# Semilagrangian methods for the BGK model

**Giovanni Russo**

Dipartimento di Matematica e Informatica
Università di Catania

Kinetic Theory and Fluid Dynamics: From micro to macroscopic modeling

In honor of Yoshio Sone's 80th and Kazuo Aoki's 65th birthday

Rakuyu Kaikan, Kyoto, Japan, May 26-28, 2016

Main collaborators:
Maria Groppi, Giuseppe Stracquadanio, Univ. of Parma
Seok-Bae Yun, Sungkyunkwan University

# Outline

# Kazuo Aoki in Catania, March 2004

# Kazuo in Syracuse

# Kazuo with Giuseppe, Kyoto, March 2006

# Kazuo with Giuseppe, Catania, December 2015

# BGK model

The BGK model (Bhatnagar-Gross-Krook '54) approximates Boltzmann equation for the evolution of a rarefied gas.

The main variable is the distribution function $f$ of the particles, as in the Boltzmann equation. The evolution of $f$ is given by:

$$\frac{\partial f}{\partial t} + v \cdot \nabla_x f = \frac{1}{\epsilon}(M[f] - f) \tag{1}$$

with initial condition $f(x, v, 0) = f_0(x, v)$.
Here $\epsilon$ represents the non dimensional collision time. Hydrodynamic regime $\rightarrow \epsilon \ll 1$
Rarefied regime $\rightarrow \epsilon \sim O(1)$

# BGK model

The BGK model (Bhatnagar-Gross-Krook '54) approximates Boltzmann equation for the evolution of a rarefied gas.

The main variable is the distribution function $f$ of the particles, as in the Boltzmann equation. The evolution of $f$ is given by:

$$\frac{\partial f}{\partial t} + v \cdot \nabla_x f = \frac{1}{\epsilon}(M[f] - f) \tag{1}$$

with initial condition $f(x, v, 0) = f_0(x, v)$.
Here $\epsilon$ represents the non dimensional collision time. Hydrodynamic regime $\rightarrow \epsilon \ll 1$
Rarefied regime $\rightarrow \epsilon \sim O(1)$

Here we present results on numerical methods for the BGK equation based on a Semi-Lagrangian formulation using BDF
(Stracquadanio, Russo, Groppi, Comm. Math. Science, accepted).
The method is compared with a RK-based approach (Russo, Santagati, 2008).

# Semi-Lagrangian formulation

Simplified model: 1D in space and velocity:

$$\frac{\partial f}{\partial t} + v\frac{\partial f}{\partial x} = \frac{1}{\epsilon}(M[f] - f). \tag{2}$$

with

$$M[f] = \frac{\rho}{(2\pi RT)^{1/2}} exp\left( - \frac{(v-u)^2}{2RT} \right)$$

$t \geq 0, \quad x, v \in \mathbb{R}.$

Semi-Lagrangian: follow the evolution along the characteristics.

$$\begin{aligned} &\frac{df(x,v,t)}{dt} = \frac{1}{\epsilon}\Big( M[f](x,v,t) - f(x,v,t) \Big), \\ &\frac{dx}{dt} = v, \quad x(0) = \tilde{x}, \quad f(0,t,v) = f_0(\tilde{x}, v) \quad t \geq 0, \quad x, v \in \mathbb{R}. \end{aligned} \tag{3}$$

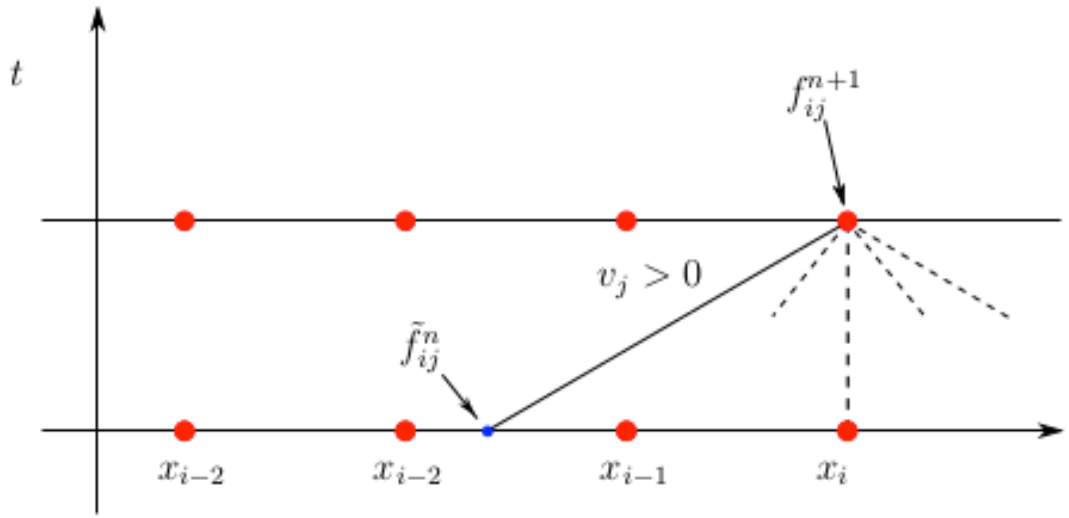
$$\Rightarrow x(t) = \tilde{x} + vt, \quad t \geq 0, \quad x, v \in \mathbb{R}, \qquad \text{(characteristic straight lines).}$$

# Implicit first order Semi-Lagrangian scheme

Let $f_{ij}^n \approx f(x_i, v_j, t^n)$ be approximate solution.
Possible stiffness (small $\epsilon$) $\Rightarrow$ implicit formulation.

$$f_{ij}^{n+1} = \tilde{f}_{ij}^n + \frac{\Delta t}{\epsilon}(M_{ij}^{n+1} - f_{ij}^{n+1}), \tag{4}$$

Here $\tilde{f}_{ij}^n = f(t^n, \tilde{x}_i = x_i - v_j \Delta t, v_j)$ can be calculated by (linear) interpolation from $\{f_{.j}^n\}$.



# Solution of the implicit step

Equation (4) is non linear.
Indeed $M[f]_{i,j}^{n+1}$ depends on $f_{ij}^{n+1}$ through its moments.

# Solution of the implicit step

Equation (4) is non linear.
Indeed $M[f]_{i,j}^{n+1}$ depends on $f_{ij}^{n+1}$ through its moments.

Let $\phi(v)$ be the vector $\phi(v) = (1, v, v^2)^T$. Compute the moments of $f_{ij}^{n+1}$:

$$\langle f_{ij}^{n+1}\phi\rangle = \langle \tilde{f}_{ij}^{n}\phi\rangle + \frac{\Delta t}{\epsilon}\langle (M_{ij}^{n+1} - f_{ij}^{n+1})\phi\rangle.$$

# Solution of the implicit step

Equation (4) is non linear.
Indeed $M[f]_{i,j}^{n+1}$ depends on $f_{ij}^{n+1}$ through its moments.

Let $\phi(v)$ be the vector $\phi(v) = (1, v, v^2)^T$. Compute the moments of $f_{ij}^{n+1}$:

$$\langle f_{ij}^{n+1}\phi\rangle = \langle \tilde{f}_{ij}^{n}\phi\rangle + \frac{\Delta t}{\epsilon}\langle (M_{ij}^{n+1} - f_{ij}^{n+1})\phi\rangle.$$

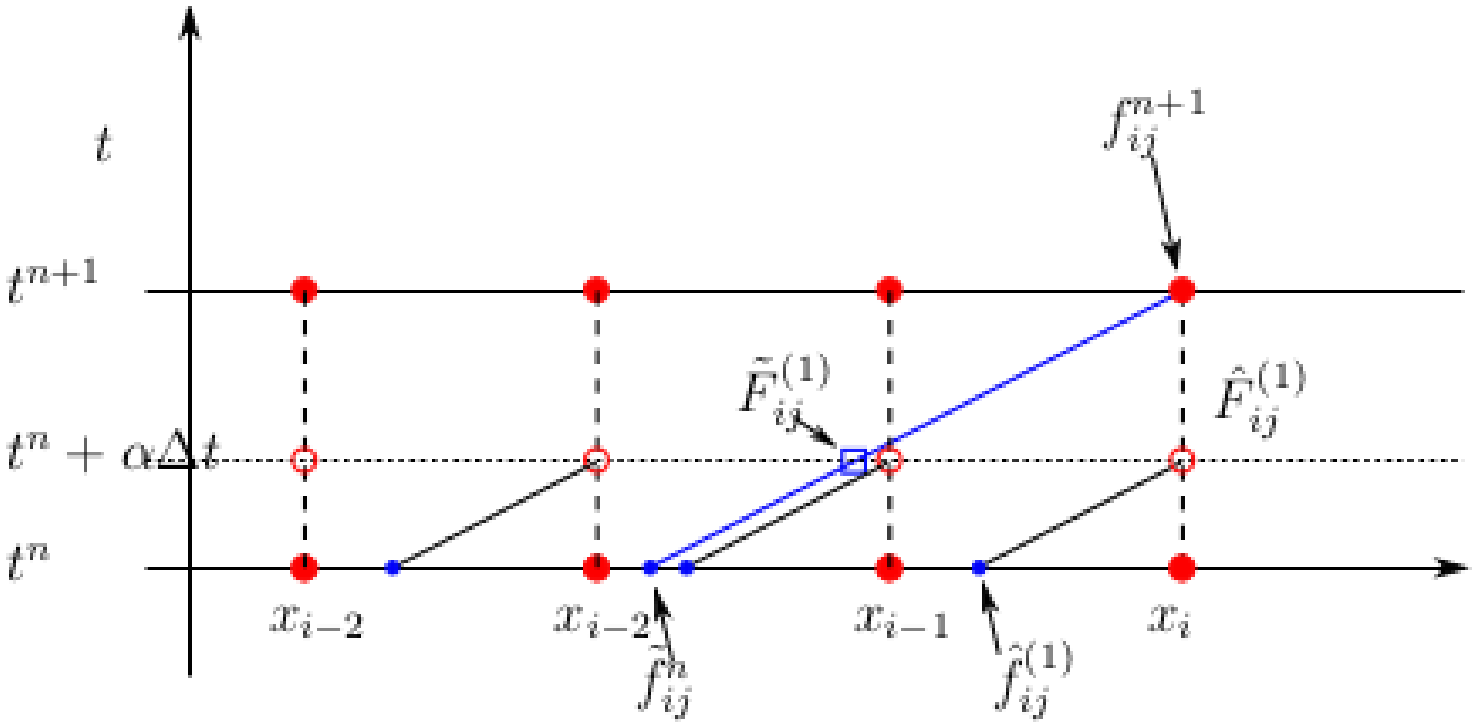
From the conservation, we have

$$\langle (M_{ij}^{n+1} - f_{ij}^{n+1})\phi\rangle = 0 \qquad \Rightarrow \qquad \langle f_{ij}^{n+1}\phi\rangle = \langle \tilde{f}_{ij}^{n}\phi\rangle$$

Hence we immediately find the macroscopic variables $\rho_i^{n+1}$, $u_i^{n+1}$ and $T_i^{n+1}$ corresponding to $f_{ij}^{n+1}$ using $\tilde{f}_{ij}^{n}$ and with these values the approximated Maxwellian is updated.

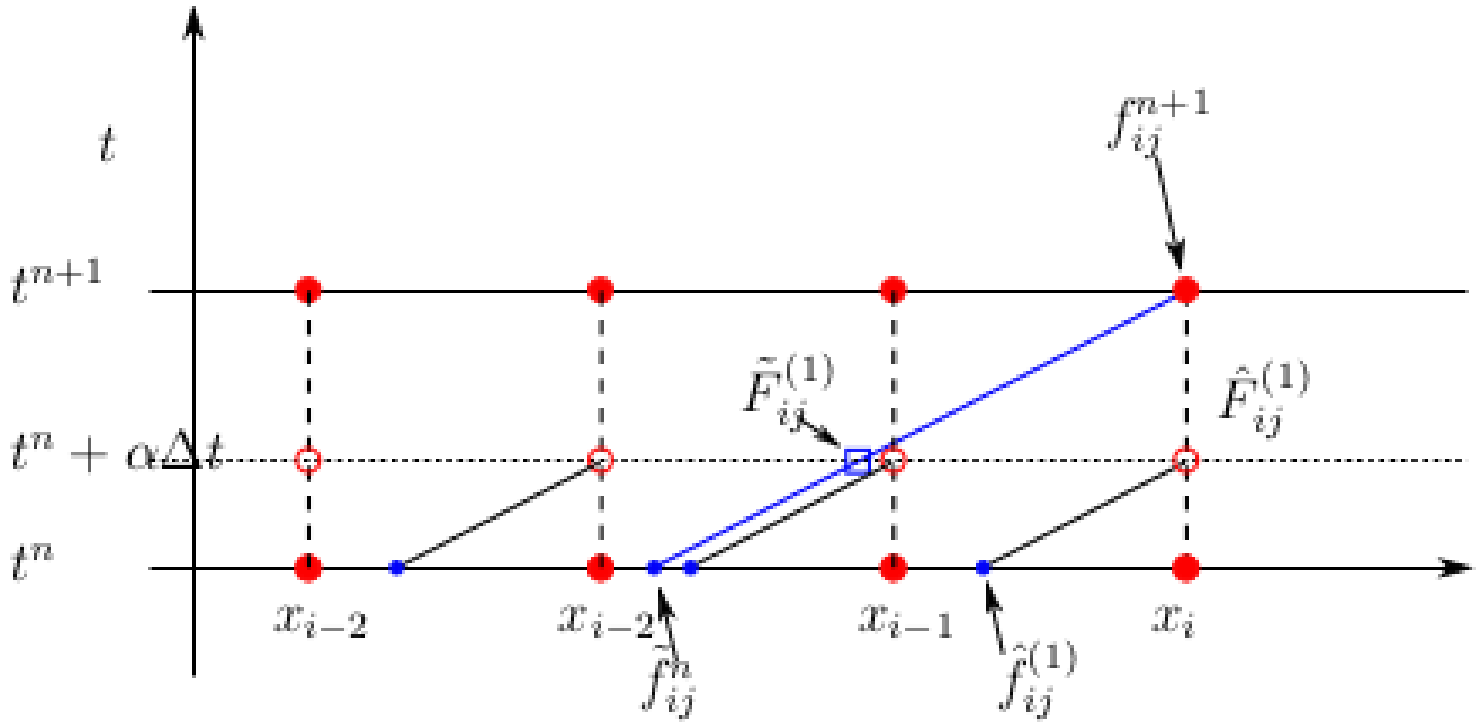
# Higher order: Runge-Kutta

Classical RK schemes can be adopted.
Stage values are computed along the characteristics.
First they are computed at grid position $x_i$ (empty cycles) and then the value of $f$ (or the RK flux) is interpolated on the characteristics (empty squares)



# Higher order: Runge-Kutta

Classical RK schemes can be adopted.
Stage values are computed along the characteristics.
First they are computed at grid position $x_i$ (empty cycles) and then the value of $f$ (or the RK flux) is interpolated on the characteristics (empty squares)




High order in space is obtained by WENO reconstruction.

