

Exponential methods for solving hyperbolic problems with application to kinetic equations

N. Crouseilles ^{1,2} L. Einkemmer ³ J. Massot ^{2,1}

¹Inria Rennes – Bretagne Atlantique

²IRMAR, Université de Rennes

³University of Innsbruck

December 4, 2020

- 1 Motivation for Vlasov-Poisson equations
- 2 Linear analysis
 - Lawson methods
 - Exponential Runge-Kutta methods
- 3 Numerical simulation: Vlasov-Poisson equations
- 4 Conclusion

- 1 Motivation for Vlasov-Poisson equations
- 2 Linear analysis
 - Lawson methods
 - Exponential Runge-Kutta methods
- 3 Numerical simulation: Vlasov-Poisson equations
- 4 Conclusion

Vlasov-Poisson equations 1D×1D

Our model: a non-linear transport in $(x, v) \in \Omega \times \mathbb{R}$ of an electron density distribution $f = f(t, x, v)$:

$$\begin{cases} \partial_t f + v \partial_x f + E \partial_v f = 0 \\ \partial_x E = \int_{\mathbb{R}} f \, dv - 1 \end{cases}$$

Motivation:

We want high order methods in (x, v)

We want high order methods in time t :

Splitting methods: could have a lot of steps

Runge-Kutta methods: stability constraints (CFL condition)

The most restrictive CFL condition is associated with the linear part ($\partial_t f + v \partial_x f = 0$)

→ We want to propose a compromise: exponential integrators.

Vlasov-Poisson equations 1D×1D

Fourier transform in x direction of Vlasov, amenable to exponential integrators:

$$\partial_t \hat{f} + ikv\hat{f} + \widehat{E\partial_v f} = 0$$

Vlasov is of the form:

$$\dot{u} = iau + F(u)$$

Variation of constant: $\partial_t(e^{-iat}u) = e^{-iat}F(u)$. No more CFL in x of the form $\Delta t \leq \sigma \frac{\Delta x}{v_{\max}}$ with $[-v_{\max}, v_{\max}] \equiv \mathbb{R}$.

Time integration:

$$u(t_n + \Delta t) = \exp(ia\Delta t)u(t_n) + \int_0^{\Delta t} \exp(ia(\Delta t - s))F(u(t_n + s)) ds$$

with $\Delta t > 0$, $t_n = n\Delta t$ with $n \in \mathbb{N}$

Linear part is exact! ✓

Idea of exponential integrators

2 classes of methods:

exponential Runge-Kutta: solve exactly what we can, and interpolate the rest. For example first order exponential Euler method:

$$u(t_n + \Delta t) \approx u^{n+1} = e^{-ia\Delta t} u^n + \Delta t \varphi_1(ia\Delta t) F(u^n)$$

where $\varphi_1(z) = \frac{1}{z} \int_0^z e^{z-\tau} d\tau = \frac{e^z - 1}{z}$



Hochbruck and Ostermann 2010, *Acta Numerica*

Lawson: Change of variable: $\tilde{u}(t) = e^{-iat} u(t)$, we solve with a RK method: $\dot{\tilde{u}} = \tilde{F}(t, \tilde{u}) = e^{-iat} F(e^{iat} \tilde{u}(t))$

For example, Lawson Euler method:

$$\tilde{u}(t_n + \Delta t) \approx \tilde{u}^{n+1} = \tilde{u}^n + \Delta t e^{-iat_n} F(e^{iat_n} \tilde{u}^n)$$

or as an expression of u :

$$u^{n+1} = e^{-ia\Delta t} u^n + \Delta t e^{ia\Delta t} F(u^n)$$



Isherwood, Grant, and Gottlieb 2018, *Journal on Numerical Analysis*

- 1 Motivation for Vlasov-Poisson equations
- 2 Linear analysis
 - Lawson methods
 - Exponential Runge-Kutta methods
- 3 Numerical simulation: Vlasov-Poisson equations
- 4 Conclusion

