Kernel Based High Order Schemes for Vlasov Simulations and Other Time Dependent Problems with Non-smooth Solutions

Wei Guo

Michigan State University
Department of Mathematics

SIAM CSE17, Mar., 2017

Joint work with Andrew Christlieb and Yan Jiang (MSU)

Outline

- Method of lines transpose approach for Vlasov simulations
- A new formulation for general nonlinear time dependent problems
- Summary and future work

Method of lines (MOL)

$$\textit{Space discretization} \Rightarrow \textit{time evolution}$$

- explicit:
  - easy to implement
  - restriction on time step
- implicit:
  - larger time step
  - need to solve the system

Method of lines transpose (MOL^T)

As opposed to MOL approaches, the method of lines transpose (MOL^T) schemes

- discretize in time first;
- solve the resulting boundary value problem (BVP) at discrete time levels;
- also known as Rothe's method Schemann, Bornemann (98), Salazar et al. (00).

One-dimensional advection equation

$$u_t + cu_x = 0, \quad x \in [a, b]. \tag{1}$$

- periodic boundary condition
- Dirichlet boundary condition

$$u(a, t) = g_1(t), \text{ for } c > 0, \quad \text{or} \quad u(b, t) = g_2(t), \text{ for } c < 0.$$

- Neumann boundary condition

$$u_x(a, t) = h_1(t), \text{ for } c > 0, \quad \text{or} \quad u_x(b, t) = h_2(t), \text{ for } c < 0.$$

- uniform mesh $a = x_0 < x_1 < \cdots < x_{M-1} < x_M = b$ with $\Delta x = \frac{(b-a)}{M}$.

MOL^T framework

- backward Euler: $\frac{u^{n+1}-u^n}{\Delta t} + c u_x^{n+1} = 0.$
- BVP ($c > 0$)

$$\mathcal{L}_L[\alpha](u^{n+1}) = (\mathcal{I} + \frac{1}{\alpha}\partial_x)u^{n+1} = u^n.$$

$$u^{n+1}(x) = \mathcal{L}_L^{-1}[\alpha](u^n) = I^L[u^n,\alpha](x) + A^{n+1}e^{-\alpha(x-a)}, \qquad (2)$$

where $\alpha = 1/(c\Delta t)$, and

$$I^L[u^n,\alpha](x) \doteq \alpha \int_a^x e^{-\alpha(x-y)} u^n(y)dy. \qquad (3)$$

Remark: The scheme is implicit. But we do not need to invert the linear matrix.

Recursive form

Let $I_i^L = I^L[u^n, \alpha](x_i)$:

$$I_i^L = I_{i-1}^L e^{-\alpha \Delta x} + J_i^L, \quad i = 1, \cdots, M, \quad I_0^L = 0, \tag{4}$$

where

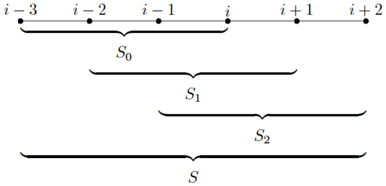
$$J_i^L \doteq \alpha \int_{x_{i-1}}^{x_i} u^n(y) e^{-\alpha(x_i - y)} dy.$$

Remark 1: The recursive form is developed in Causley, Christlieb, Guclu, Wolf (13).

Remark 2: The existing MOL^T schemes mainly use linear interpolation (quadrature) methods to compute J_i^L, which work well for smooth problems.

Weighted essentially non-oscillatory (WENO)

$J_i^L \doteq \alpha \int_{x_{i-1}}^{x_i} u^n(y) e^{-\alpha(x_i - y)} dy$:



- small stencil S_r: $u^n(x) \Rightarrow p_r(x) \Rightarrow J_{i,r}^L = \sum_{j=0}^{2} c_{i-3+r+j}^{(r)} u_{i-3+r+j}^n$
- big stencil S: $u^n(x) \Rightarrow p(x) \Rightarrow J_i^L = \sum_{r=0}^{2} d_r J_{i,r}^L$
- nonlinear weights: $d_r \Rightarrow \omega_r$ by smoothness indicators

$$\omega_r = d_r + O(\Delta x^2) \text{ or } \omega_r = \begin{cases} O(1), & u^n(x) \text{ is smooth in } S_r \\ O(\Delta x^4), & u^n(x) \text{ has a discontinuity inside } S_r \end{cases}$$

