\phi](b) \quad \text{and} \quad \mathcal{D}_R^p[\phi](a) = \mathcal{D}_R^p[\phi](b);$$

Theorem (Stability: Von Neumann analysis)

Suppose the suggested method employs the k-th order SSP RK scheme or k-th order multistep strategy derived above for $k = 1, 2, 3$. There exists constant $\beta_{k,max} > 0$ such that the scheme is A-stable provided $0 < \beta \leq \beta_{k,\max}$.

k	1	2	3
$\beta_{k,max}$	2	1	1.243

Reference:

- A.J. Christlieb, W. Guo, and Y. Jiang, *A kernel based high order "explicit" unconditionally stable scheme for time dependent Hamilton-Jacobi equations*, J. Comp. Phys. (2019).
- A.J. Christlieb and H. Yang, *High order single-stage multi-step schemes with MOL^T approach*, In progress, 2020.
- A.J. Christlieb, W. Guo, Y. Jiang and H. Yang, *A kernel based high order "explicit" unconditionally stable scheme for nonlinear degenerate advection-diffusion equation*, J. Sci. Comp. (2020).

Kernel Based Methods
with Successive Convolution and WENO

Kernel based schemes for nonlinear equations

Recap: For distributed spatial domain $\{x_i\}$, we approximate

$$\phi_x^+(x_i) \approx -\alpha \sum_{p=1}^{k} \mathcal{D}_L^p[\phi](x_i),$$

$$\phi_x^-(x_i) \approx \alpha \sum_{p=1}^{k} \mathcal{D}_R^p[\phi](x_i),$$

where $\mathcal{D}_L^p = \mathcal{D}_L[\mathcal{D}_L^{p-1}]$ and $\mathcal{D}_R^p = \mathcal{D}_R[\mathcal{D}_R^{p-1}]$ for $p > 1$ and

$$\mathcal{D}_L = \mathcal{I} - \mathcal{L}_L^{-1}, \quad \mathcal{D}_R = \mathcal{I} - \mathcal{L}_R^{-1},$$

with

$$\mathcal{L}_L^{-1}[v](x_i) = \alpha \int_{x_i}^{b} e^{-\alpha(y-x_i)} v(y)\,dy + Be^{-\alpha(b-x)},$$

$$\mathcal{L}_R^{-1}[v](x_i) = \alpha \int_{a}^{x_i} e^{-\alpha(x_i-y)} v(y)\,dy + Ae^{-\alpha(x-a)}.$$

Successive convolution for I_L and I_R

Then

$$I_L[v](x_i) := \alpha \int_{x_i}^{b} e^{-\alpha(y-x_i)} v(y)\,dy \quad \text{and} \quad I_R[v](x_i) := \alpha \int_{a}^{x_i} e^{-\alpha(x_i-y)} v(y)\,dy$$

can be calculated by

$$\begin{aligned} I_L[v](x_i) &= e^{-\alpha \Delta x_{i+1}} I_L[v](x_{i+1}) + J_L[v](x_i), \quad i = 0, \dots, N-1, \\ I_R[v](x_i) &= e^{-\alpha \Delta x_i} I_R[v](x_{i-1}) + J_R[v](x_i), \quad i = 1, \dots, N, \end{aligned}$$

where $I^L[v](x_N) = 0$ and $I^R[v](x_0) = 0$, and

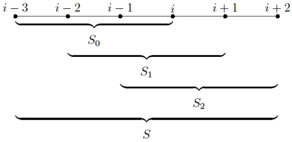
$$J_L[v](x_i) = \alpha \int_{x_i}^{x_{i+1}} v(y) e^{-\alpha(y-x_i)}\,dy \quad \text{and} \quad J_R[v](x_i) = \alpha \int_{x_{i-1}}^{x_i} v(y) e^{-\alpha(x_i-y)}\,dy.$$

Therefore, once $J_L[v](x_i)$ and $J_R[v](x_i)$ are computed for all i, we then can obtain $I_L[v](x_i)$ and $I_R[v](x_i)$ via the recursive relation above.

Evaluate $J_L[v](x_i)$ and $J_R[v](x_i)$ with WENO methodology.

WENO based quadrature

Consider the approximation for $J_R[v](x_i)$ on the 6-point stencil $\{x_{i-3}, \cdots, x_{i+2}\}$:



On the stencil $S(i)$ and $S_r(i), r = 0, 1, 2$, we obtain the approximation as

$$J_i := \alpha \int_{x_{i-1}}^{x_i} e^{-\alpha(x_i - s)} p(s)\, ds, \quad \text{and} \quad J_{i,r} := \alpha \int_{x_{i-1}}^{x_i} e^{-\alpha(x_i - s)} p_r(s)\, ds,$$

where p and p_r are interpolants to v on $S(i)$ and $S_r(i)$ that satisfy

$$J_i = J_R[v](x_i) + \alpha \mathcal{O}(\Delta x^6) \quad \text{and} \quad J_{i,r} = J_R[v](x_i) + \alpha \mathcal{O}(\Delta x^4)$$

if v is smooth. Then we can find linear weights d_r s.t.