Reminder of stability tools

If we want to study stability of:

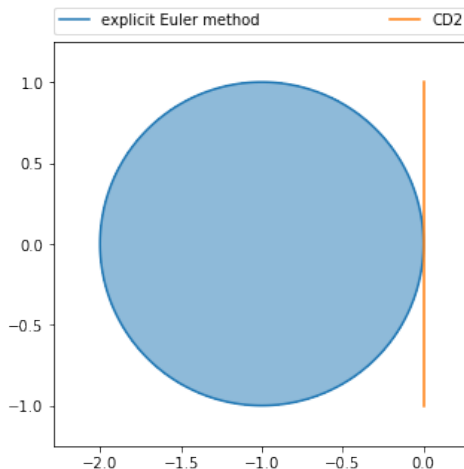
$$\partial_t u + \partial_x u = 0$$

with centered scheme (CD2) $(\partial_x u)_j \approx \frac{1}{2\Delta x}(u_{j+1} - u_{j-1})$. After a Fourier transform (*von Neumann analysis*):

$$\dot{u} + i \frac{\sin(k\Delta x)}{\Delta x} u = 0$$

Explicit Euler method in time: we have to stretch **eigenvalues** (or **Fourier symbol**) of CD2 into explicit Euler **stability domain**.

Reminder of stability tools



Stability function $p(z)$: $u^{n+1} = p(\Delta t \lambda) u^n$ to solve $\dot{u} = \lambda u$

Stability domain: $\mathcal{D} = \{z \in \mathbb{C}, |p(z)| \leq 1\}$

From linear Vlasov equation to toy model

Linear Vlasov equation:

$$\partial_t f + a \partial_x f + b \partial_v f = 0$$

Fourier transform in x , CD2 in v plus a Fourier transform in v , formally:

$$\frac{df}{dt} + iakf + b \frac{i \sin(\varphi)}{\Delta x} f = 0$$

Toy model:

$$\dot{u} + iau + \lambda u = 0$$

with $a \in \mathbb{R}$, $\lambda \in \mathbb{C}$ (diffusive scheme for example).

λ is the Fourier symbol (or eigenvalues) of FD method to approximate $\partial_v f$.

Phase space discretization

In v direction we use a FD method:

CD2 (centered difference of order 2): $(\partial_v f)(v_j) \approx \frac{f_{j+1} - f_{j-1}}{2\Delta v}$

WENO5 (weighted essentially non-oscillatory of order 5):

WENO5: non linear scheme: ~~Von Neumann analysis~~

LW5 (linearized WENO5): linear scheme (this is Lagrange interpolation of order 5)

$$(\partial_v f)(v_j) \approx \frac{1}{\Delta v} \left(-\frac{1}{30}f_{j-3} + \frac{1}{4}f_{j-2} - f_{j-1} + \frac{1}{3}f_j + \frac{1}{2}f_{j+1} - \frac{1}{20}f_{j+2} \right)$$

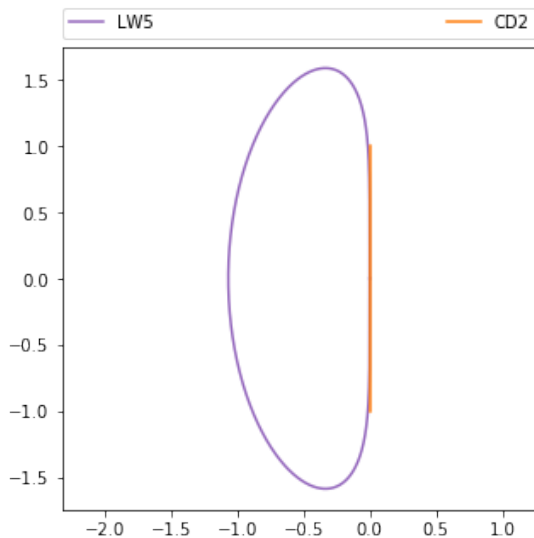


Wang and Spiteri 2007, *Journal on Numerical Analysis*



Motamed, Macdonald, and Ruuth 2010, *Journal of Scientific Computing*

Fourier symbols



Lawson methods stability domain

For our toy model:

$$\dot{u} = iau + \lambda u$$

Change of variable: $\tilde{u}(t) = e^{-iat} u(t)$

$$\dot{\tilde{u}} = e^{-iat} \lambda e^{iat} \tilde{u}$$

Apply a Runge-Kutta method to compute stability function of Lawson method:

$$\tilde{u}^{n+1} = \underbrace{p(\lambda \Delta t)}_{\text{stability function of RK}} \tilde{u}^n$$

i.e.:

$$u^{n+1} = \underbrace{p(\lambda \Delta t) e^{-ia \Delta t}}_{\text{stability function of Lawson}} u^n$$

Stability domain: $\mathcal{D} = \{z \in \mathbb{C}, |p(z)| \leq 1\}$ of Lawson method is **the same** as the underlying Runge-Kutta method **because** $ia \in i\mathbb{R}$

Considered $Lawson(RK(s, p))$ methods

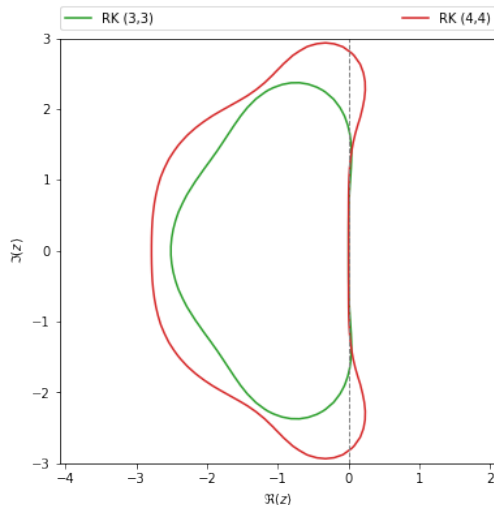
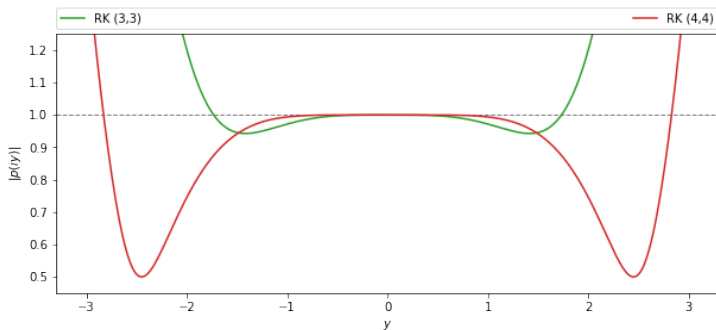


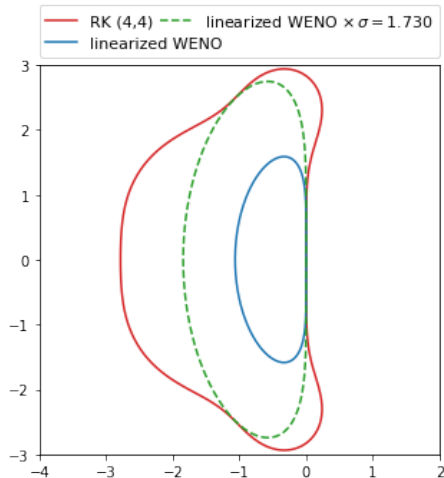
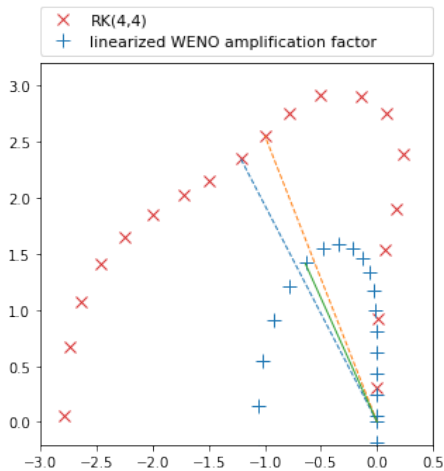
Figure: $\{z \in \mathbb{C}, |p(z)| = 1\}$