- final result

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

High order time discretization

$$u_t = F(u)$$

Strong-stability-preserving (SSP) diagonally implicit Runge-Kutta (DIRK) methods: RK(s,k)

$$u^{(i)} = u^n + \Delta t \sum_{j=1}^{i} a_{ij} F(u^{(j)}, t_n + c_j \Delta t), \quad i \le s, \tag{5a}$$

$$u^{n+1} = u^n + \Delta t \sum_{j=1}^{s} b_j F(u^{(j)}, t_n + c_j \Delta t). \tag{5b}$$

Splitting framework

$$u_t + f(y,t)u_x + g(x,t)u_y = 0 \Rightarrow \begin{cases} u_t + f(y,t)u_x &= 0, \\ u_t + g(x,t)u_y &= 0. \end{cases}$$

- The fourth order splitting is

$$\begin{aligned} u^{n+1} = & Q_2((\alpha + 1/2)\Delta t) \cdot Q_1((2\alpha+1)\Delta t) \cdot Q_2(-\alpha \Delta t) \cdot Q_1(-(4\alpha+1)\Delta t) \cdot \\ & Q_2(-\alpha\Delta t) \cdot Q_1((2\alpha+1)\Delta t) \cdot Q_2((\alpha+1/2)\Delta t) u^n, \end{aligned}$$

where $\alpha = (2^{1/3} + 2^{-1/3} - 1)/6$.

Rigid body rotation

$$u_t + yu_x - xu_y = 0, \quad x, y \in \Omega \tag{6}$$

- continuous initial condition, with $\Omega = [-\frac{1}{2}\pi, \frac{1}{2}\pi]^2$

$$u(x, y, 0) = 0.5B(\sqrt{x^2 + 8y^2}) + 0.5B(\sqrt{8x^2 + y^2}),$$

where $B(r) = \begin{cases} \cos(r)^6, & \text{if } r \le \frac{1}{2}\pi, \\ 0, & \text{otherwise.} \end{cases}$

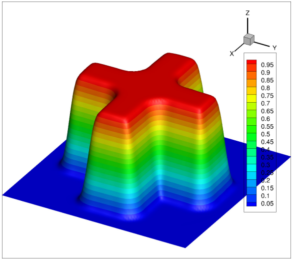
- dicontinuous initial condition, with $\Omega = [-1, 1]^2$

$$u(x, y, 0) = \begin{cases} 1, & (x, y) \in [-0.75, 0.75] \times [-0.25, 0.25] \bigcup \\ & [-0.25, 0.25] \times [-0.75, 0.75], \\ 0, & \text{otherwise.} \end{cases}$$

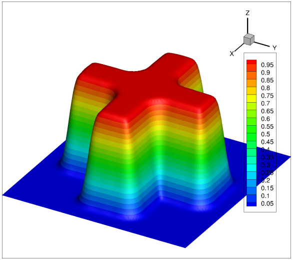
- 5-th order WENO integration. RK(4,4) with $CFL = 2.9$.

Table: $T = 2\pi$.

		$N_x \times N_y$	L_1 errors	order	L_∞ error	order	min value
Periodic boundary condition.	Without PP-limiters	40×40	6.26E-02	–	7.43E-02	–	-1.21E-03
		80×80	5.10E-03	3.62	7.49E-03	3.31	-6.88E-05
		160×160	1.53E-04	5.05	2.75E-04	4.77	-4.76E-06
		320×320	4.13E-06	5.22	7.56E-06	5.18	-1.03E-07
	With PP-limiters	40×40	6.19E-02	–	7.43E-02	–	1.00E-16
		80×80	5.06E-03	3.61	7.50E-03	3.31	1.00E-16
		160×160	1.52E-04	5.06	2.75E-04	4.77	1.00E-16
		320×320	4.10E-06	5.21	7.56E-06	5.18	1.00E-16
Dirichlet boundary condition.	Without PP-limiters	40×40	6.08E-02	–	7.43E-02	–	-1.67E-05
		80×80	4.98E-03	3.61	7.49E-03	3.31	-2.50E-05
		160×160	1.34E-04	5.22	2.75E-04	4.77	-4.76E-06
		320×320	3.73E-06	5.16	7.56E-06	5.18	-1.03E-07
	With PP-limiters	40×40	6.09E-02	–	7.43E-02	–	0.00E+00
		80×80	4.98E-03	3.61	7.49E-03	3.31	0.00E+00
		160×160	1.34E-04	5.22	2.75E-04	4.77	0.00E+00
		320×320	3.72E-06	5.17	7.56E-06	5.18	0.00E+00
Neumann boundary condition.	Without PP-limiters	40×40	6.08E-02	–	7.43E-02	–	-1.66E-05
		80×80	4.98E-03	3.61	7.49E-03	3.31	-2.50E-05
		160×160	1.34E-04	5.22	2.75E-04	4.77	-4.76E-06
		320×320	3.73E-06	5.16	7.56E-06	5.18	-1.03E-07
	With PP-limiters	40×40	6.09E-02	–	7.43E-02	–	0.00E+00
		80×80	4.98E-03	3.61	7.49E-03	3.31	0.00E+00
		160×160	1.34E-04	5.22	2.75E-04	4.77	0.00E+00
		320×320	3.72E-06	5.17	7.56E-06	5.18	0.00E+00