$$J_i = \sum_{r=0}^{2} d_r J_{i,r}.$$

WENO based quadrature: nonlinear approximation

From

$$\sum_{r=0}^{2} d_r J_{i,r} \approx J_R[v](x_i),$$

we construct the final approximation to $J_R[v](x_i)$ as

$$\sum_{r=0}^{2} \omega_r J_{i,r}$$

where the nonlinear weight ω_r for each of local solutions $J_{i,r}$ is defined by

$$\omega_r = \frac{\alpha_r}{\sum_k \alpha_k} \quad , \quad \alpha_r = \frac{d_r}{\epsilon + \beta_r}$$

so that $\omega_r \approx d_r$ and

$$\sum_{r=0}^{2} \omega_r J_{i,r} = J_R[v](x_i) + \alpha \mathcal{O}(\Delta x^6).$$

Here β_r is derived to measure the smoothness of the function on each of substencils $S_r(i)$, $r = 0, 1, 2$ and $\epsilon > 0$.

Numerical Results (Part 1)

Linear equations

Example

$$\phi_t + (\phi_x + \phi_y + 1) = 0, \quad -2 \le x, y \le 2,$$
$$\phi(x, y, 0) = -\cos(\pi(x + y)/2).$$

CFL	$N_x \times N_y$	$k = 1.\ \beta = 1.$		$k = 2.\ \beta = 0.5.$		$k = 3.\ \beta = 0.6.$	
		error	order	error	order	error	order
0.5	20×20	1.28e-02	–	1.73e-01	–	2.25e-03	–
	40×40	3.22e-03	1.987	4.48e-02	1.947	1.69e-04	3.736
	80×80	8.07e-04	1.998	1.13e-02	1.990	1.65e-05	3.356
	160×160	2.02e-04	2.000	2.82e-03	1.998	1.90e-06	3.115
	320×320	5.05e-05	2.000	7.06e-04	2.000	2.33e-07	3.032
2	20×20	1.94e-01	–	9.92e-01	–	3.08e-01	–
	40×40	5.09e-02	1.931	5.66e-01	0.810	3.16e-02	3.283
	80×80	1.29e-02	1.984	1.73e-01	1.710	2.36e-03	3.747
	160×160	3.23e-03	1.996	4.48e-02	1.948	1.86e-04	3.661
	320×320	8.07e-04	1.999	1.13e-02	1.990	1.80e-05	3.370

Table: L_∞-errors and orders of accuracy at $T = 2$.

Nonlinear equations in uniform grids

Example

$$\phi_t + \frac{1}{2}(\phi_x + \phi_y + 1)^2 = 0, \quad -2 \le x, y \le 2,$$
$$\phi^0(x,y) = -\cos(\pi(x+y)/2).$$

CFL	$N_x \times N_y$	$k=1.\ \beta=1.$		$k=2.\ \beta=0.5.$		$k=3.\ \beta=0.6.$	
		error	order	error	order	error	order
0.5	20×20	5.48e-02	–	1.68e-02	–	6.36e-04	–
	40×40	2.98e-02	0.877	4.72e-03	1.836	5.11e-05	3.637
	80×80	1.63e-02	0.874	1.28e-03	1.880	4.29e-06	3.574
	160×160	8.37e-03	0.961	3.30e-04	1.956	5.13e-07	3.065
	320×320	4.27e-03	0.970	8.46e-05	1.965	6.03e-08	3.091
2	20×20	1.66e-01	–	2.41e-01	–	5.63e-02	–
	40×40	9.94e-02	0.739	5.96e-02	2.017	8.39e-03	2.745
	80×80	5.64e-02	0.817	1.70e-02	1.810	9.15e-04	3.197
	160×160	3.05e-02	0.886	4.74e-03	1.841	6.26e-05	3.871
	320×320	1.63e-02	0.906	1.28e-03	1.886	4.76e-06	3.717

Table: L_∞-errors and orders of accuracy at $T = 0.5/\pi^2$.

Nonlinear equations and grids with random perturbations

Example

$$\phi_t + \frac{1}{2}(\phi_x + \phi_y + 1)^2 = 0, \quad -2 \le x, y \le 2,$$
$$\phi(x, y, 0) = -\cos(\pi(x + y)/2).$$

CFL	$N_x \times N_y$	$k = 1.\ \beta = 1.$		$k = 2.\ \beta = 0.5.$		$k = 3.\ \beta = 0.6.$	
		error	order	error	order	error	order
0.5	40×40	1.09e-01	–	6.43e-02	–	7.76e-03	–
	80×80	6.14e-02	0.835	1.66e-02	1.951	1.00e-03	2.952
	160×160	3.30e-02	0.895	4.47e-03	1.894	7.88e-05	3.669
	320×320	1.71e-02	0.945	1.19e-03	1.916	4.69e-06	4.069
	640×640	8.72e-03	0.975	3.05e-04	1.959	3.36e-07	3.802
2	40×40	2.16e-01	–	3.21e-01	–	1.73e-01	–
	80×80	1.72e-01	0.327	1.98e-01	0.696	4.68e-02	1.888
	160×160	1.10e-01	0.653	6.45e-02	1.621	7.72e-03	2.600
	320×320	6.15e-02	0.835	1.66e-02	1.962	1.00e-03	2.948
	640×640	3.30e-02	0.897	4.46e-03	1.894	7.62e-05	3.716

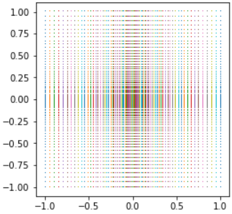
Table: L_∞-errors and orders of accuracy at $T = 0.5/\pi^2$.