For stability between a Lawson method and CD2, we solve:

$$|p(iy)| = 1, \quad y \in \mathbb{R}$$



Lawson methods – LW5



Lawson methods – CD2/LW5: CFL estimates

-	Lawson($RK(3, 3)$)	Lawson($RK(4, 4)$)
CD2 (y_{\max})	$\sqrt{3}$	$2\sqrt{2}$
LW5 (σ)	1.433	1.73

Table: CFL number for some Lawson schemes.

Other RK–CD2 CFL estimates:

 Baldauf 2008, *Journal of Computational Physics*

Same results for RK–WENO CFL estimates:

 Motamed, Macdonald, and Ruuth 2010, *Journal of Scientific Computing*

 Lunet et al. 2017, *Monthly Weather Review*

More results of CFL estimates:

<http://jmassot.perso.math.cnrs.fr/ponio.html>

Exponential Runge-Kutta methods

$$\dot{u} = iau + F(u)$$

Example on ExpRK(2,2):

$$\begin{aligned}u^{(1)} &= e^{-ia\Delta t}u^n - \Delta t\varphi_1F(u^n) \\ u^{n+1} &= e^{-ia\Delta t}u^n - \Delta t\left[(\varphi_1 - \varphi_2)F(u^n) + \varphi_2F(u^{(1)})\right]\end{aligned}$$

Stability function becomes:

$$p_{\text{ExpRK}(2,2)}(z) = \frac{1}{2}\varphi_1\varphi_{1,2}z^2 + \left(\varphi_1 + i\frac{\varphi_1\varphi_{1,2}}{2}a\right)z + 1 + i\varphi_1a$$

Stability domain depends of $a\Delta t \dots$ ✗

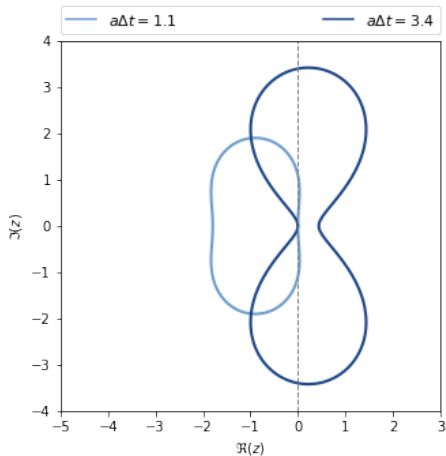


Figure: Stability domain of ExpRK(2,2) for $a\Delta t \in \{1.1, 3.4\}$

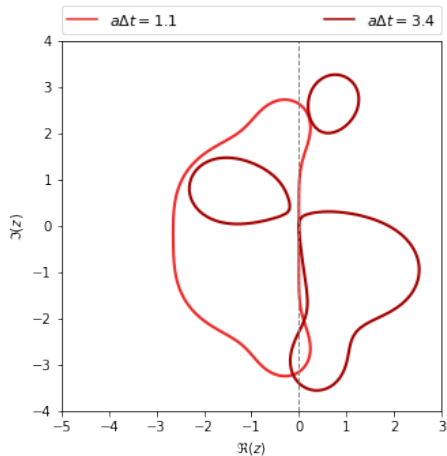


Figure: Stability domain of Cox-Matthews for $a\Delta t \in \{1.1, 3.4\}$

- 1 Motivation for Vlasov-Poisson equations
- 2 Linear analysis
 - Lawson methods
 - Exponential Runge-Kutta methods
- 3 Numerical simulation: Vlasov-Poisson equations
- 4 Conclusion

$$\begin{cases} \partial_t f + v \partial_x f + E \partial_v f = 0 \\ \partial_x E = \int_{\mathbb{R}} f \, dv - 1 \end{cases}$$