(a) Periodic boundary condition.



(b) Dirichlet boundary condition.

Figure: $T = 2\pi$.

Vlasov-Poisson (VP) system

$$f_t + \mathbf{v} \cdot \nabla_{\mathbf{x}} f + \mathbf{E}(\mathbf{x}, t) \cdot \nabla_{\mathbf{v}} f = 0, \quad \mathbf{x} \times \mathbf{v} \in \Omega_{\mathbf{x}} \times \Omega_{\mathbf{v}} \tag{7a}$$

$$\mathbf{E}(\mathbf{x}, t) = -\nabla_{\mathbf{x}} \phi(\mathbf{x}, t), \quad -\Delta_{\mathbf{x}} \phi(\mathbf{x}, t) = \rho(\mathbf{x}, t) - 1 \tag{7b}$$

- describe the dynamics of charged particles due to the self-consistent electric force
- $f(\mathbf{x}, \mathbf{v}, t)$: probability of finding a particle with velocity $\mathbf{v}$ at position $\mathbf{x}$ at time t
- $\mathbf{E}$: electrostatic field
- ϕ: self-consistent electrostatic potential
- $\rho(\mathbf{x}, t)$: electron charge density $\rho(\mathbf{x}, t) = \int_{\Omega_{\mathbf{v}}} f(\mathbf{x}, \mathbf{v}, t) d\mathbf{v}$

The VP systems

- Strong Landau damping:

$$f(x,v,0) = \frac{1}{\sqrt{2\pi}}(1+\alpha\cos(kx))\exp(-\frac{v^2}{2}),$$

$x \in [0, L]$, $v \in [-V_c, V_c]$, where $\alpha = 0.5$, $k = 0.5$, $L = 4\pi$ and $V_c = 2\pi$.

- Two-stream instability I:

$$f(x,v,0) = \frac{2}{7\sqrt{2\pi}}(1+5v^2)(1+\alpha((\cos(2kx)+\cos(3kx))/1.2+cos(kx)))\exp(-\frac{v^2}{2}),$$

$x \in [0, L]$, $v \in [-V_c, V_c]$, where $\alpha = 0.01$, $k = 0.5$, $L = 4\pi$ and $V_c = 2\pi$.

- Two-stream instability II:

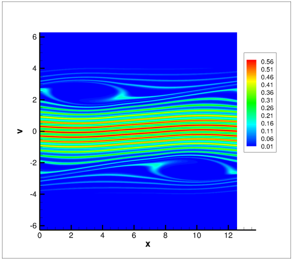
$$f(x,v,0) = \frac{1}{\sqrt{2\pi}}(1+\alpha\cos(kx))v^2\exp(-\frac{v^2}{2}),$$

$x \in [0, L]$, $v \in [-V_c, V_c]$, where $\alpha = 0.05$, $k = 0.5$, $L = 4\pi$ and $V_c = 2\pi$.

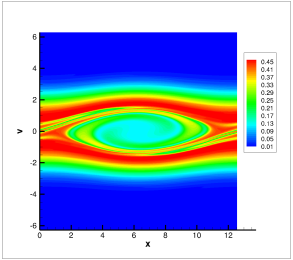
- Bump-on-tail instability:

$$f(x,v,0) = \frac{1}{\sqrt{2\pi}}(1+\alpha\cos(kx))(0.9\exp(-0.5v^2)+0.2\exp(-4(v-4.5)^2))$$

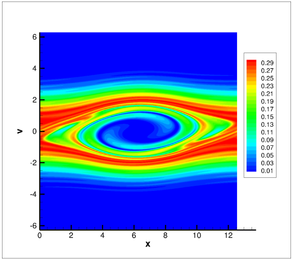
$x \in [-L, L]$, $v \in [-V_c, V_c]$, where $\alpha = 0.04$, $k = 0.3$, $L = \frac{10}{3}\pi$ and $V_c = 10$.