Riemann problem in nonuniform grids: CFL=7 for small cells

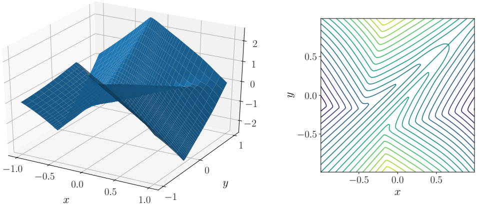
Example

$$\phi_t + \sin(\phi_x + \phi_y) = 0, \quad -1 \le x, y \le 1$$
$$\phi(x, y, 0) = \pi(|y| - |x|).$$

Construct a nonuniform mesh consisting of 60×60 grid points using a geometric series, selecting the ratio between the smallest cell size and the biggest cell size to be $1 : 7$.



(a) Generated mesh



(b) Numerical solutions' surfaces and contour at $T = 1$.

NEW - Multistage Methods vs Multistep Methods (CFL=1)

Example 1 (H-J form: Burgers equation)

$\phi_t + \frac{1}{2}(\phi_x + \phi_y + 1)^2 = 0, \quad -2 \leq x, y \leq 2,$
$\phi^0(x, y) = -\cos(\pi(x + y)/2).$

$N_x \times N_y$	Runge-Kutta 3		Multistep-3	
	error	order	error	order
20×20	5.63e-02	–	1.68e-02	–
40×40	8.39e-03	2.74	3.14e-03	2.42
80×80	9.15e-04	3.19	4.18e-04	2.91
160×160	6.26e-05	3.87	5.29e-05	2.98

Example 2 (Advection-Diffusion equation)

$u_t + u_x + u_y = 0.01(u_{xx} + u_{yy}), \quad -2 \leq x, y \leq 2,$
$u(x, y, 0) = \sin(\pi(x + y)/2).$

$N_x \times N_y$	Runge-Kutta 3		Multistep-3	
	error	order	error	order
20×20	5.81e-01	–	7.38e-02	–
40×40	1.45e-01	2.00	2.03e-02	1.86
80×80	2.07e-02	2.81	2.64e-03	2.94
160×160	2.69e-03	2.94	3.53e-04	2.91

NEW - Multistage Methods vs Multistep Methods (CFL = 2)

Example 1 (H-J form: Burgers equation)

$$\phi_t + \frac{1}{2}(\phi_x + \phi_y + 1)^2 = 0, \quad -2 \leq x, y \leq 2,$$
$$\phi^0(x, y) = -\cos(\pi(x + y)/2).$$

$N_x \times N_y$	Runge-Kutta 3		Multistep-3	
	error	order	error	order
20×20	6.88e-02	–	1.68e-02	–
40×40	5.92e-02	0.22	1.68e-02	0.00
80×80	8.34e-03	2.83	3.16e-03	2.41
160×160	9.21e-04	3.18	4.19e-04	2.92

Example 2 (Advection-Diffusion equation)

$$u_t + u_x + u_y = 0.01(u_{xx} + u_{yy}), \quad -2 \leq x, y \leq 2,$$
$$u(x, y, 0) = \sin(\pi(x + y)/2).$$

$N_x \times N_y$	Runge-Kutta 3		Multistep-3	
	error	order	error	order
20×20	8.03e-01	–	7.60e-02	–
40×40	5.84e-01	0.46	7.38e-02	0.04
80×80	1.45e-01	2.00	2.03e-02	1.86
160×160	2.07e-02	2.81	2.64e-03	2.94

Generalized Kernel Based Methods

Kernel based approximation with arbitrary boundary conditions

From the linear wave equation

$$\partial_t y - c\partial_x y = 0 \quad + B.C.\,, \qquad x \in [a, b],$$

the numerical solution is updated from

$$y^{n+1}(x) = y^{n+1}(b) e^{-\alpha(b-x)} + \alpha \int_x^b e^{-\alpha(\tau - x)} y^n(\tau) d\tau$$

with $\alpha = 1/(c\Delta t)$. Therefore the operators are defined by

$$\mathcal{L}_L = (\mathcal{I} - \frac{1}{\alpha}\partial_x)(\cdot) + B.C. \;, \quad \mathcal{L}_L^{-1}[(\cdot)^n] = B e^{-\alpha(b-x)} + \alpha \int_x^b e^{-\alpha(\tau - x)} (\cdot)^n d\tau$$

with $B := B[(\cdot)^n] = (\cdot)^{n+1}(b)$.