Numerical tools:

FFT in x direction

CD2 or WENO5 in v direction

Lawson(*RK*(s, p)) method in time t

CFL: $\Delta t_n \leq \frac{C \Delta v}{\|E^n\|_{\infty}} \leq \frac{C \Delta v}{\max_n \|E^n\|_{\infty}}$ where $C = y_{\max}$ or σ from the linear theory.

We can choose: $\Delta t = \min \left(0.1, \frac{C \Delta v}{\max_n \|E^n\|_{\infty}} \right)$

$$f(t=0, x, v) = f_0(x, v) = \frac{1}{\sqrt{2\pi}} e^{-\frac{v^2}{2}} (1 + 0.001 \cos(0.5x))$$

$$x \in [0, 4\pi], v \in [-8, 8], N_x = 81, N_v = 128$$

Because of damping:

$$\max_n \|E^n\|_\infty = \|E^0\|_\infty$$

So, we choose $\Delta t = 0.1$ (with $\Delta t = 100$ it is still stable!)

Landau damping: numerical results

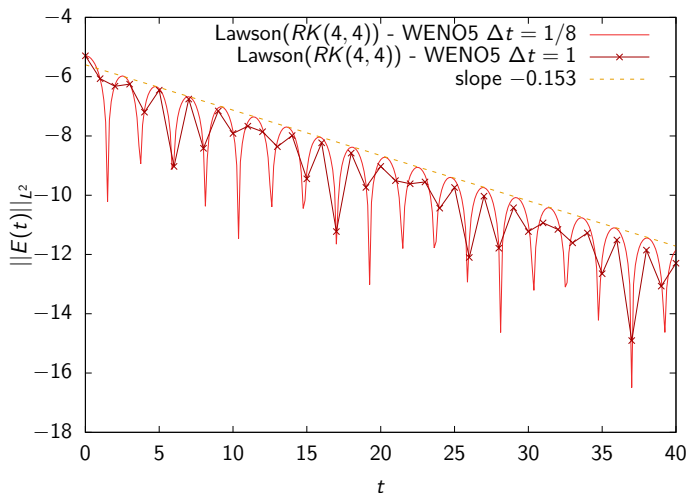


Figure: Landau damping test: time history of $\|E(t)\|_{L^2}$ (semi-log scale) obtained with Lawson($RK(4,4)$) and WENO5 with $\Delta t = 1/8$ and $\Delta t = 1$.

Landau damping: numerical results

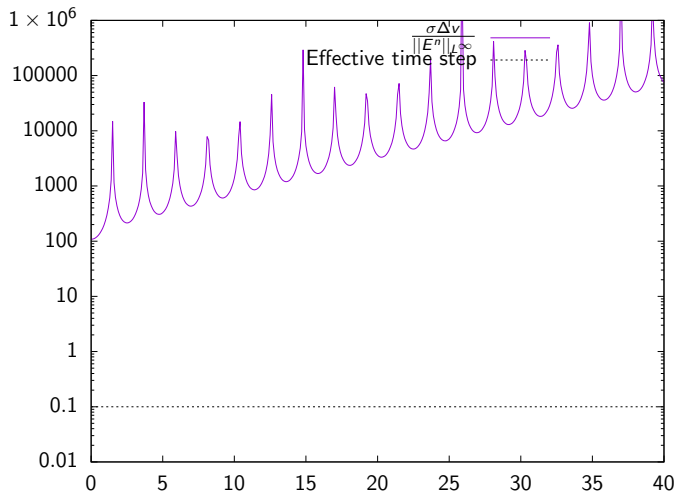


Figure: Landau damping test: time history of the CFL condition (semi-log scale).