# High order BDF schemes

Runge-Kutta methods may be expensive.
The BDF (Backward Difference Formula) methods allow same order of accuracy at lower cost.

We will show some numerical results concerning the BDF methods with 2 (BDF2) and 3 (BDF3) steps. Applying these methods to the Lagrangian formulation of the BGK model we obtain the following schemes:

$$f_{ij}^{n+1} = \frac{4}{3}\overset{(1)}{f}{}_{ij}^{n} - \frac{1}{3}\overset{(2)}{f}{}_{ij}^{n-1} + \frac{\Delta t}{\epsilon}(M_{ij}^{n+1} - f_{ij}^{n+1})$$

$$f_{ij}^{n+1} = \frac{11}{18}\overset{(1)}{f}{}_{ij}^{n} - \frac{9}{11}\overset{(2)}{f}{}_{ij}^{n-1} + \frac{2}{11}\overset{(3)}{f}{}_{ij}^{n-2} + \frac{\Delta t}{\epsilon}(M_{ij}^{n+1} - f_{ij}^{n+1})$$

where $\overset{(s)}{f}{}^{n}{}_{ij} = f^n(x_i - sv_j\Delta t, v_j),\ s = 1, 2, 3$, obtained by interpolation.

# High order BDF schemes

Runge-Kutta methods may be expensive.
The BDF (Backward Difference Formula) methods allow same order of accuracy at lower cost.

We will show some numerical results concerning the BDF methods with 2 (BDF2) and 3 (BDF3) steps. Applying these methods to the Lagrangian formulation of the BGK model we obtain the following schemes:

$$f_{ij}^{n+1} = \frac{4}{3}\overset{(1)}{f}{}_{ij}^{n} - \frac{1}{3}\overset{(2)}{f}{}_{ij}^{n-1} + \frac{\Delta t}{\epsilon}(M_{ij}^{n+1} - f_{ij}^{n+1})$$

$$f_{ij}^{n+1} = \frac{11}{18}\overset{(1)}{f}{}_{ij}^{n} - \frac{9}{11}\overset{(2)}{f}{}_{ij}^{n-1} + \frac{2}{11}\overset{(3)}{f}{}_{ij}^{n-2} + \frac{\Delta t}{\epsilon}(M_{ij}^{n+1} - f_{ij}^{n+1})$$

where $\overset{(s)}{f}{}^{n}{}_{ij} = f^n(x_i - sv_j\Delta t, v_j),\ s = 1, 2, 3$, obtained by interpolation.
High order in space is obtained by WENO reconstruction.

# Extension to BGK equation 3D in velocity (Chu reduction)

The technique used in 1D in space, 3D velocity, cylindrical axial symmetry.

Given $f = f(x, (v_x, v_y, v_z), t)$ we introduce

$$g_1 = g_1(x, v_x, t) = \int f \, dv_y dv_z, \quad g_2 = g_2(x, v_x, t) = \int (v_y^2 + v_z^2) f \, dv_y dv_z.$$

Macroscopic moments $\rho$, $\rho u$ and $E$ can be expressed in terms of $g_1$ and $g_2$:

$$\rho = \int g_1 \, dv_x, \qquad \rho u = \int g_1 v_x \, dv_x,$$

$$E = \frac{1}{2} \int v_x^2 g_1 \, dv_x + \frac{1}{2} \int g_2 \, dv_x. \tag{5}$$

# Chu reduction

Integrating BGK eq. in $dv_y dv_z$, multiplying it by $(v_y^2 + v_z^2)$ and integrating in $dv_y dv_z$, we obtain the following system in $g_1$ and $g_2$, 1D in velocity ($v_x = v$):

$$\begin{cases} \dfrac{\partial g_1}{\partial t} + v\dfrac{\partial g_1}{\partial x} = \dfrac{1}{\epsilon}(M_1 - g_1) \\ \dfrac{\partial g_2}{\partial t} + v\dfrac{\partial g_2}{\partial x} = \dfrac{1}{\epsilon}(M_2 - g_2) \end{cases} \tag{6}$$

where $M_2 = M_1 2RT$, with

$$M_1 = M[g_1, g_2] = \frac{\rho}{(2\pi RT)^{1/2}} \exp\left( - \frac{(v-u)^2}{2RT} \right)$$

Once system (6) is solved, macroscopic moments of $f(x, v, t)$ can be obtained through formulas (5).

Implicit step can me explicitly solved in a way similar to the 1D case.

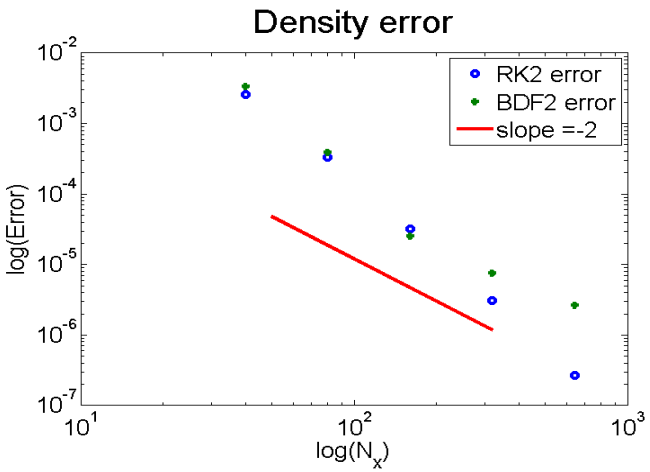
# Numerical test for the problem 1+1D

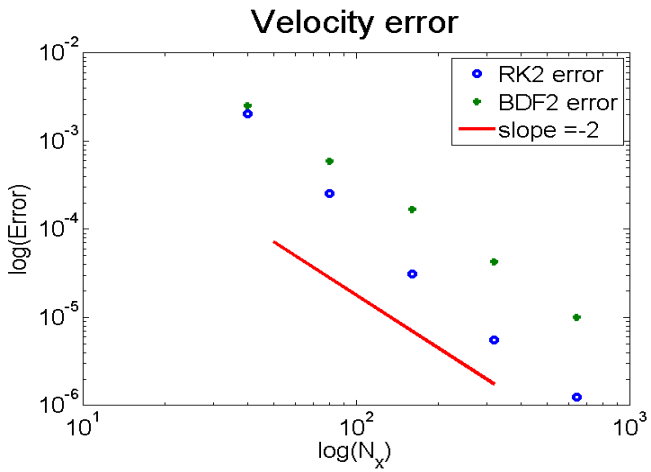
We have considered two numerical test:

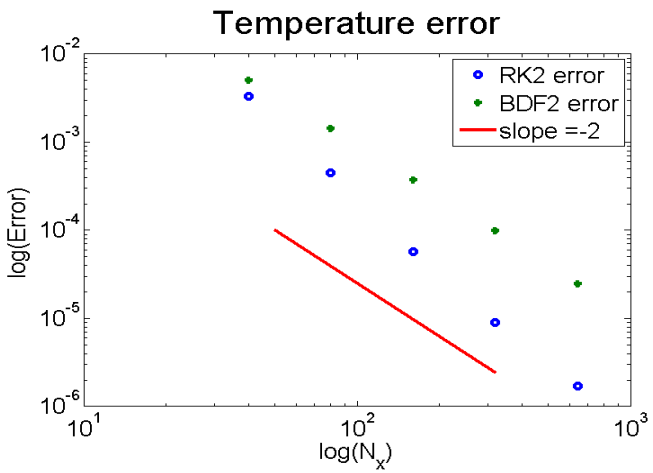
1. Smooth initial data
   $\left(f_0 = M[v, \rho = 1, u = 0.1exp(-(10x-1)^2) - 2exp(-(10x+3)^2), T = 1]\right)$
   - time interval [0,0.04];
   - space interval [-1,1];
   - velocity interval [-10,10];
   - $N_v = 40$;
   - $\Delta t = CFL\,\Delta x/|v_{max}|$;
2. Riemann problem (jump in $x = 0.5$):
   - $(\rho_L, u_L, T_L) = (2.25, 0, 1.125)$, $(\rho_R, u_R, T_R) = (3/7, 0, 1/6)$
   - time interval [0,0.16];
   - space interval [0,1];
   - velocity interval [-10,10];
   - $N_x = 100$;
   - $N_v = 60$;
   - $\Delta t = CFL\,\Delta x/|v_{max}|$;

For each test the cases $\epsilon = 10^{-2}$ and $\epsilon = 10^{-6}$ have been studied.

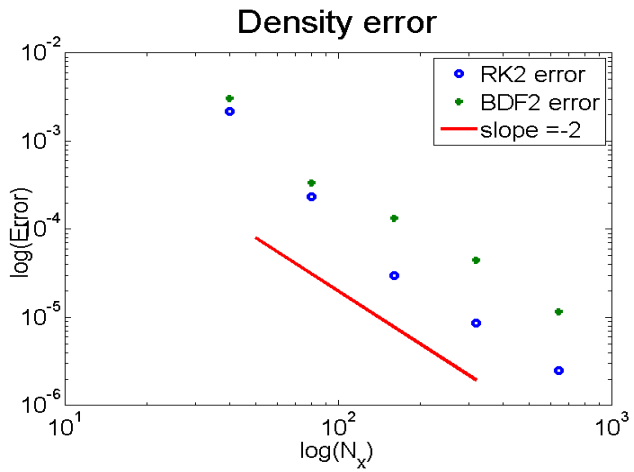
# RK2 and BDF2 accuracy 1+1D in rarefied regime ($\epsilon = 10^{-2}$)

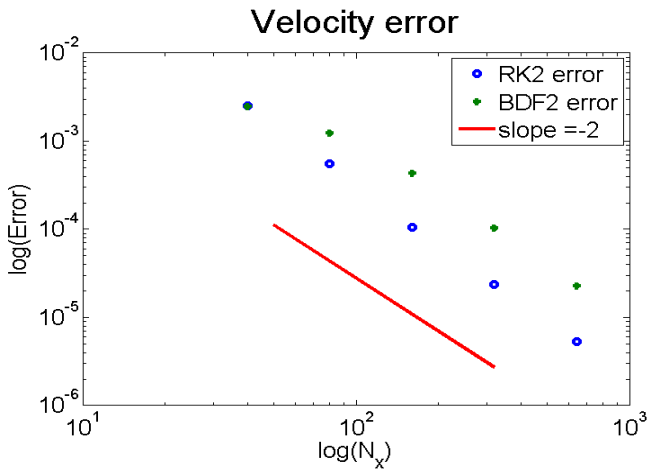


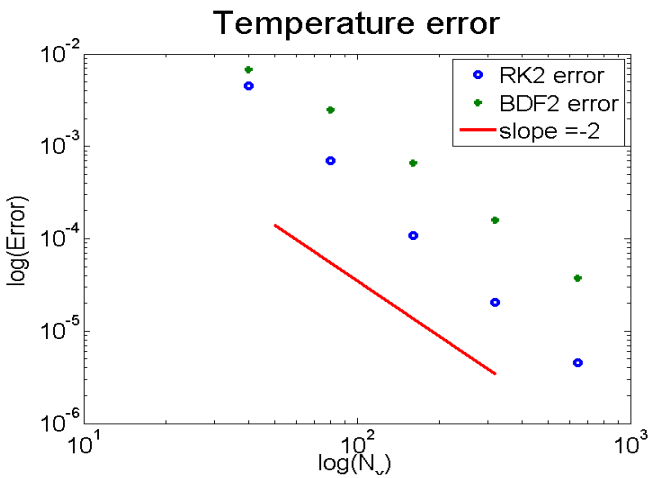



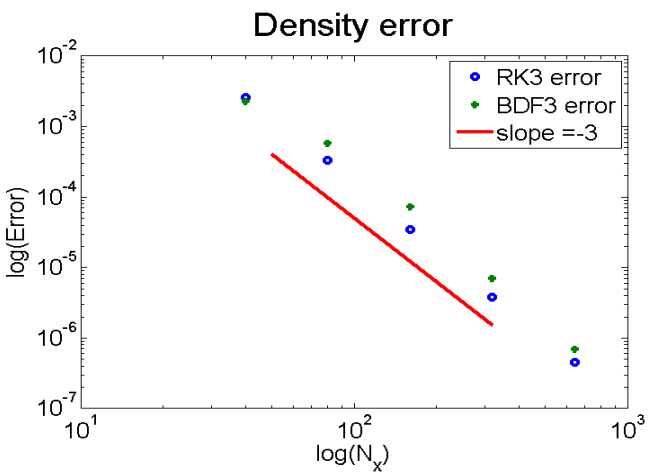
# RK2 and BDF2 accuracy 1+1D in hydrodynamic regime ($\epsilon = 10^{-6}$)