Now, the boundary term can be approximated by Taylor expansion:

$$\begin{aligned} B[y^n] := y^{n+1}(b) &= y^n(b) + \Delta t y_t^n(b) + \mathcal{O}(\Delta t^2) \\ &= y^n(b) + c\Delta t y_x^n(b) + \mathcal{O}(\Delta t^2) \\ &= y^n(b) + \frac{1}{\alpha} y_x^n(b) + \mathcal{O}(\Delta t^2). \end{aligned}$$

Reference:

- A.J. Christlieb, S. Gottlieb, W. Sands and H. Yang, *Generalized kernel based explicit unconditionally stable schemes for arbitrary boundary conditions*, In progress, 2020.

Kernel based approximation with arbitrary boundary conditions

Lemma

Suppose $\phi \in \mathcal{C}^{k+1}[a,b]$ and we set the operator $\mathcal{D}_L$ with general boundary treatments. Then we can obtain that

$$\mathcal{D}_L[\phi](x) = -\sum_{p=1}^{k} \frac{1}{\alpha^p} \partial_x^p \phi(x) + \sum_{p=2}^{k} \frac{1}{\alpha^p} \partial_x^p \phi(b) e^{-\alpha(b-x)} - \frac{1}{\alpha^{k+1}} I_L[\partial_x^{k+1}\phi](x).$$

Consider $k=2$ case:

$$\mathcal{D}_L[\phi](x) = -\frac{1}{\alpha}\phi'(x) - \frac{1}{\alpha^2}\phi''(x) + \frac{1}{\alpha^2}\phi''(b)e^{-\alpha(b-x)} - \frac{1}{\alpha^3} I_L[\phi'''](x).$$

Instead of adding successively defined term $\mathcal{D}_L^p = \mathcal{D}_L[\mathcal{D}_L^{p-1}]$, $p \geq 2$, we define

$$\widetilde{\mathcal{D}}_L[\phi](x) := \mathcal{D}_L[\phi](x) - \frac{1}{\alpha^2}\phi''(b)e^{-\alpha(b-x)}$$

and apply the $\mathcal{D}_L$ operator:

$$\mathcal{D}_L[\widetilde{\mathcal{D}}_L[\phi]](x) = \frac{1}{\alpha^2}\phi''(x) + \frac{1}{\alpha^3} I_L[\phi'''](x) + \frac{1}{\alpha^3} I_L^2[\phi'''](x).$$

Then we obtain higher order approximation to $\phi'(x)$ with the modified partial sum:

$$\mathcal{D}_L[\phi](x) + \mathcal{D}_L[\widetilde{\mathcal{D}}_L[\phi]](x) - \frac{1}{\alpha^2}\phi''(b)e^{-\alpha(b-x)} = -\frac{1}{\alpha}\phi'(x) + \mathcal{O}(\frac{1}{\alpha^3}).$$

Kernel based approximation with arbitrary boundary conditions

Using the general boundary treatments, we can modify the partial sums for the first derivative operators:

$$\phi_x^+(x) \approx \mathcal{P}_k^L[\phi](x) = -\alpha \sum_{p=1}^{k} \widetilde{\mathcal{D}}_L^p[\phi](x) + \frac{1}{\alpha}\partial_x^2 \phi(b) e^{-\alpha(b-x)}$$

where

$$\begin{cases} \widetilde{\mathcal{D}}_L^1 = \mathcal{D}_L, \\ \widetilde{\mathcal{D}}_L^p = \mathcal{D}_L[\widetilde{\mathcal{D}}_L^{p-1}] - \left(\frac{1}{\alpha}\right)^p \partial_x^p \phi(b) e^{-\alpha(b-x)}, \quad p \geq 2, \end{cases}$$

It can be easily derived for the case $\phi_x^-(x) \approx \mathcal{P}_k^R[\phi](x)$ by a similar process.

Theorem

Suppose $\phi \in \mathcal{C}^{k+1}[a, b]$, $k = 1, 2, 3$. Then, the modified partial sums satisfy

$$\|\partial_x \phi(x) - \mathcal{P}_k^*[\phi](x)\|_\infty \leq C \left(\frac{1}{\alpha}\right)^k \|\partial_x^{k+1}\phi(x)\|_\infty$$

where $*$ indicates L and R operators.

Numerical Results (Part 2)

Linear equations

Example

$$\phi_t + (\phi_x) = 0, \quad -1 \le x \le 1,$$
$$\phi(x,0) = -\cos(\pi x).$$

Table: (Periodic boundary conditions) L_∞-errors and orders of accuracy at $T = 1$.

CFL	N	$k=1.$		$k=2.$		$k=3.$	
		error	order	error	order	error	order
0.5	40	3.07e-01	–	7.92e-02	–	8.01e-03	–
	80	1.69e-01	0.864	2.07e-02	1.933	9.51e-04	3.074
	160	8.84e-02	0.933	5.27e-03	1.976	1.13e-04	3.067
	320	4.52e-02	0.967	1.33e-03	1.990	1.38e-05	3.042
	640	2.29e-02	0.983	3.33e-04	1.995	1.69e-06	3.023

Table: (Dirichlet boundary conditions) L_∞-errors and orders of accuracy at $T = 1$.