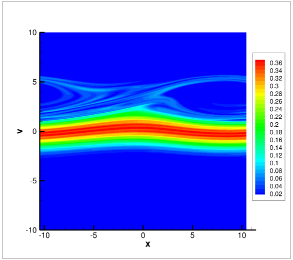
(a) Strong Landau Damping. T=40.



(b) Two-stream instability I. T=40.

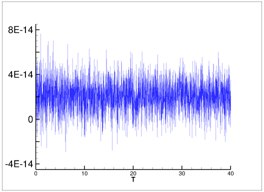


(c) Two-stream instability II. T=40.

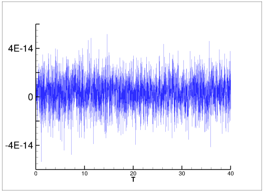


(d) Bump-on-tail instability. T=60.

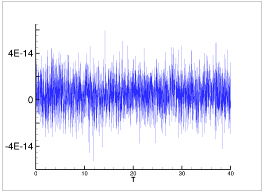
The time evolution of relative deviation in L^1 norm



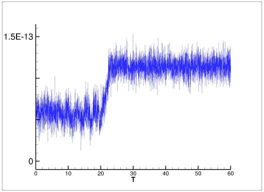
(e) Strong Landau damping



(f) Two-stream instability I



(g) Two-stream instability II



(h) Bump-on-tail instability

New differential operators

Define

$$\mathcal{L}_L = \mathcal{I} + \frac{1}{\alpha}\partial_x, \quad \mathcal{D}_L = \mathcal{I} - \mathcal{L}_L^{-1}, \tag{8a}$$

$$\mathcal{L}_R = \mathcal{I} - \frac{1}{\alpha}\partial_x, \quad \mathcal{D}_R = \mathcal{I} - \mathcal{L}_R^{-1}, \tag{8b}$$

where, $\alpha > 0$ is a constant. $\mathcal{L}_L^{-1}$ and $\mathcal{L}_R^{-1}$ can be computed via the WENO method we discussed.
Then,

$$\frac{1}{\alpha}\partial_x = \mathcal{L}_L(\mathcal{I} - \mathcal{L}_0^{-1}) = \mathcal{D}_L/(\mathcal{I} - \mathcal{D}_L) = \sum_{p=1}^{\infty} \mathcal{D}_L^p, \tag{9a}$$

$$\frac{1}{\alpha}\partial_x = \mathcal{L}_R(\mathcal{L}_R^{-1} - \mathcal{I}) = -\mathcal{D}_R/(\mathcal{I} - \mathcal{D}_R) = -\sum_{p=1}^{\infty} \mathcal{D}_R^p, \tag{9b}$$

- Successive convolution Causley, Christlieb (14).

Second order derivative

$$\mathcal{L}_0 = \mathcal{I} - \frac{1}{\alpha^2}\partial_{xx}, \quad \mathcal{D}_0 = \mathcal{I} - \mathcal{L}_0^{-1} \tag{10a}$$

$$\frac{1}{\alpha^2}\partial_{xx} = \mathcal{L}_0(\mathcal{L}_0^{-1} - \mathcal{I}) = -\mathcal{D}_0/(\mathcal{I} - \mathcal{D}_0) = -\sum_{p=1}^{\infty} \mathcal{D}_0^p \tag{11}$$

and

$$\mathcal{L}_0^{-1}[v,\alpha](x) = \frac{\alpha}{2}\int_a^b e^{-\alpha|x-y|}v(y)dy + A_0 e^{-\alpha(x-a)} + B_0 e^{-\alpha(b-x)} \tag{12}$$

where A_0, B_0 are obtained through boundary conditions. We can evaluate the convolution integral based on the WENO scheme as well.