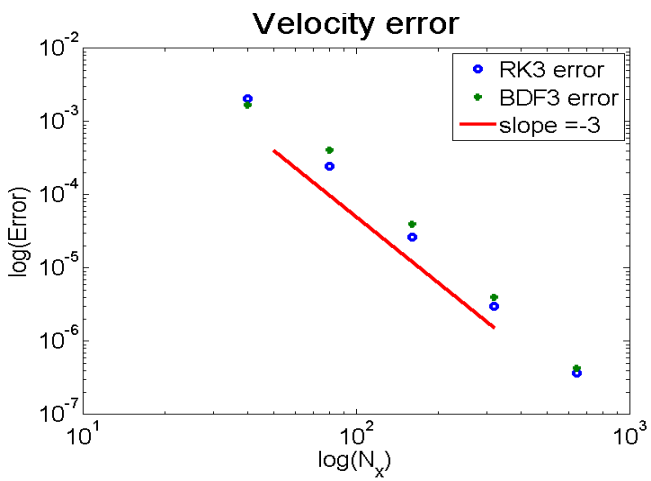


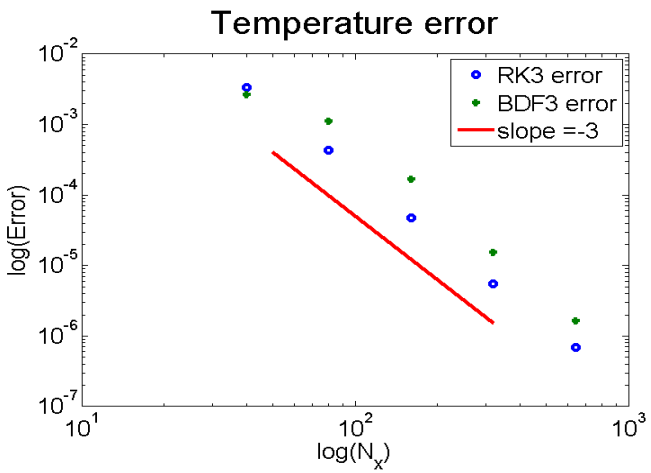



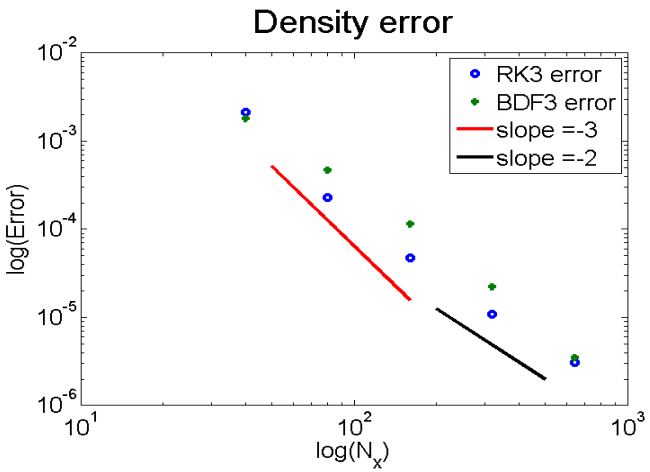
# RK3 and BDF3 accuracy 1+1D in rarefied regime ($\epsilon = 10^{-2}$)

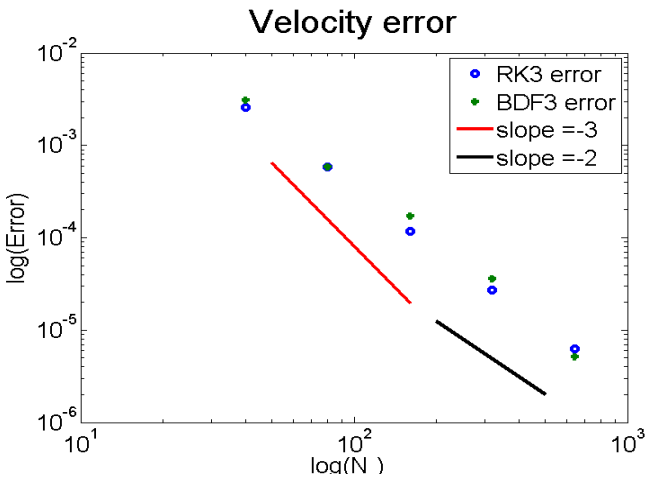


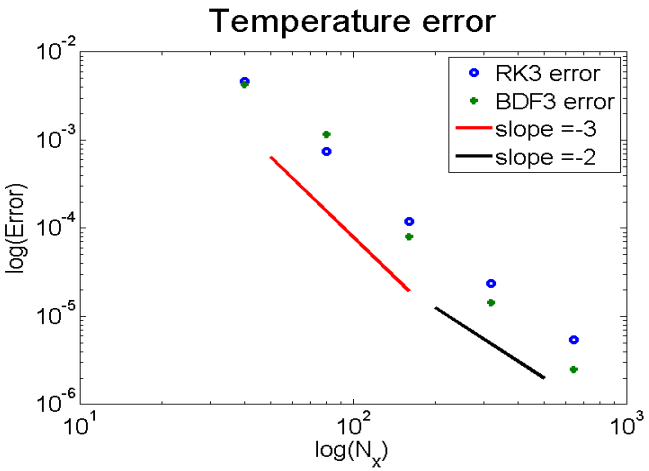



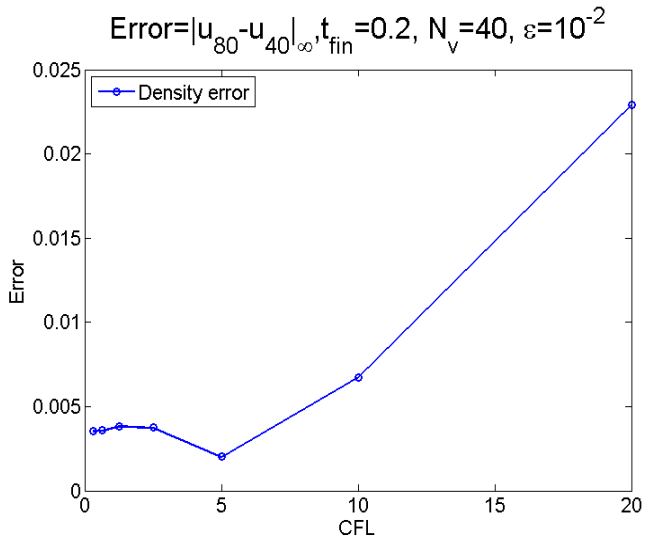
# RK3 and BDF3 accuracy 1+1D in hydrodynamic regime ($\epsilon = 10^{-6}$)

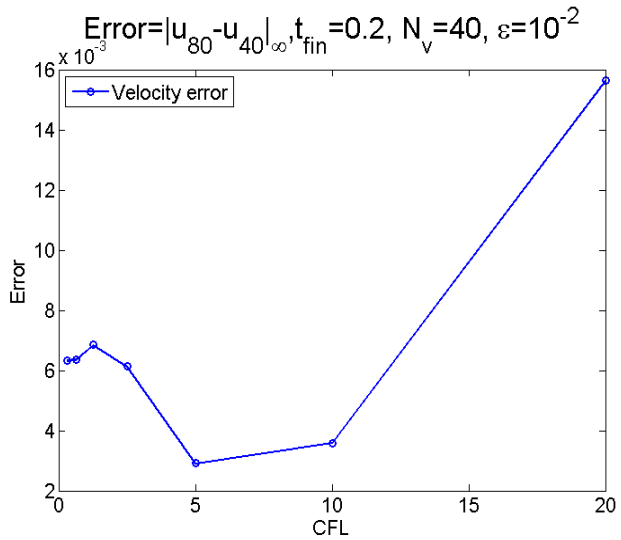


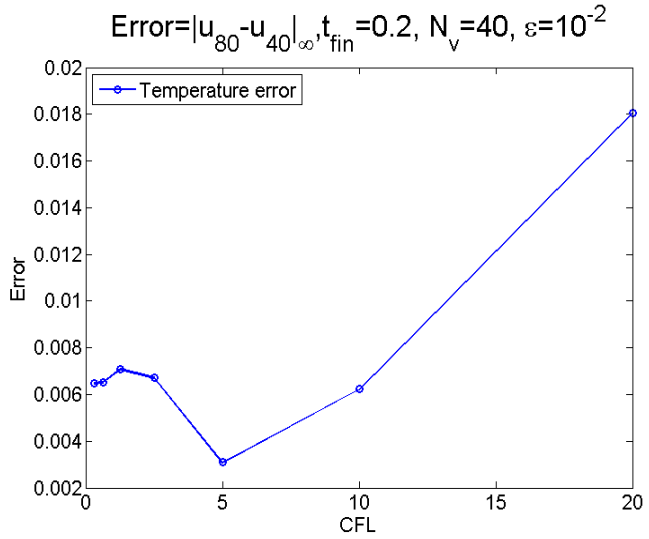



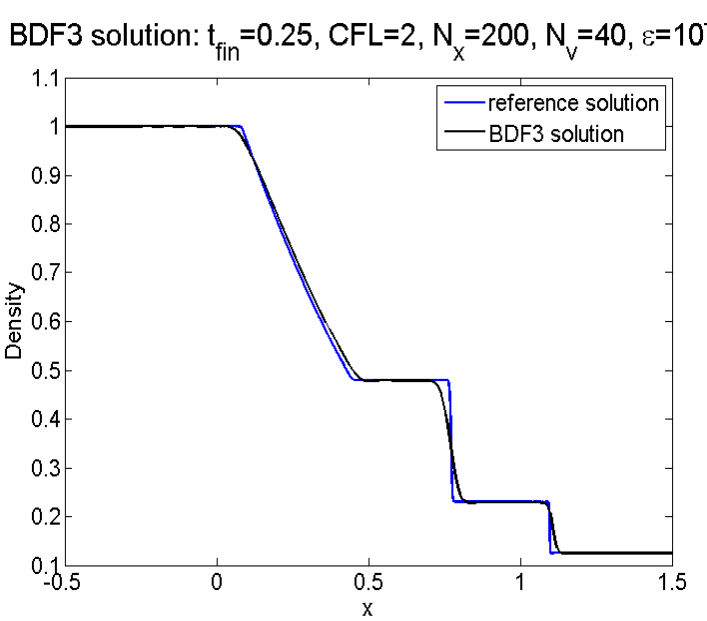
# BDF3-weno3-5, CFL-Error for the problem 1D+1D

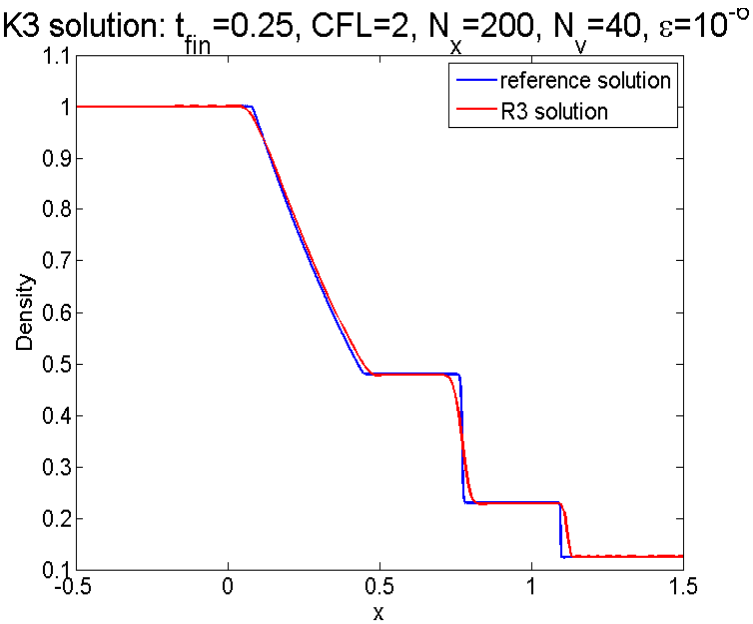


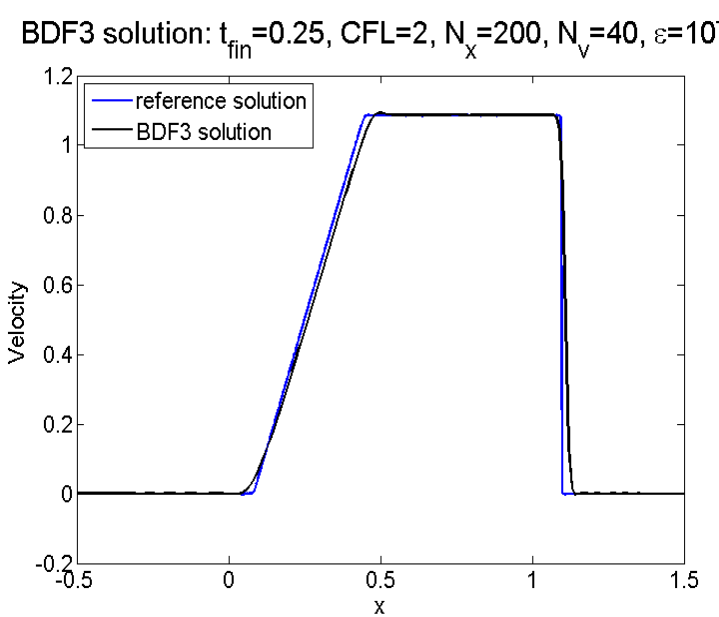



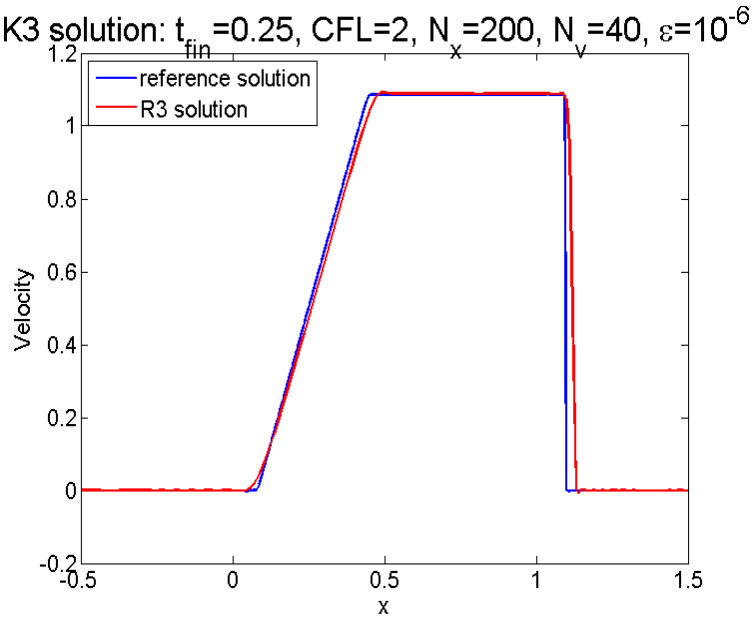
# Comparison with the solution of gas dynamics: density given by BDF3 and RK3 for 1+3D