CFL	N	$k=1.$		$k=2.$		$k=3.$	
		error	order	error	order	error	order
0.5	40	3.06e-01	–	7.93e-02	–	7.11e-03	–
	80	1.68e-01	0.862	2.06e-02	1.942	8.31e-04	3.097
	160	8.83e-02	0.931	5.23e-03	1.981	9.81e-05	3.081
	320	4.52e-02	0.966	1.31e-03	1.992	1.18e-05	3.050
	640	2.29e-02	0.983	3.29e-04	1.997	1.45e-06	3.027

Nonlinear equations

Example

$$\phi_t + \frac{1}{2}(\phi_x + 1)^2 = 0, \quad -1 \le x \le 1,$$
$$\phi(x,0) = -\cos(\pi x).$$

Table: (Periodic boundary) L_∞-errors and orders of accuracy at $T = 0.3/\pi^2$.

CFL	N	$k=1.$ error	$k=1.$ order	$k=2.$ error	$k=2.$ order	$k=3.$ error	$k=3.$ order
0.5	40	3.52e-02	–	2.81e-03	–	2.30e-03	–
	80	6.59e-03	2.419	1.15e-03	1.285	4.03e-04	2.515
	160	3.30e-03	0.997	4.00e-04	1.525	5.73e-05	2.814
	320	1.66e-03	0.996	1.26e-04	1.665	7.60e-06	2.914
	640	8.30e-04	0.995	3.43e-05	1.881	9.61e-07	2.983

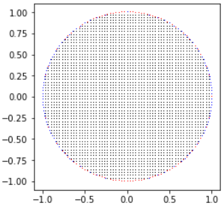
Table: (Dirichlet boundary) L_∞-errors and orders of accuracy at $T = 0.3/\pi^2$.

CFL	N	$k=1.$ error	$k=1.$ order	$k=2.$ error	$k=2.$ order	$k=3.$ error	$k=3.$ order
0.5	40	3.47e-02	–	2.81e-03	–	1.77e-03	–
	80	6.59e-03	2.395	1.01e-03	1.476	2.85e-04	2.634
	160	3.30e-03	0.997	3.76e-04	1.424	3.95e-05	2.848
	320	1.66e-03	0.996	1.11e-04	1.758	5.07e-06	2.962
	640	8.30e-04	0.995	3.02e-05	1.878	6.28e-07	3.012

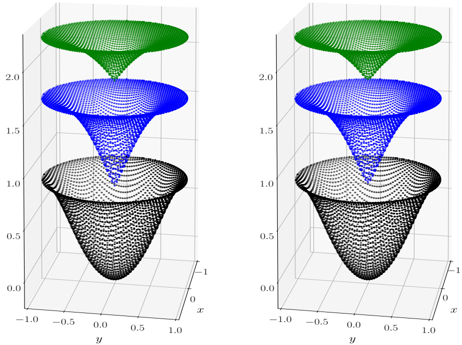
A propagating problem with Dirichlet boundaries in nonuniform grids

Example

$\phi_t - \sqrt{\phi_x^2 + \phi_y^2 + 1} = 0, \quad x^2 + y^2 \leq 1$
$\phi(x, y, 0) = \sin\left(\frac{\pi}{2}(x^2 + y^2)\right)$
with the Dirichlet boundary $\phi(x, y, t) = 1 + t$ for all $x^2 + y^2 = 1$.



(a) Domain discretization



CFL $= 0.5$

CFL $= 2$

(b) Propagating solutions at T=0, 0.6 and 1.2

Magnetohydrodynamics Examples

Magnetohydrodynamics equations

Consider the MHD equations

$$\partial_t \begin{bmatrix} \rho \\ \rho \mathbf{u} \\ \mathcal{E} \\ \mathbf{B} \end{bmatrix} + \nabla \cdot \begin{bmatrix} \rho \mathbf{u} \\ \rho \mathbf{u} \otimes \mathbf{u} + p_{\text{tot}} \mathbb{I} - \mathbf{B} \otimes \mathbf{B} \\ \mathbf{u}(\mathcal{E} + p_{\text{tot}}) - \mathbf{B}(\mathbf{u} \cdot \mathbf{B}) \\ \mathbf{u} \otimes \mathbf{B} - \mathbf{B} \otimes \mathbf{u} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \nabla \times (\eta \mathbf{J} + \frac{1}{ne} \mathbf{J} \times \mathbf{B}) \end{bmatrix}$$

$$\nabla \cdot \mathbf{B} = 0,$$

with the equation of state as

$$\mathcal{E} = \frac{p}{\gamma - 1} + \frac{\rho \|\mathbf{u}\|^2}{2} + \frac{\|\mathbf{B}\|^2}{2}.$$

Methods to overcome $\nabla \cdot B = 0$:
The projection method, 8-wave scheme, hyperbolic divergence cleaning method, the constrained transport method.

Reference:

- A.J. Christlieb, F. Cakir and Y. Jiang, *A Kernel Based High Order "Explicit" Unconditionally Stable Constrained Transport Method for Ideal Magnetohydrodynamics*, Revised, 2020.
- A.J. Christlieb, X. Feng, Y. Jiang and Q. Tang, *A high-order finite difference WENO scheme for ideal magnetohydrodynamics on curvilinear meshes*, SIAM J. Sci. Comput. (2017)

Magnetic vector potential formulation

Observe: $\mathbf{B}$ is a field, therefore there exists an $\mathbf{A}$, magnetic vector potential, such that $\nabla \times \mathbf{A} = \mathbf{B}$.