Bump on Tail (BoT)

$$f(t=0, x, v) = \left[\frac{0.9}{\sqrt{2\pi}} e^{-\frac{v^2}{2}} + \frac{0.2}{\sqrt{2\pi}} e^{-2(v-4.5)^2} \right] (1 + 0.001 \cos(0.5x))$$

$x \in [0, 20\pi]$, $v \in [-8, 8]$, $N_x = 135$, $N_v = 256$

Numerical estimation of $\max_n \|E^n\|_\infty \approx 0.6$, we choose $\Delta t = \frac{C\Delta v}{0.6}$

BoT: numerical results

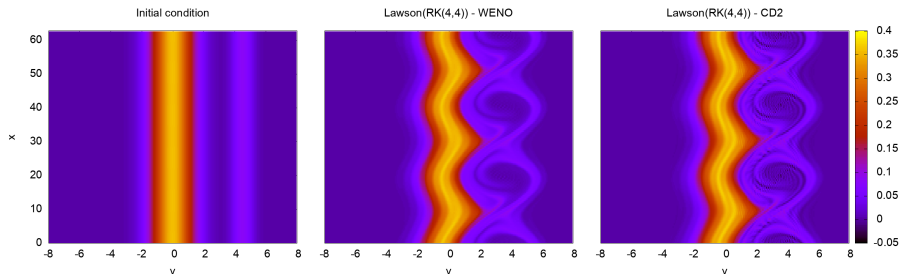


Figure: Distribution function at time $t = 0$ as a function of x and v (left), at time $t = 40$ for Lawson($RK(4,4)$) + WENO5 (center) and Lawson($RK(4,4)$) + centered scheme (right).

BoT: numerical results

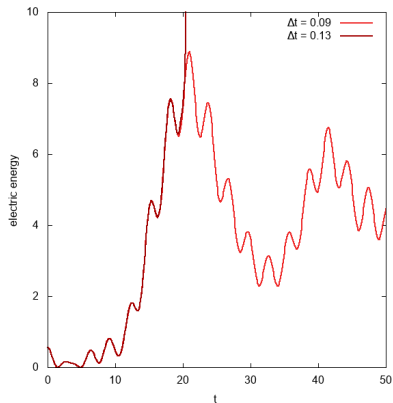


Figure: Illustration of the accuracy of the CFL estimate obtained from the linear theory. History of electric energy with Lawson($RK(4,4)$) + WENO5

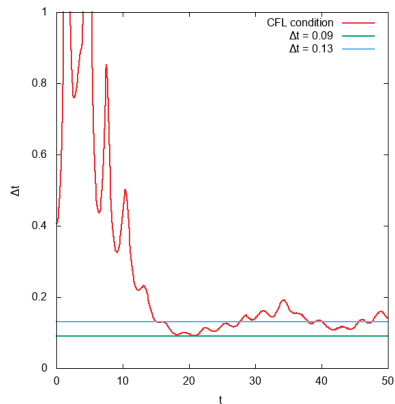


Figure: History of CFL condition for Lawson($RK(4,4)$) + WENO5 case

- 1 Motivation for Vlasov-Poisson equations
- 2 Linear analysis
 - Lawson methods
 - Exponential Runge-Kutta methods
- 3 Numerical simulation: Vlasov-Poisson equations
- 4 Conclusion

Summary

Better understanding on stability of Lawson or ExpRK methods in transport equations

Python script with `sympy` to compute estimates of CFL of Lawson – CD2, Lawson – WENO (5 or 3)

Future works

We can improve method with an embedded Runge-Kutta method (Dormand-Prince method, used in `ode45` of Matlab)

Compare performance between exponential integrators and splitting methods (same order)

Use semi-Lagrangian method to remove dependency on periodic space (Fourier transform)

Thank you for your attention