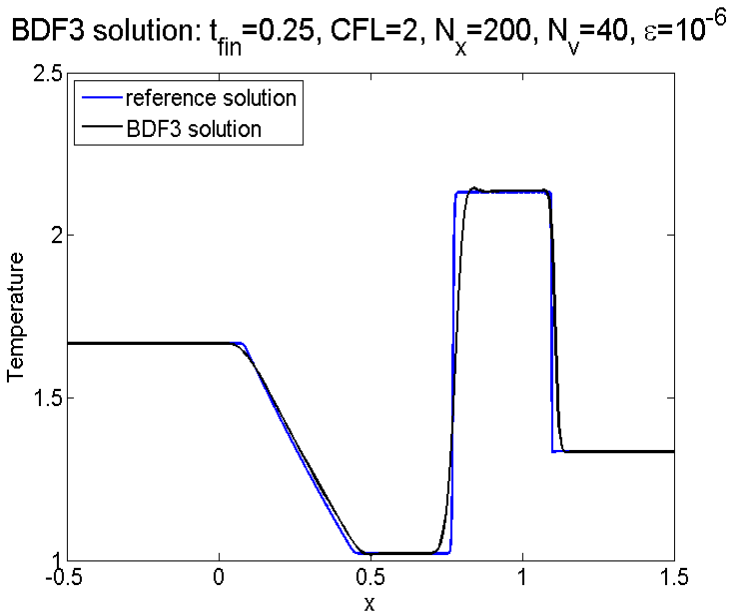



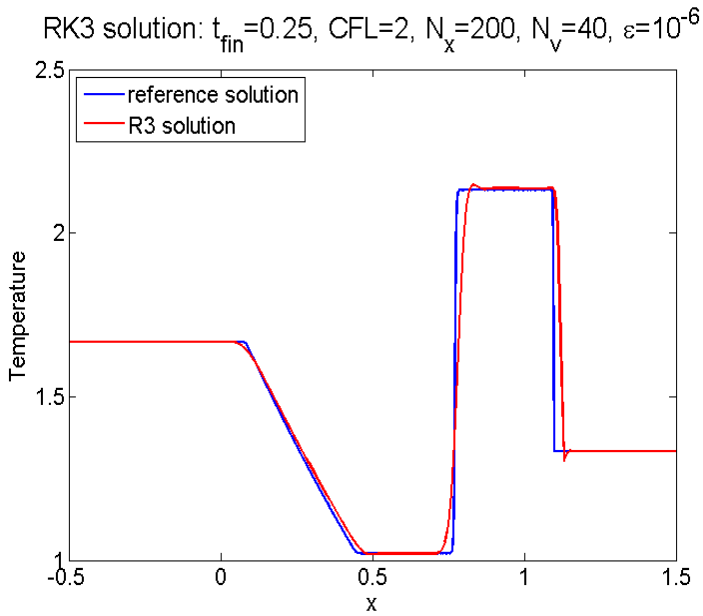
# Comparison with the solution of gas dynamics: velocity given by BDF3 and RK3 for 1+3D



# Comparison with the solution of gas dynamics: temperature given by BDF3 and RK3 for 1+3D



# Reflective BC

The foot of the characteristic is *reflected back* into the domain. Then interpolation can be used.
In one space dimension one can simply add *ghost points* and assign them a symmetric distribution function.

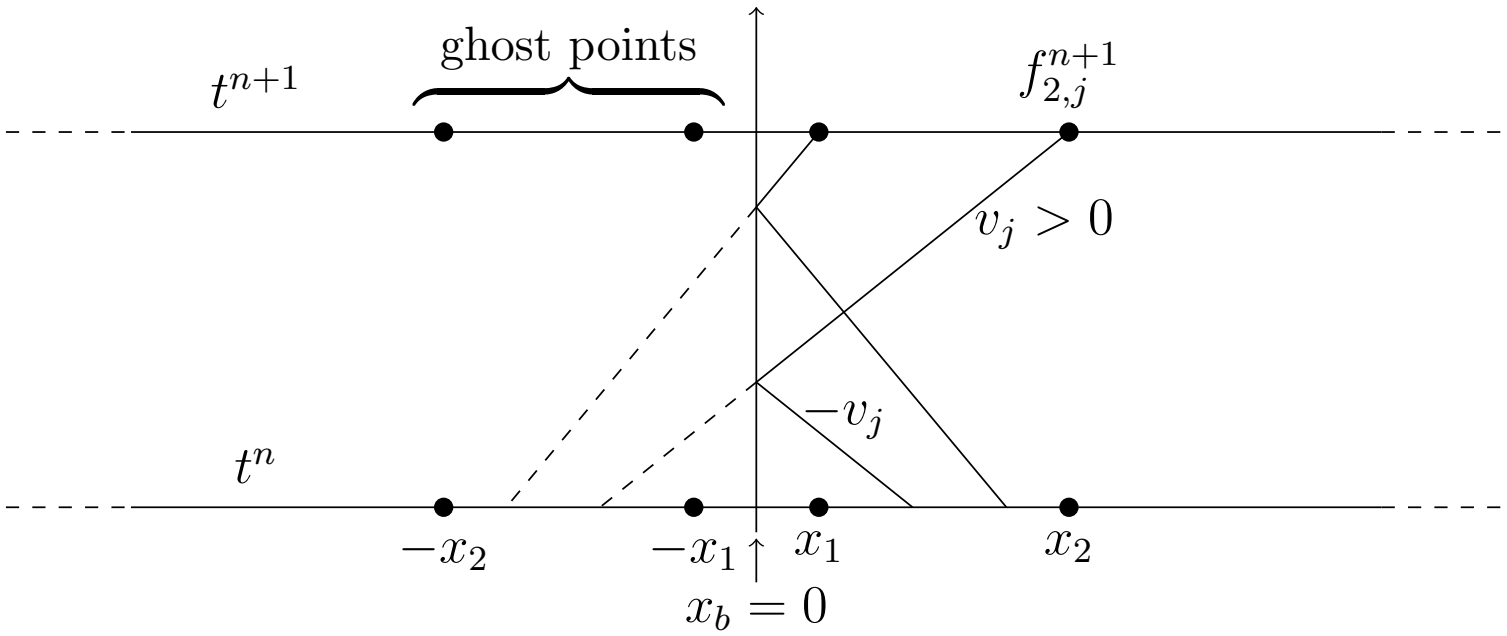
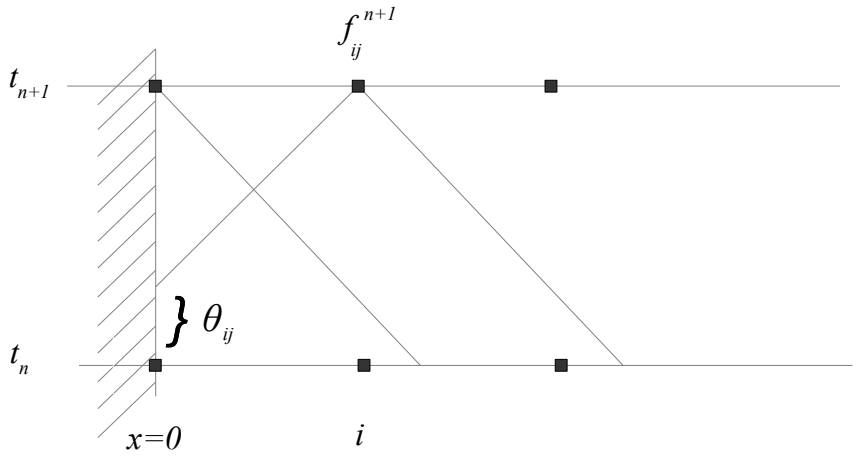



Figure: Discretization of specular reflective conditions.

# Diffusive Boundary Conditions

Let us assume that the boundary is located on the left, at position $x = 0$ with temperature $T_b$.




We assume that we know the solution at time $t_n$:
$\left\{f_{ij}^n, i = 1, ..., N_x,\ j = 1, ..., N_v\right\}$ and the density at $x = 0,\ \rho_b^n$.
Three techniques are adopted:

- Iterative procedure (IP)
- Extrapolation procedure (EP)
- Inverse Lax-Wendroff technique (ILW)

# Diffusive Boundary Conditions: Iterative procedure

The density $\rho_{bj}^{n+1}$ is computed by imposing zero mass flux at $t = t_{n+1}$, $x = 0$, that is $\sum_j v_j f_{bj}^{n+1} = 0$, where

$$f_{bj}^{n+1} = \begin{cases} \rho_b^{n+1} \dfrac{\exp(-v_j^2/2T_b)}{(2\pi T_b)^{1/2}} & \text{for } v_j > 0, \\ f_{\xi_j j}^n + \dfrac{\Delta t}{\varepsilon}(M_{bj}^{n+1} - f_{bj}^{n+1}) & \text{for } v_j < 0 \end{cases}$$

$\xi_j = -v_j \Delta t$. The Maxwellian $M_{bj}^{n+1}$ cannot be computed in the usual way. We shall therefore use an iterative procedure. Let

$$f_{bj}^{(0)} = \begin{cases} \rho_b^{(0)} E_{bj} & \text{for } v_j > 0, \\ f_{\xi_j j}^n & \text{for } v_j < 0 \end{cases}$$

with $E_{bj} = \exp(-v_j^2/2T_b)/(2\pi T_b)^{1/2}$. Skip to tests

# Diffusive Boundary Conditions

Imposing $\sum_j v_j f_{bj}^{(0)} = 0$ one determins $\rho^{(0)}$. Once $\rho^{(0)}$ is known, one computes the moments $m_\beta^{(0)} = \sum_j v_j^\beta f_{bj}^{(0)}, \quad \beta = 0, 1, 2$. From the moments one computes the Maxwellian $M_{bj}^{(0)}$. Then one can iterate untill convergence:

$$f_{bj}^{(k)} = \begin{cases} \rho_b^{(k)} E_{bj} & \text{for } v_j > 0, \\ f_{\xi_j j}^n + \frac{\Delta t}{\varepsilon}(M_{bj}^{(k-1)} - f_{bj}^{(k-1)}) & \text{for } v_j < 0 \end{cases}$$

Finally we set $f_{bj}^{n+1} = \lim_{k\to\infty} f_{bj}^{(k)}$ and $\rho_b^{n+1} = \lim_{k\to\infty} \rho_b^{(k)}$

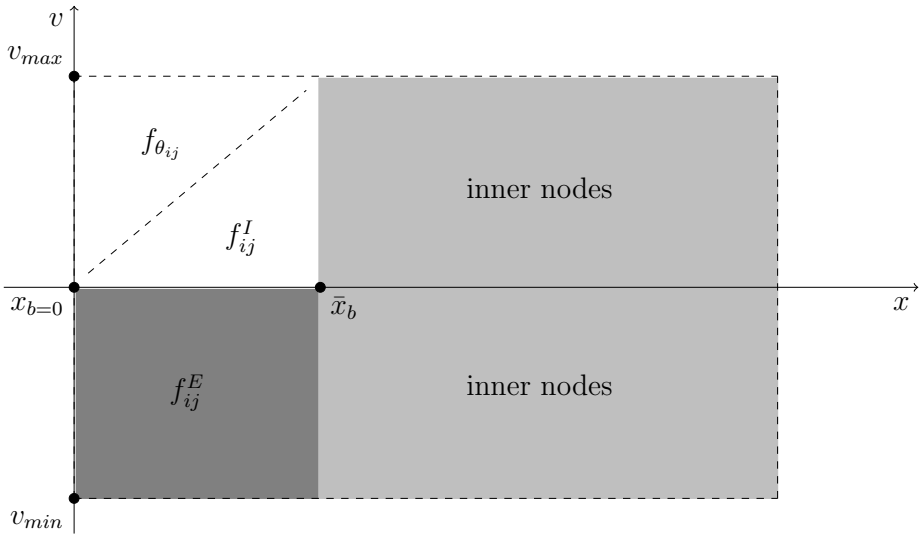
# Diffusive Boundary Conditions

Once the density at the new time has been found, one can then compute the function at other grid points, as follows. Consider, for example, point $x_i$ in the previous figure. Then one has:

$$f_{ij}^{n+1} = \begin{cases} f_{\xi_j j}^{n} + \frac{\Delta t}{\varepsilon}(M_{ij}^{n+1} - f_{ij}^{n+1}) & \text{if } \xi_j = x_i - v_j \Delta t > 0, \\ \\ \left(\theta_{ij}\rho_b^{n+1} + (1-\theta_{ij})\rho_b^{n}\right) E_{bj} & \text{if } \xi_j < 0. \end{cases}$$

The geometrical factor $\theta_{ij} = 1 - x_i/(v_j \Delta t)$ can be computed for each velocity. The Maxwellian $M_{ij}^{n+1}$ may be computed by using an iterative procedure similar to the one used for the computation of $\rho_b^{n+1}$.

# Extrapolation procedure (EP)




The discretized distribution function is defined in $[x_b, +\infty] \times [v_{min}, v_{max}]$.
The distances $|x_b - \bar{x}_b|$ is $v_{max}\Delta t$.