Goal: Express $\mathbf{B}$ in MHD in terms of $\mathbf{A}$.

Start with Electron inertial equation, solve for $\mathbf{E}$:

$$\mathbf{E} = -\mathbf{u} \times \mathbf{B} + \eta \mathbf{J} + \frac{1}{ne}\mathbf{J} \times \mathbf{B} + \frac{m_e}{ne^2}\partial_t \mathbf{J}.$$

In Maxwell, replace $\mathbf{E}$ in $\partial_t \mathbf{B} = -\nabla \times \mathbf{E}$, with the above,

$$\partial_t \mathbf{B} = \nabla \times \left(\mathbf{u} \times \mathbf{B} - \left(\eta \mathbf{J} + \frac{1}{ne}\mathbf{J} \times \mathbf{B} + \frac{m_e}{ne^2}\partial_t \mathbf{J}\right)\right).$$

Noting that $\nabla \times \mathbf{A} = \mathbf{B}$ and using the Weyl gauge,

$$\partial_t \mathbf{A} + (\nabla \times \mathbf{A}) \times \mathbf{u} = -\left(\eta \mathbf{J} + \frac{1}{ne}\mathbf{J} \times \nabla \times \mathbf{A} + \frac{m_e}{ne^2}\partial_t \mathbf{J}\right).$$

Reference:

- A.J. Christlieb, J.A. Rossmanith and Q. Tang, *Finite difference weighted essentially non-oscillatory schemes with constrained transport for ideal magnetohydrodynamics*, J. Comput. Phys. (2014).

Example1: 2D smooth vortex problem

The initial conditions are

$$(\rho, u^1, u^3, u^3, p, B^1, B^2, B^3) = (1, 1, 1, 0, 1, 0, 0, 0)$$

with perturbations on u^1, u^2, B^1, B^2 and p as :

$$\begin{aligned}
(\delta u^1, \delta u^2) &= \frac{\mu}{2\pi} e^{0.5(1-r^2)}(-y, x),\\
(\delta B^1, \delta B^2) &= \frac{\kappa}{2\pi} e^{0.5(1-r^2)}(-y, x),\\
\delta p &= \frac{\mu^y(1-r^2) - \kappa^2}{8\pi^2} e^{(1-r^2)},
\end{aligned}$$

and the initial magnetic potential is

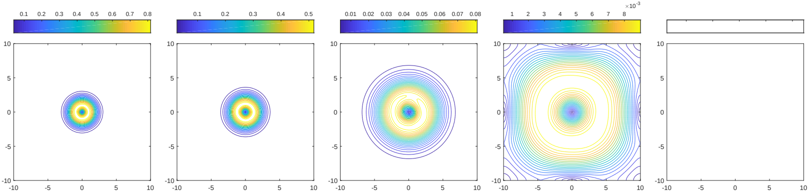
$$A^3(0, x, y) = \frac{\mu}{2\pi} e^{0.5(1-r^2)},$$

where $r^2 = x^2 + y^2$. The vortex strength is taken as $\mu = 5.389489439$ and $\kappa = \sqrt{2}\mu$ such that the lowest pressure is around 5.3×10^{-12} which happens in the center of the vortex. The domain is $[-10, 10] \times [-10, 10]$ and periodic boundary condition is used on all four boundaries.

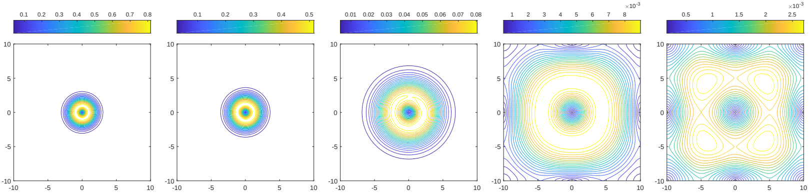
Example1: 2D smooth vortex in resistive MHD

- Solve 2D smooth vortex problem with several resistive terms η.
- Contour plots of $\|\mathbf{B}\|$ at time $t = 20$ with $\Delta x = \Delta y = 0.2$:

(i) FD scheme



(ii) Kernel based scheme



(a) $\eta = 0$ (b) $\eta = 0.01$ (c) $\eta = 0.1$ (d) $\eta = 0.5$ (e) $\eta = 1$

Example2: 3D Blast wave

The initial conditions are

$$(\rho, u^1, u^2, u^3, B^1, B^2, B^3) = (1, 0, 0, 0, 50/\sqrt{2\pi}, 50/\sqrt{2\pi}, 0)$$

with a spherical pressure pulse

$$p = \begin{cases} 1000 & r \leq 0.1 \\ 0.1 & \text{otherwise.} \end{cases}$$

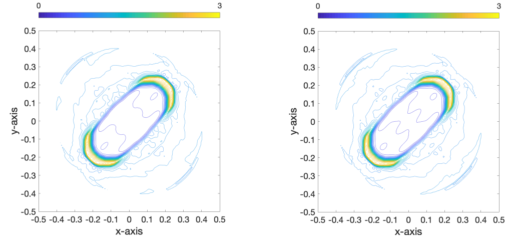
where $r = \sqrt{x^2 + y^2 + z^2}$. The initial condition for magnetic potential :