k^{th} order scheme

Consider the nonlinear convection-diffusion equation:

$$u_t = -f(u)_x + g(u)_{xx}.$$

Then k^{th} order scheme is obtained by truncating the series with the first k terms. In particular,

$$-f(u)_x \approx -\frac{\gamma}{c\Delta t}\sum_{p=1}^{k}\mathcal{D}_L^p[f^+(u), \frac{\gamma}{c\Delta t}](x) + \frac{\gamma}{c\Delta t}\sum_{p=1}^{k}\mathcal{D}_R^p[f^-(u), \frac{\gamma}{c\Delta t}](x),$$

and

$$g(u)_{xx} \approx -\frac{\gamma}{b\Delta t}\sum_{p=1}^{k}\mathcal{D}_0^p[g(u), \sqrt{\frac{\gamma}{b\Delta t}}](x),$$

where $\gamma > 0$ is a parameter associated with the stability of the scheme, e.g., $\gamma_{max} = 0.4167$ for $k = 3$. $b = \max_u |g'(u)|$, $c = \max_u |f'(u)|$, and $f^{\pm}(u)$ are obtained by flux splitting of $f(u)$. Then the scheme is k^{th} order.

Time discretization

- We use a third order Runge-Kutta numerical scheme for time discretization.
- In the numerical tests, we choose the time step as

$$\Delta t = CFL \times \Delta x / \max(b, c).$$

Linear advection-diffusion equation

Consider

$$\begin{cases} u_t + cu_x = bu_{xx}, & x \in [-\pi, \pi] \\ u(x,0) = \sin(x) \end{cases}$$

with $b = 0.01$ and $c = 1$.

Table: *Accuracy test: errors and orders of accuracy at $T = 2$. Third order scheme.*

CFL	N_x	L_1 errors	order	L_∞ error	order
1	40	6.00E-02	–	9.43E-02	–
	80	8.81E-03	2.77	1.38E-02	2.77
	160	1.16E-03	2.93	1.82E-03	2.93
	320	1.46E-04	2.99	2.30E-04	2.99
	640	1.84E-05	3.00	2.88E-05	3.00
5	40	4.36E-01	–	6.85E-01	–
	80	3.34E-01	0.38	5.25E-01	0.38
	160	1.06E-01	1.65	1.67E-01	1.65
	320	1.67E-02	2.67	2.63E-02	2.67
	640	2.23E-03	2.91	3.50E-03	2.91

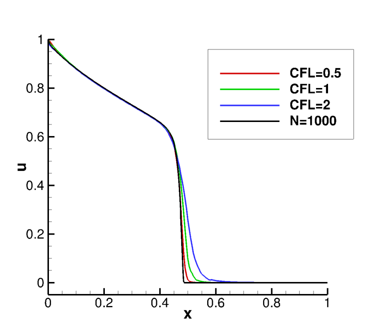
Buckley-Leverett equation

Consider

$$u_t + \left(\frac{u^2}{u^2 + (1-u)^2}\right)_x = 0.01(\nu(u)u_x)_x. \qquad (13)$$

In the numerical simulation, we choose $\epsilon = 0.01$, and

$$\nu(u) = \begin{cases} 4u(1-u), & 0 \le u \le 1 \\ 0, & \textit{otherwise} \end{cases}, \quad u(x,0) = \begin{cases} 1-3x, & 0 \le x \le 1/3 \\ 0, & 1/3 < x \le 1 \end{cases}$$



(i) $k = 3$. $T = 0.2$. $N = 200$.

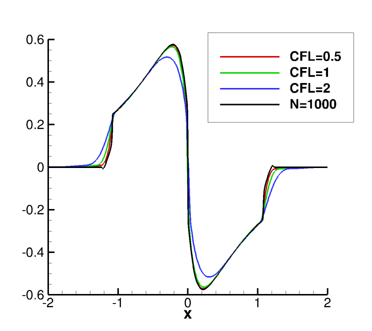
Strong degenerate parabolic convection-diffusion equation

Consider

$$u_t + (u^2)_x = 0.1(\nu(u)u_x)_x \tag{14}$$

where

$$\nu(u) = \begin{cases} 0, & |u| \le 0.25 \\ 1, & |u| > 0.25 \end{cases} \quad u(x,0) = \begin{cases} 1, & -\frac{1}{\sqrt{2}} - 0.4 < x < -\frac{1}{\sqrt{2}} + 0.4 \\ -1, & \frac{1}{\sqrt{2}} - 0.4 < x < \frac{1}{\sqrt{2}} + 0.4 \\ 0, & \textit{otherwise} \end{cases}$$



(j) $k = 3$. $T = 0.2$. $N = 400$.

Conclusion and future work

- The proposed high order kernel based on algorithm is very efficient to simulate the Vlasov equation and other time dependent problems including the nonlinear degenerate convection-diffusion equations.
- The extension to systems and more complex models is under development.

Thank You!