Inner region (light gray): characteristics fan do not touch the boundary.
Outflow region (dark gray): extrapolate the distribution function from the inner region.
Inflow region: (white region, $v_j \cdot n(x) \geq 0$) we use information coming from the wall.
The new distribution function at time $t^{n+1}$ is therefore

$$f_{bj}^{n+1} = \begin{cases} f_{bj}^{E,n+1} & \text{if } v_j \cdot n(x) < 0 \\ \\ \rho_b^{n+1} E_{bj} & \text{if } v_j \cdot n(x) \geq 0. \end{cases} \tag{7}$$

# Inverse Lax-Wendroff procedure

Normally used to extend the distribution function at the ghost points for the inflow boundary. At the boundary $x_b = 0$, a first order Taylor expansion gives

$$f_j(x) = f_{bj} + (x - x_b)\frac{\partial f}{\partial x}|_{x=x_b} + O(\Delta x^2)$$

Hence, a first order approximation of f at ghost points $x_g$ is

$$f_{gj} = f_{bj} + (x_g - x_b)\frac{\partial f}{\partial x}|_{x=x_b}.$$

By reformulating the BGK equation, we have

$$\frac{\partial f}{\partial x}|_{x=x_b} = \frac{1}{v}\left( - \frac{\partial f}{\partial t} + \frac{1}{\varepsilon}(M[f] - f)\right)|_{x=x_b}$$

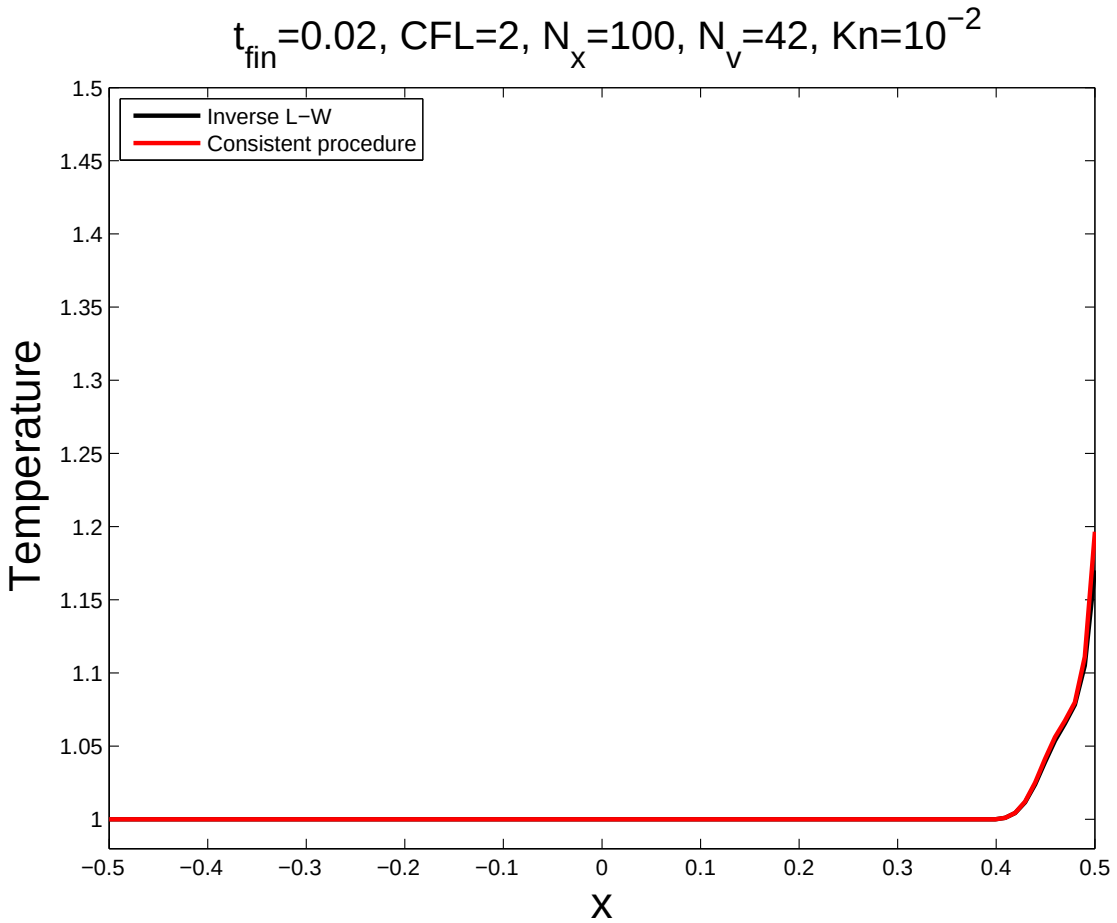
Now instead of approximating the first spatial derivative, one computes the time derivative and the collision operator in $x = x_b$.

# Diffusive boundary condition test

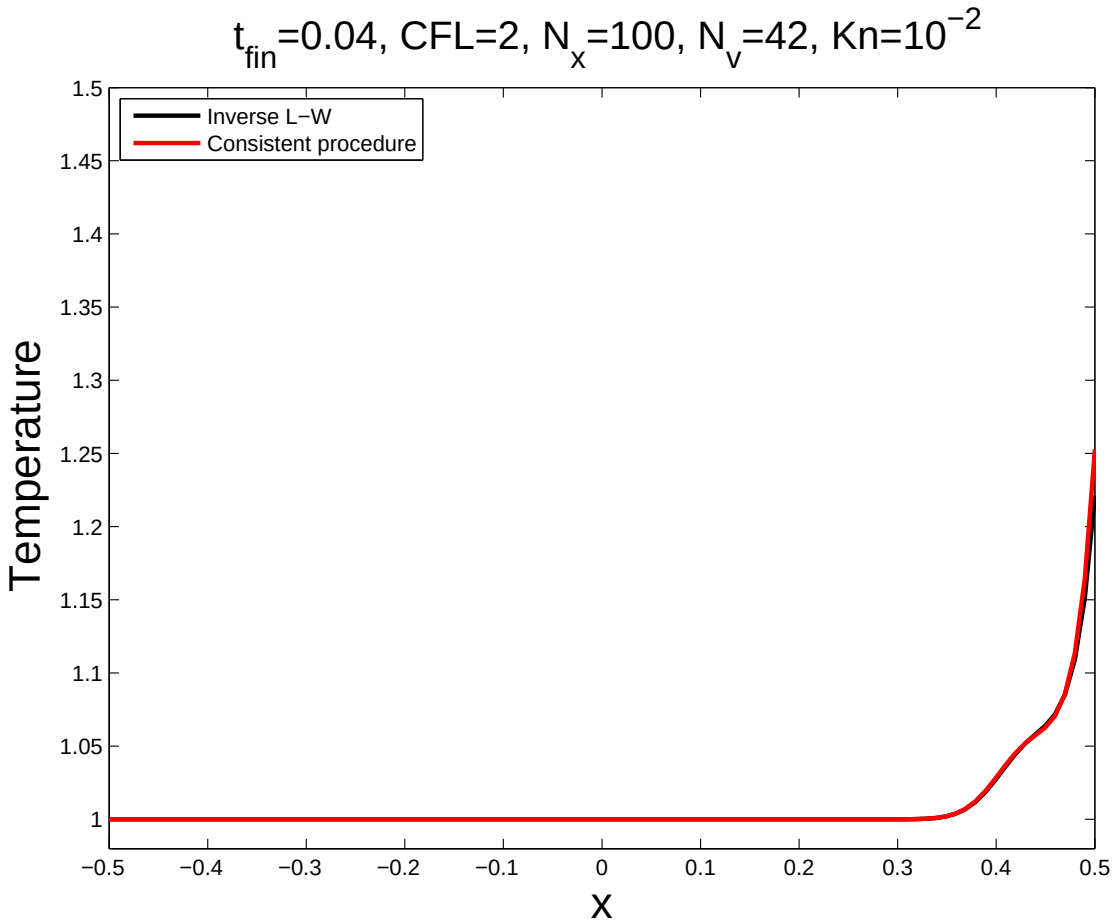
As initial data we have chosen the following:

- $x \in [-0.5, 0.5]$;
- $v \in [-10, 10]$;
- $\rho_0(x) = 1$;
- $u_0(x) = 0$;
- $T_0(x) = 1$, $T_L = 1$, $T_R = 1.44$;

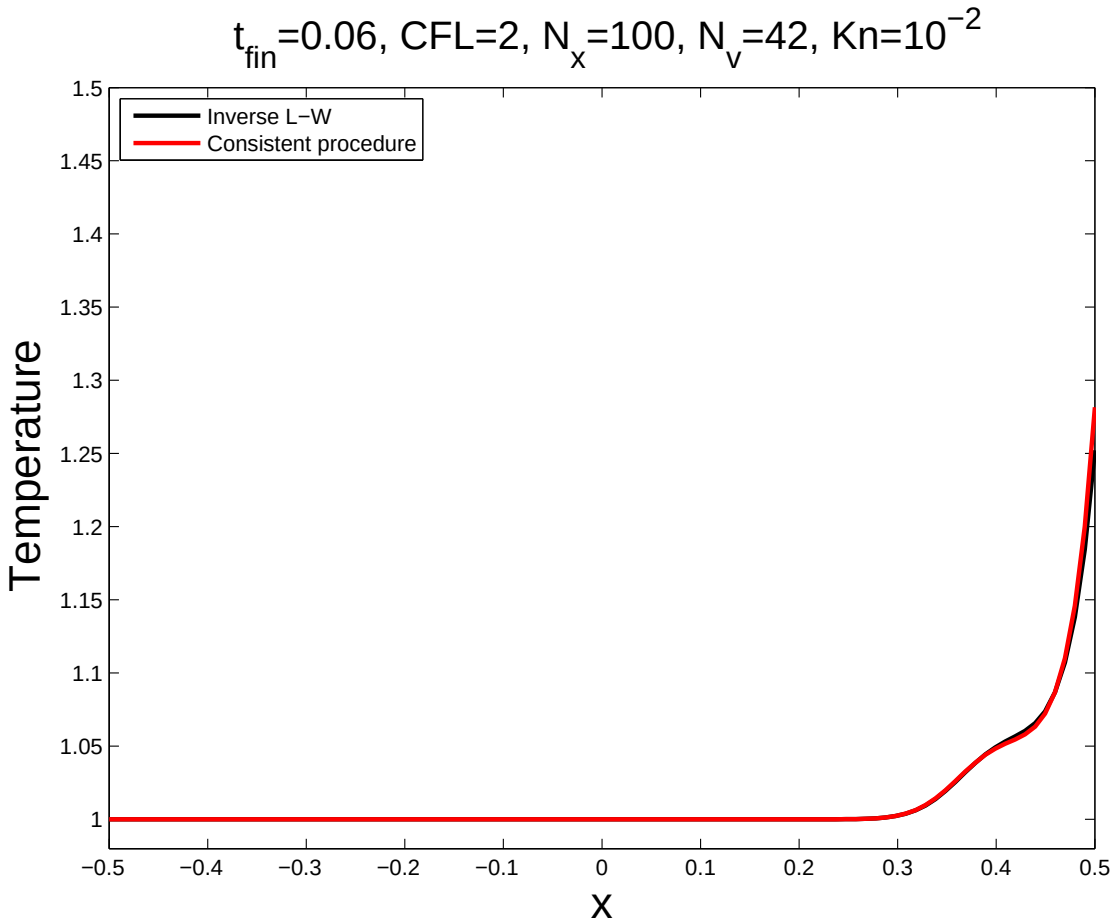
# Temperature, $Kn = 10^{-2}$. Comparison between IP and ILW. $t = 0.02$



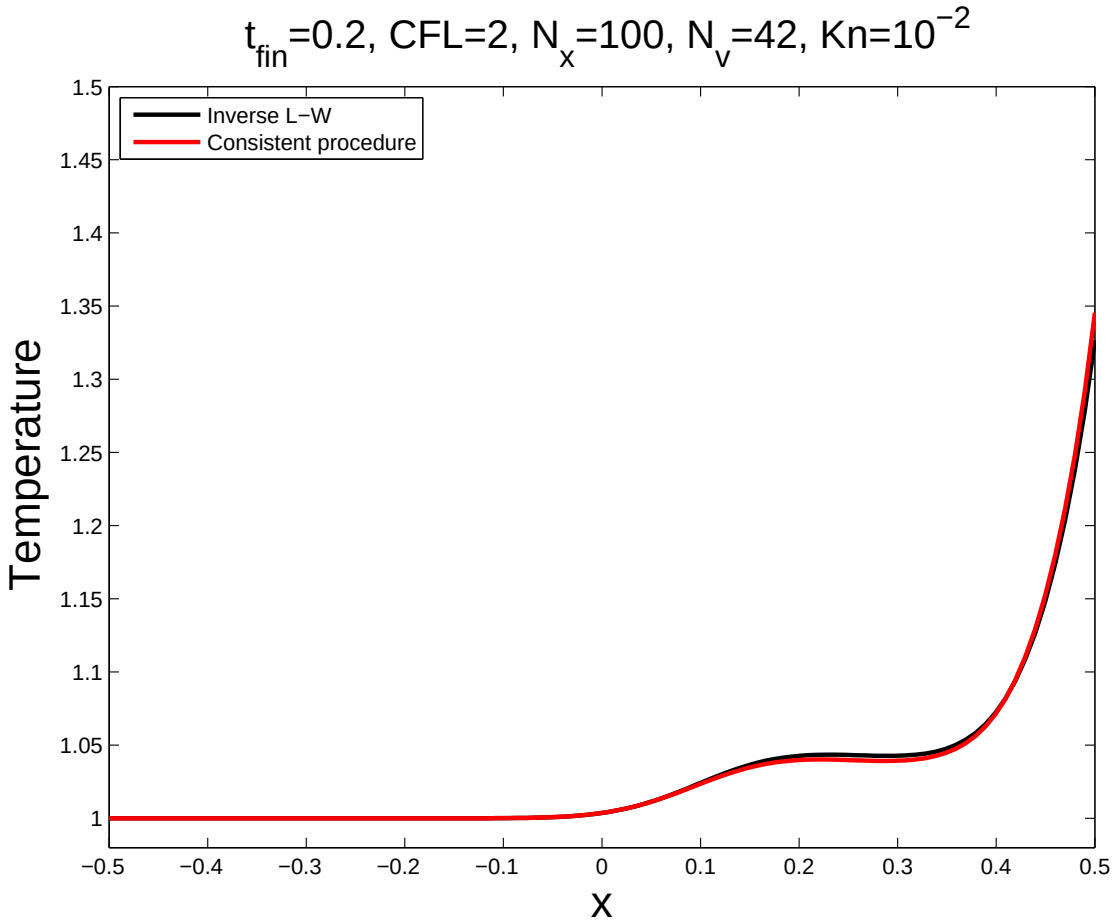
# Temperature, $Kn = 10^{-2}$. Comparison between IP and ILW. $t = 0.04$