$$\mathbf{A}(0, x, y, z) = (0, 0, 50/\sqrt{2\pi}y - 50/\sqrt{2\pi}x).$$

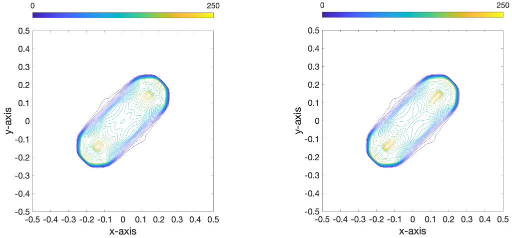
We use domain $[-0.5, 0.5] \times [-0.5, 0.5] \times [-0.5, 0.5]$ and outflow boundary conditions are applied everywhere.

Example2: 3D Blast wave

(i) Density contour plots at $t = 0.01$ with $150 \times 150 \times 150$ grid point



(ii) Pressure contour plots at $t = 0.01$ with $150 \times 150 \times 150$ grid point



(a) FD scheme (b) Kernel based scheme

THANK YOU!!

Appendix

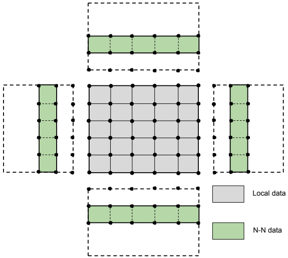
Parallel Algorithm

Domain Decomposition

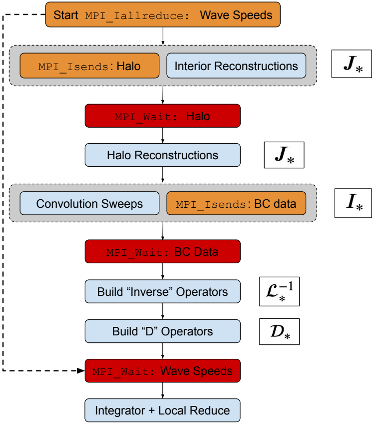
$$\mathcal{D}_L = \mathcal{I} - \mathcal{L}_L^{-1}, \quad \mathcal{D}_R = \mathcal{I} - \mathcal{L}_R^{-1},$$

with

$$\mathcal{L}_L^{-1}[v](x_i) = \alpha \int_{x_i}^{b} e^{-\alpha(y-x_i)} v(y)dy + Be^{-\alpha(b-x)},$$
$$\mathcal{L}_R^{-1}[v](x_i) = \alpha \int_{a}^{x_i} e^{-\alpha(x_i-y)} v(y)dy + Ae^{-\alpha(x-a)}.$$



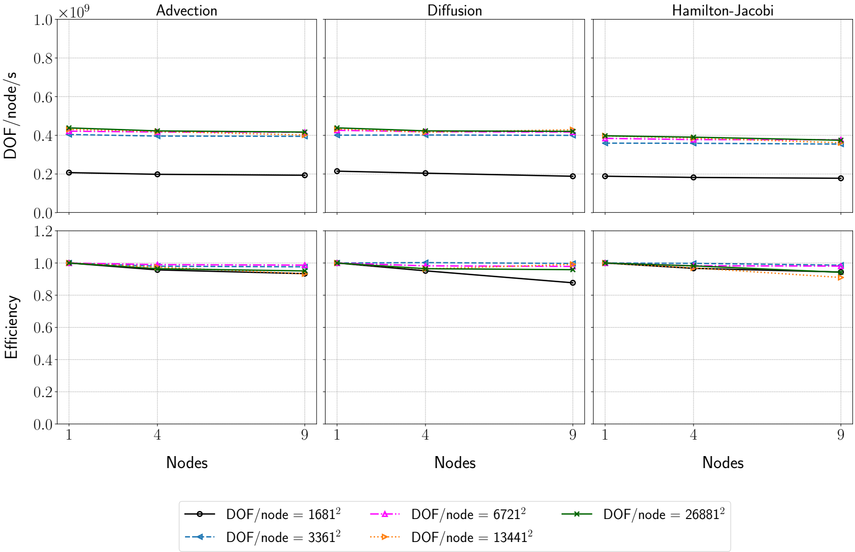
Non-blocking All Reduce Parallel Code Diagram



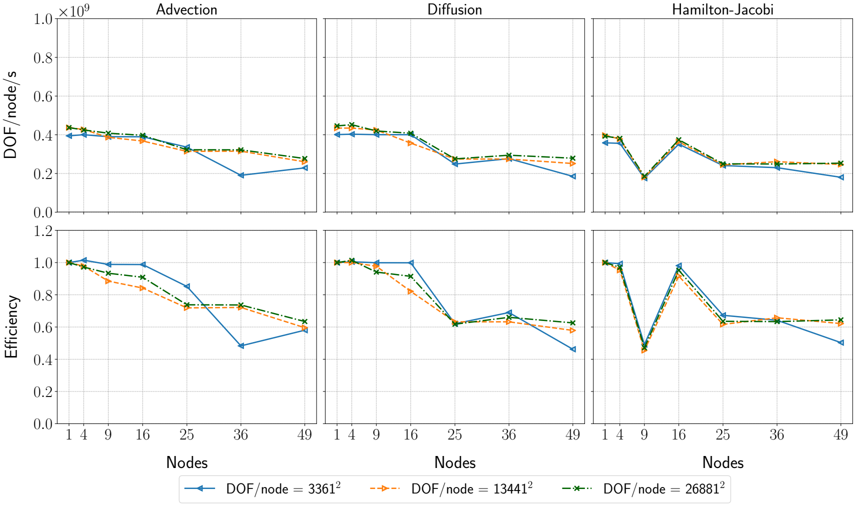
Reference:

- A.J. Christlieb, P.T. Guthrey, W.A. Sands and M. Thavappiragasm, "Parallel Algorithms for Successive Convolution", *arXiv and submitted*

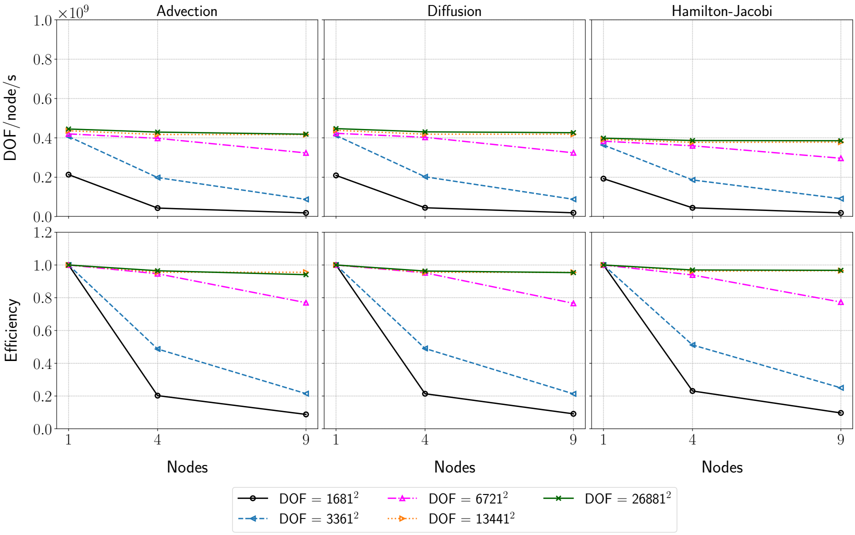
Weak scaling results: up to 9 nodes



Weak scaling results: up to 49 nodes



Strong scaling results: up to 9 nodes



Example3: Hartmann problem with PEC boundary conditions

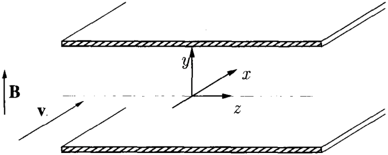
The Hartmann flow model for MHD is taken to be incompressible ($\nabla \cdot \mathbf{v} = 0$), and $\rho = 1, p = 1$. The momentum and magnetic field equations are given as

$$\rho\Big[\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla)\mathbf{v}\Big] = -\nabla p + \mathbf{f} + N(\mathbf{j} \times \mathbf{B}) + \frac{1}{R}\nabla^2 \mathbf{v},$$
$$\frac{\partial \mathbf{B}}{\partial t} + (\mathbf{v} \cdot \nabla)\mathbf{B} = (\mathbf{B} \cdot \nabla)\mathbf{v} + \frac{1}{Rm}\nabla^2 \mathbf{B}.$$

With the assumptions of linearity in above equations, we have that

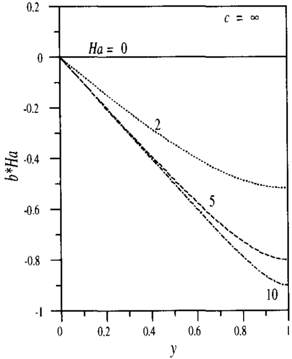
$$\mathbf{v} = v_x(y)\hat{\mathbf{x}} \quad \text{and} \quad \mathbf{B} = \hat{\mathbf{y}} + \frac{Rm}{Ha} B_x(y)\hat{\mathbf{x}}.$$

The geometry of the problem consists of two infinite parallel plates and there is flow in the gap between the plates.

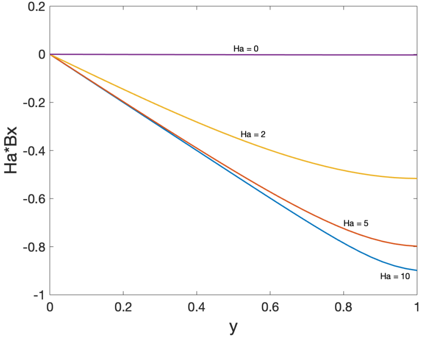


Example3: Hartmann problem with PEC boundary conditions

Figure: Induced magnetic field for the Hartmann problem at $Ha = 0, 2, 5, 10$.



(a) [Muller]



(b) Kernel based scheme

Reference:

- U. Müller and L. Bühler, *Magnetofluiddynamics in channels and containers*, Springer Science & Business Media (2013)