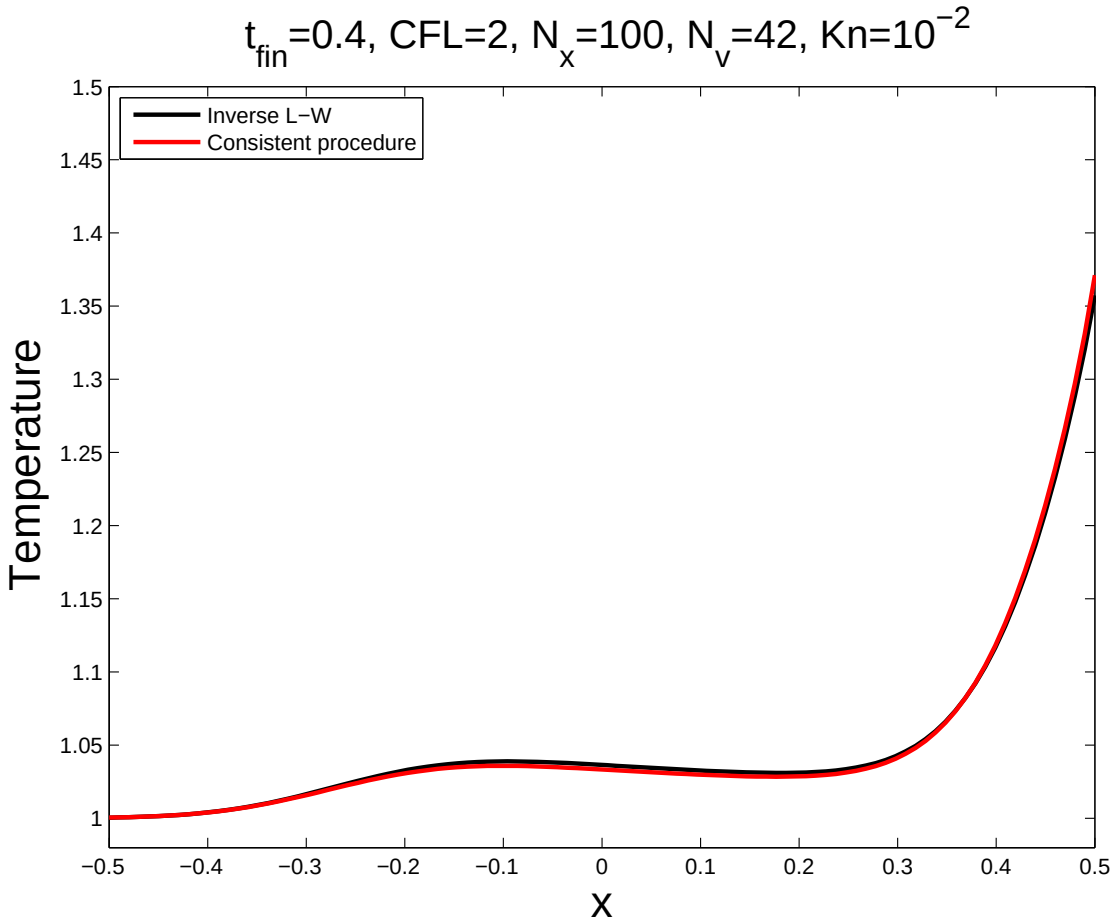
# Temperature, $Kn = 10^{-2}$. Comparison between IP and ILW. $t = 0.06$



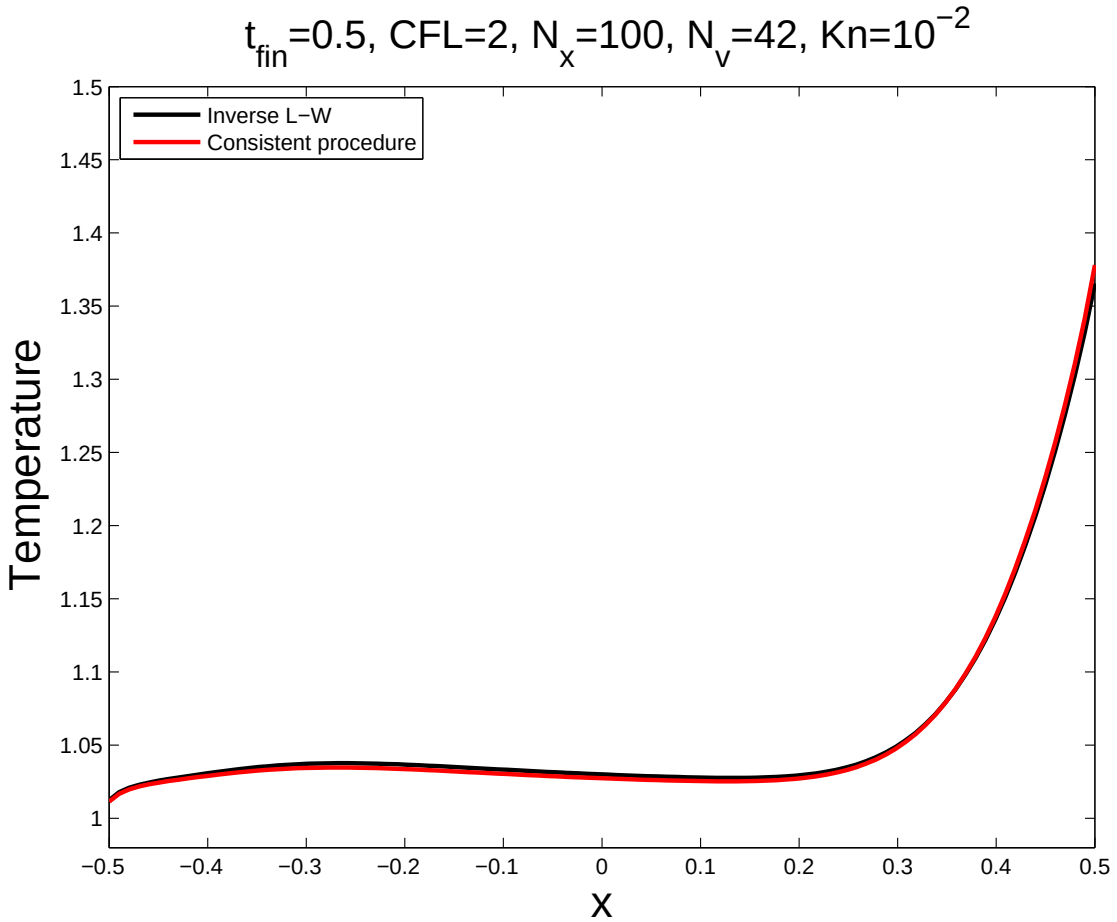
# Temperature, $Kn = 10^{-2}$. Comparison between IP and ILW. $t = 0.2$



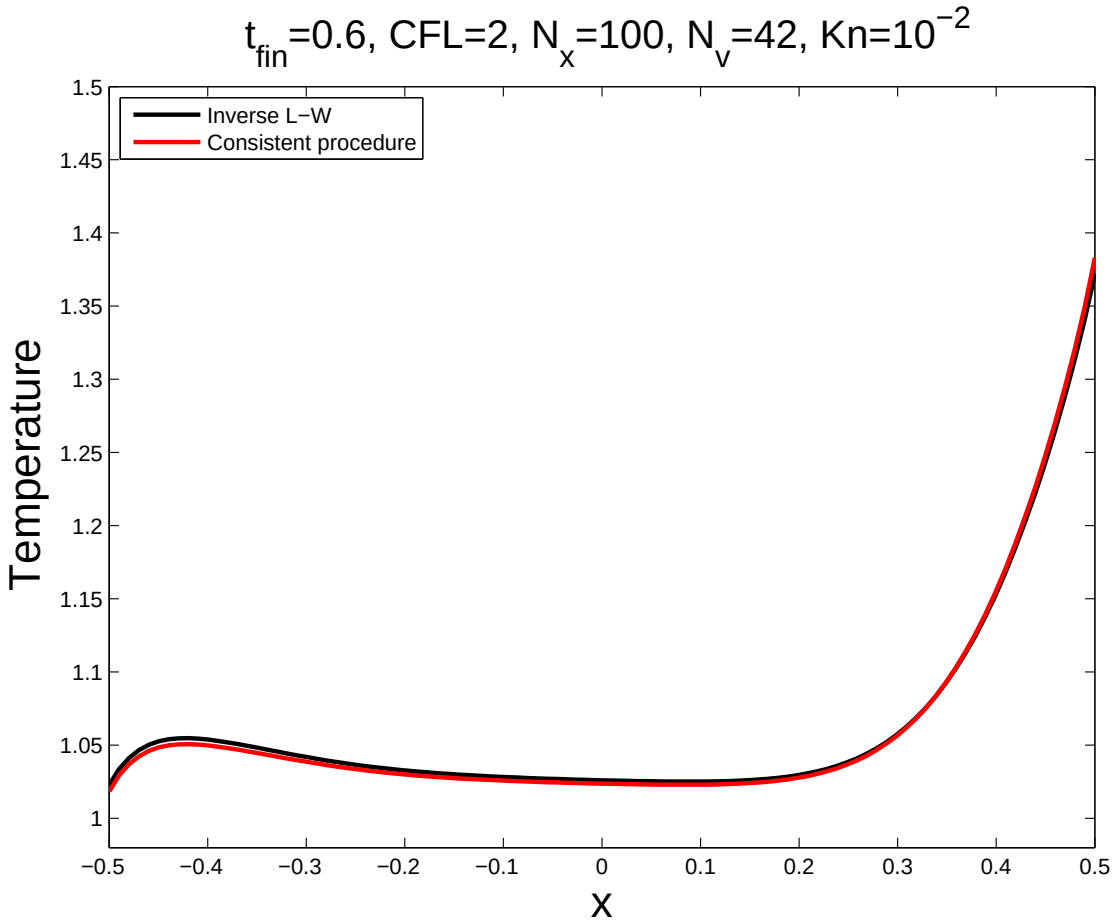
# Temperature, $Kn = 10^{-2}$. Comparison between IP and ILW. $t = 0.4$



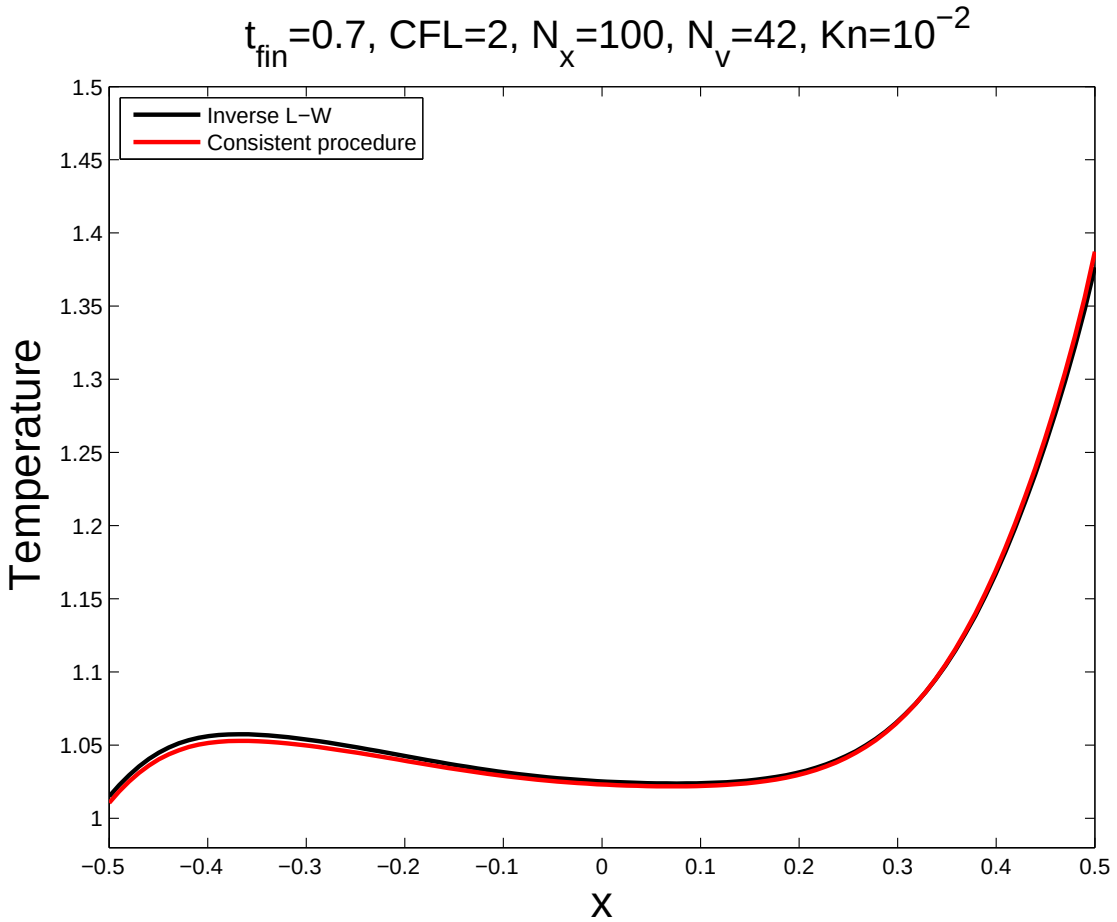
# Temperature, $Kn = 10^{-2}$. Comparison between IP and ILW. $t = 0.5$



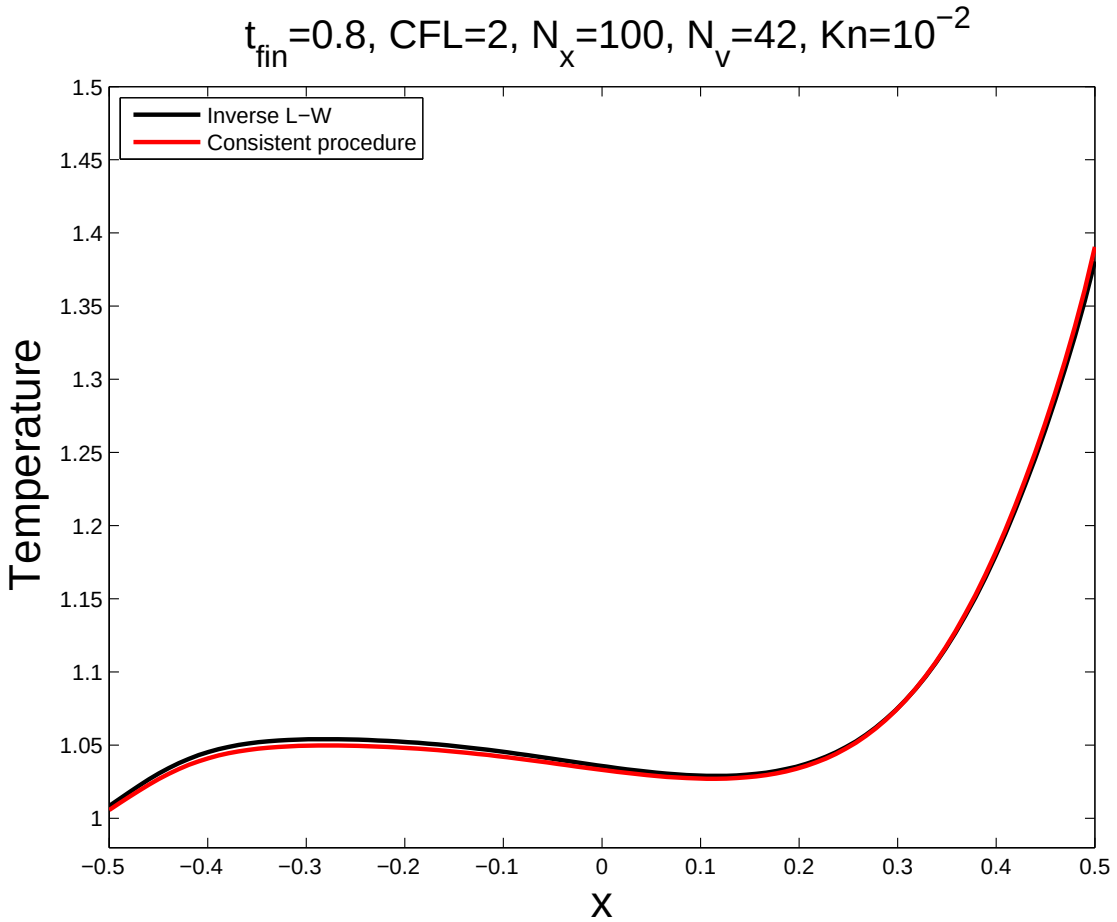
# Temperature, $Kn = 10^{-2}$. Comparison between IP and ILW. $t = 0.6$



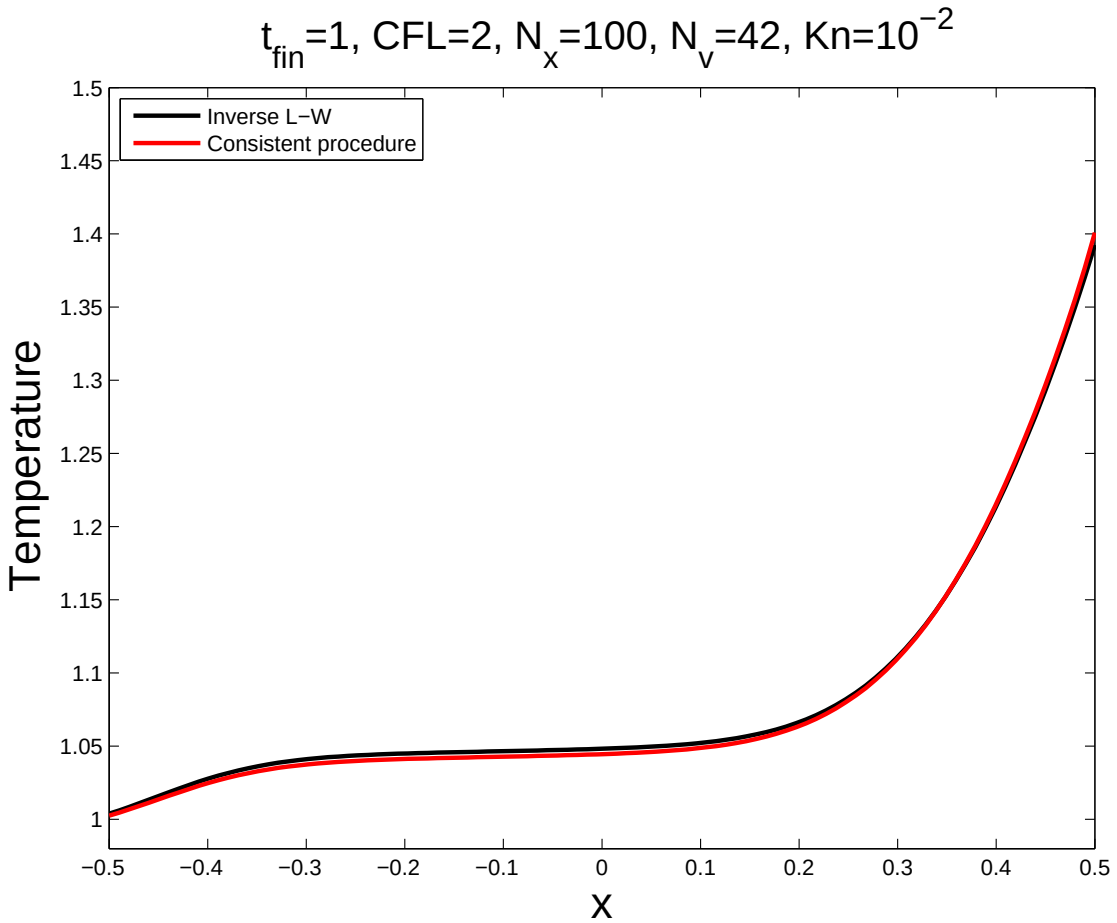
# Temperature, $Kn = 10^{-2}$. Comparison between IP and ILW. $t = 0.7$



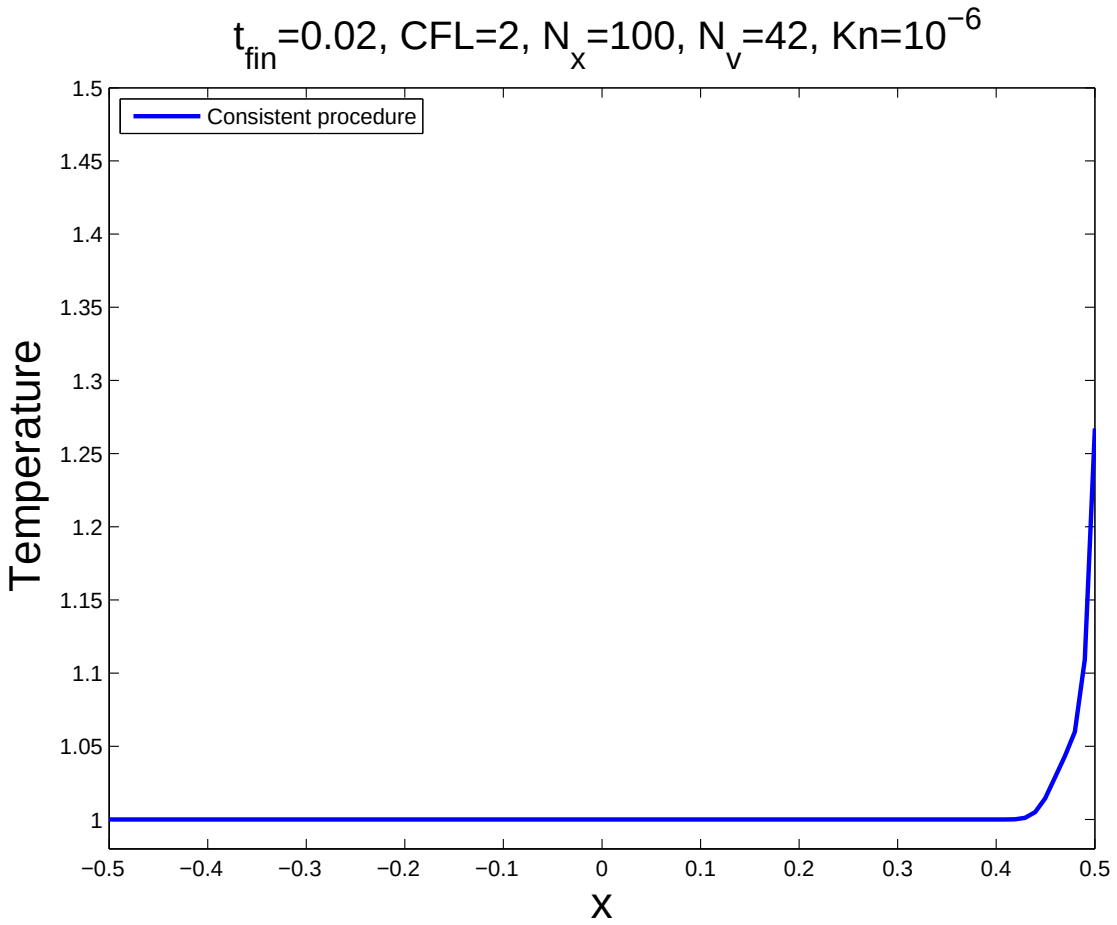
# Temperature, $Kn = 10^{-2}$. Comparison between IP and ILW. $t = 0.8$



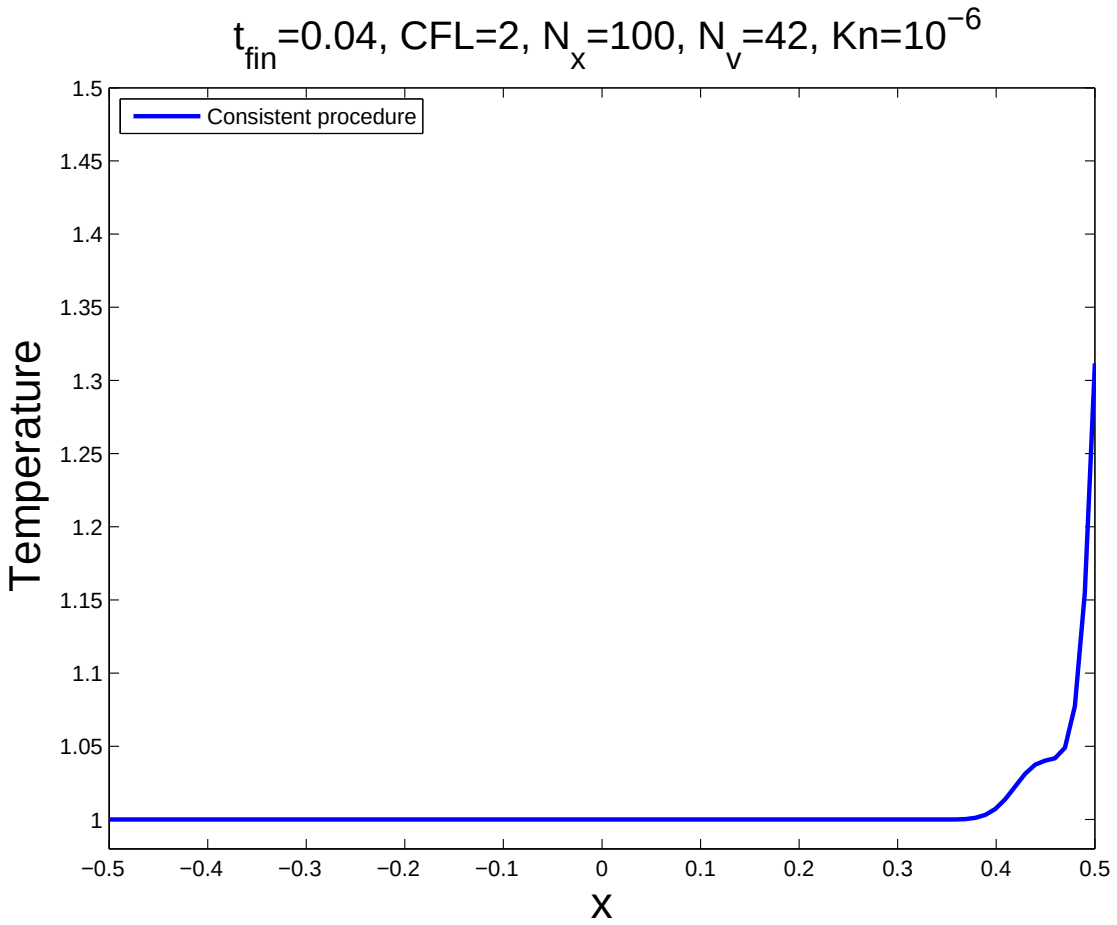
# Temperature, $Kn = 10^{-2}$. Comparison between IP and ILW. $t = 1$



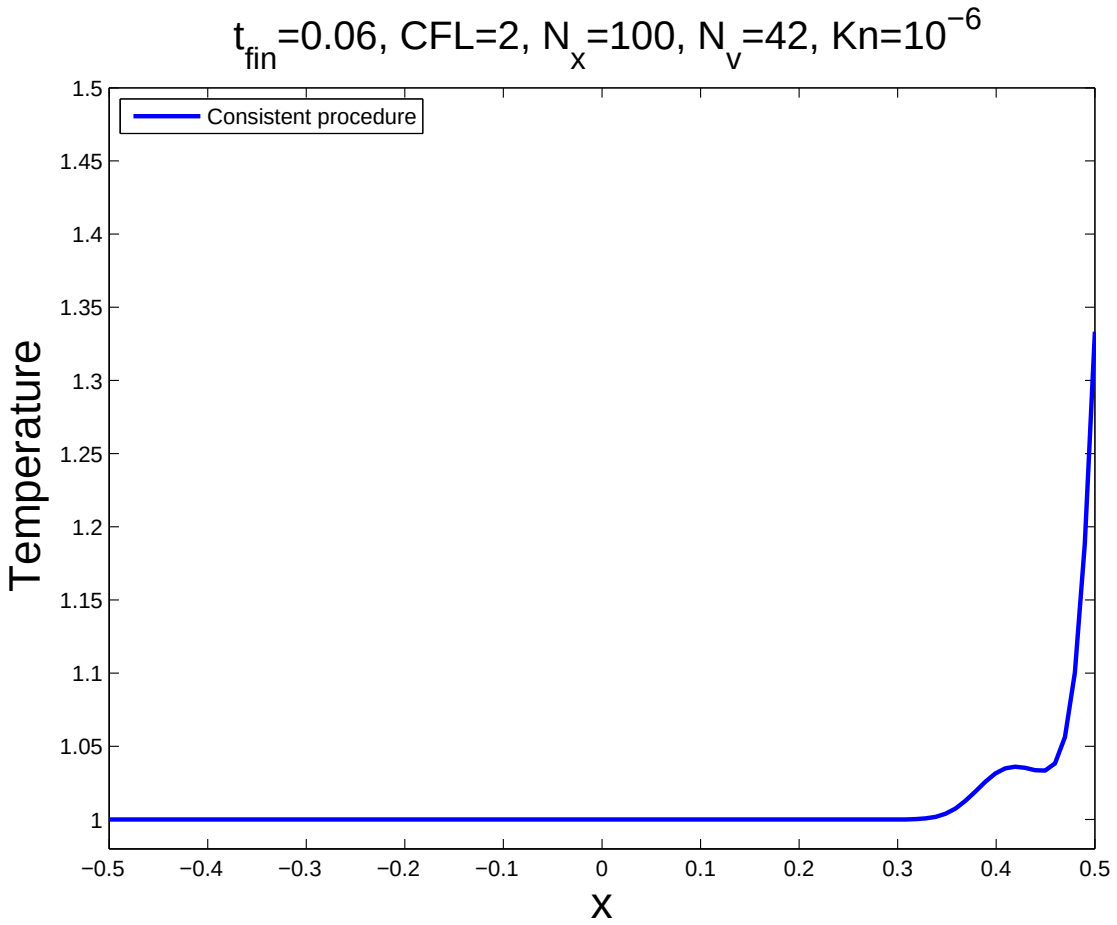
# Temperature, $Kn = 10^{-6}$. IP.
# $t = 0.02$



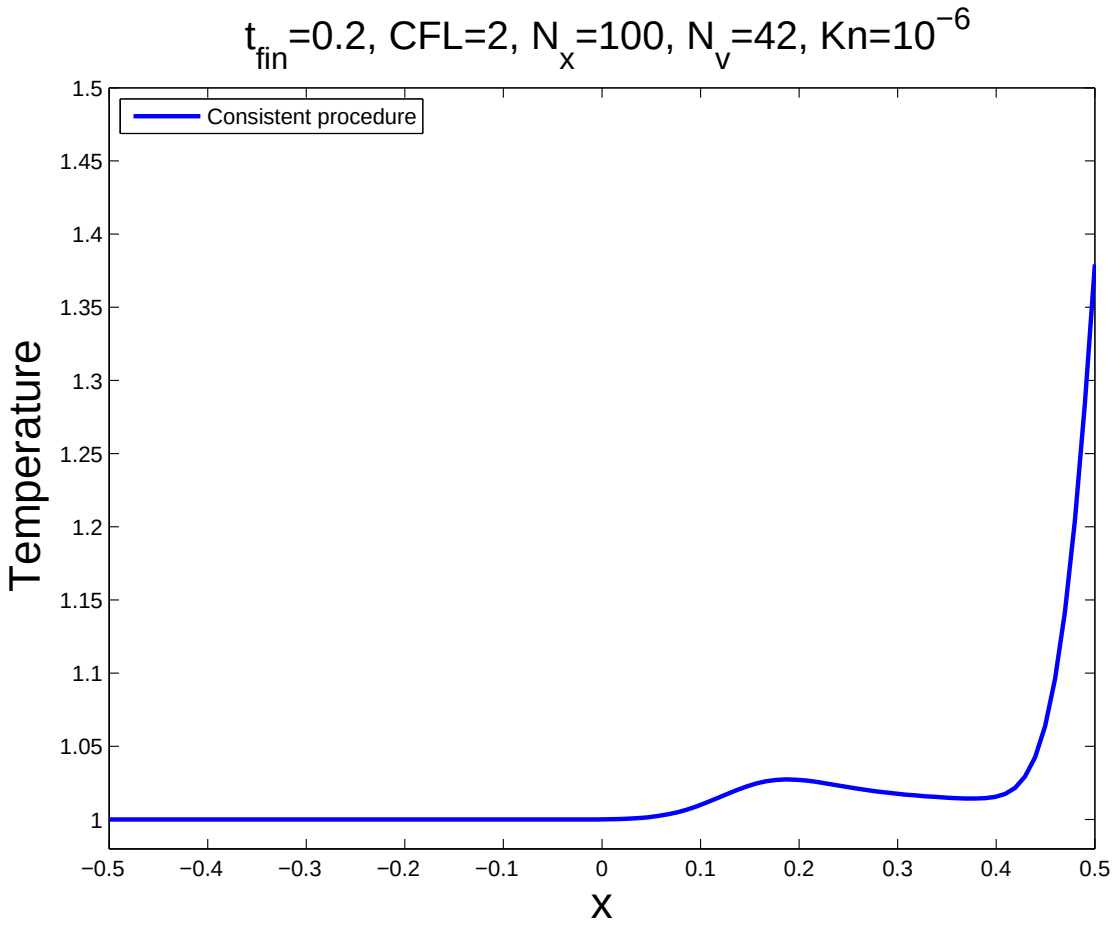
# Temperature, $Kn = 10^{-6}$. IP.
# $t = 0.04$



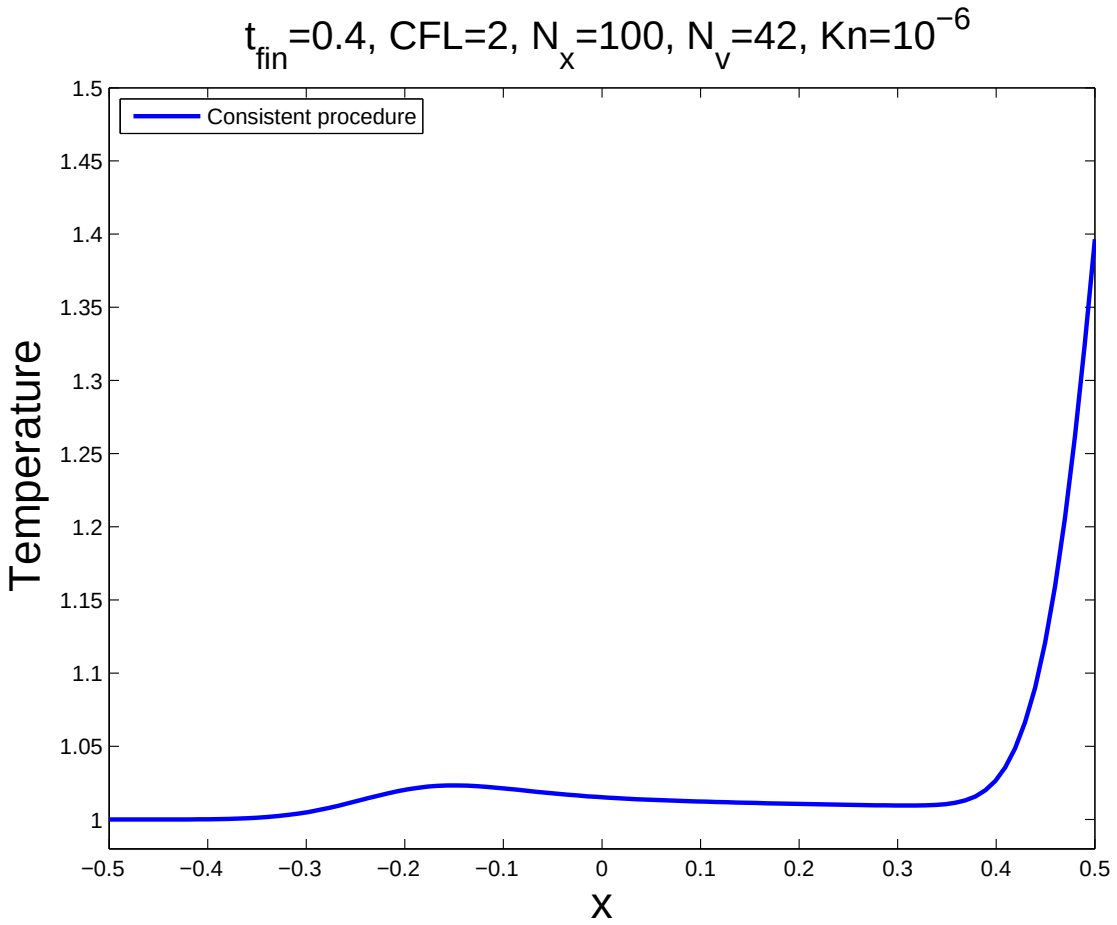
# Temperature, $Kn = 10^{-6}$. IP.
# $t = 0.06$



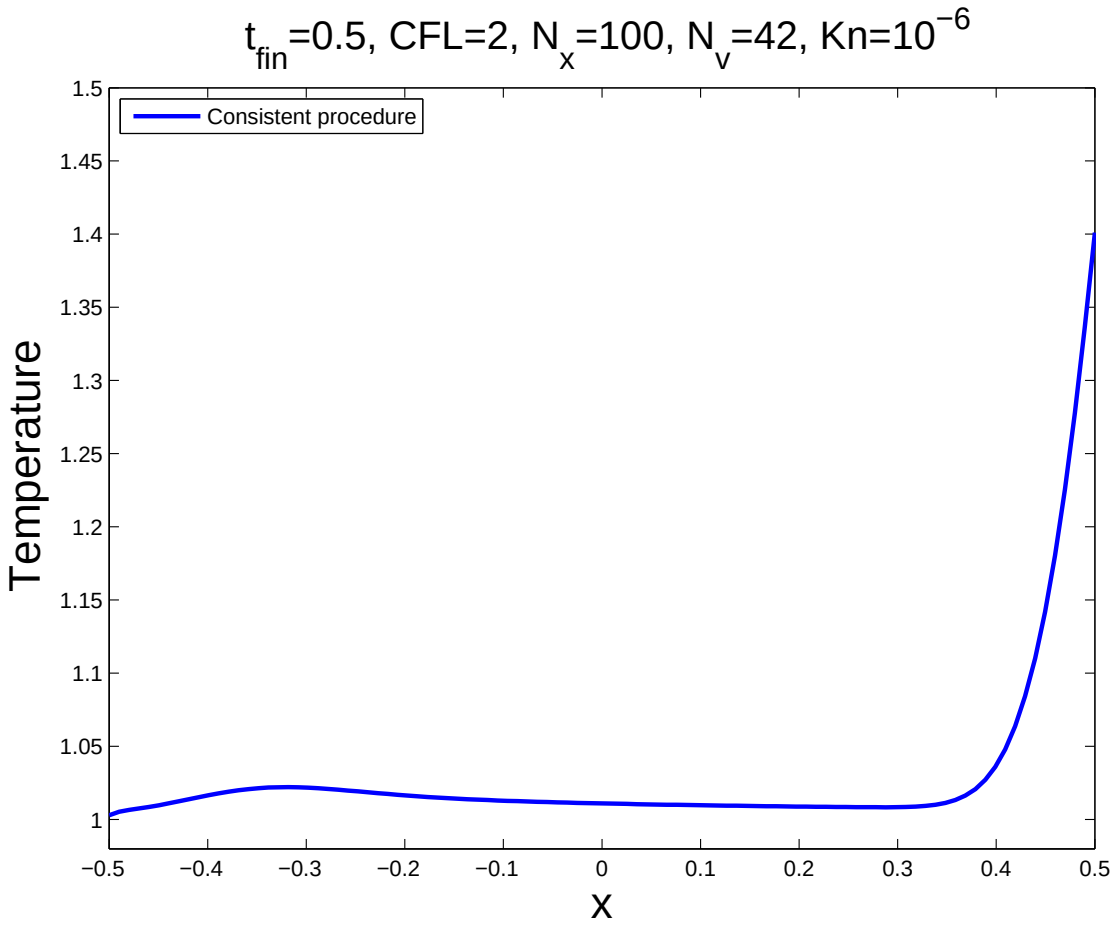
# Temperature, $Kn = 10^{-6}$. IP.
# $t = 0.2$



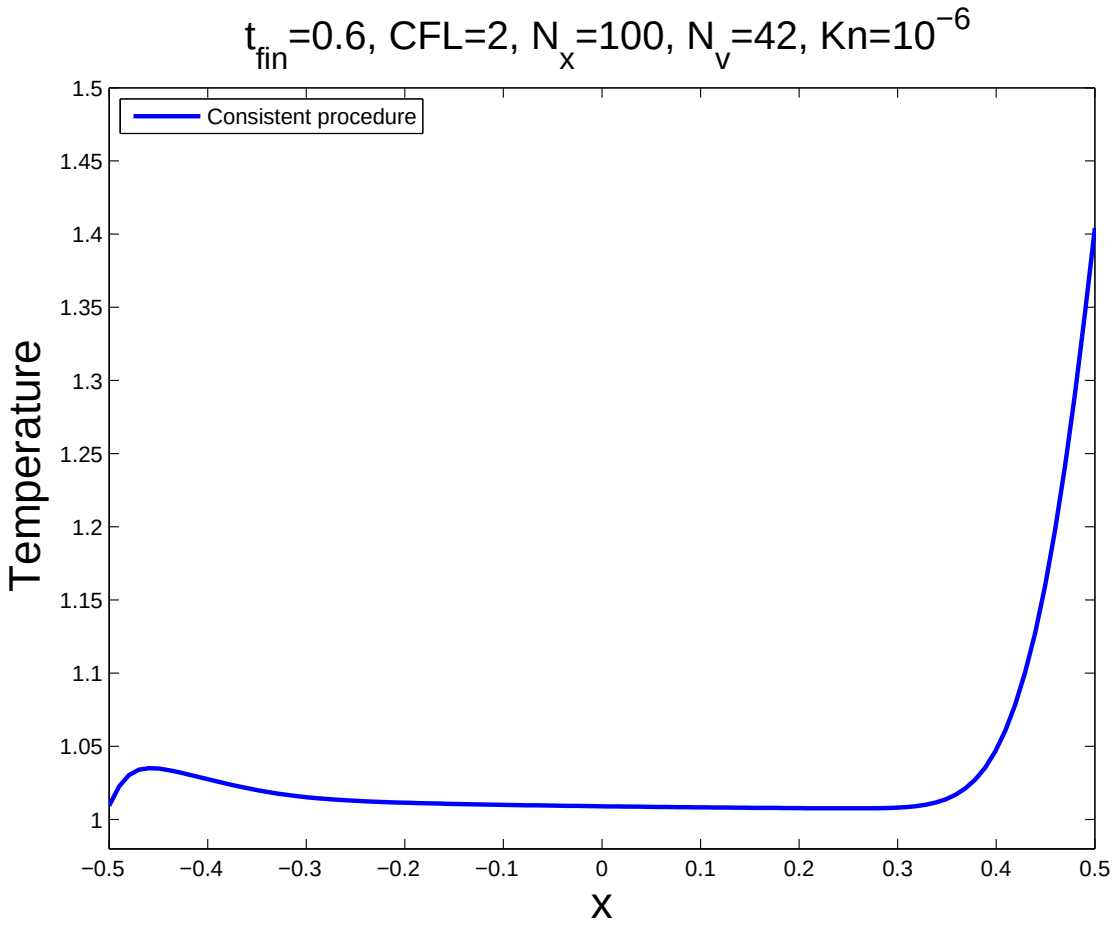
# Temperature, $Kn = 10^{-6}$. IP.
$t = 0.4$



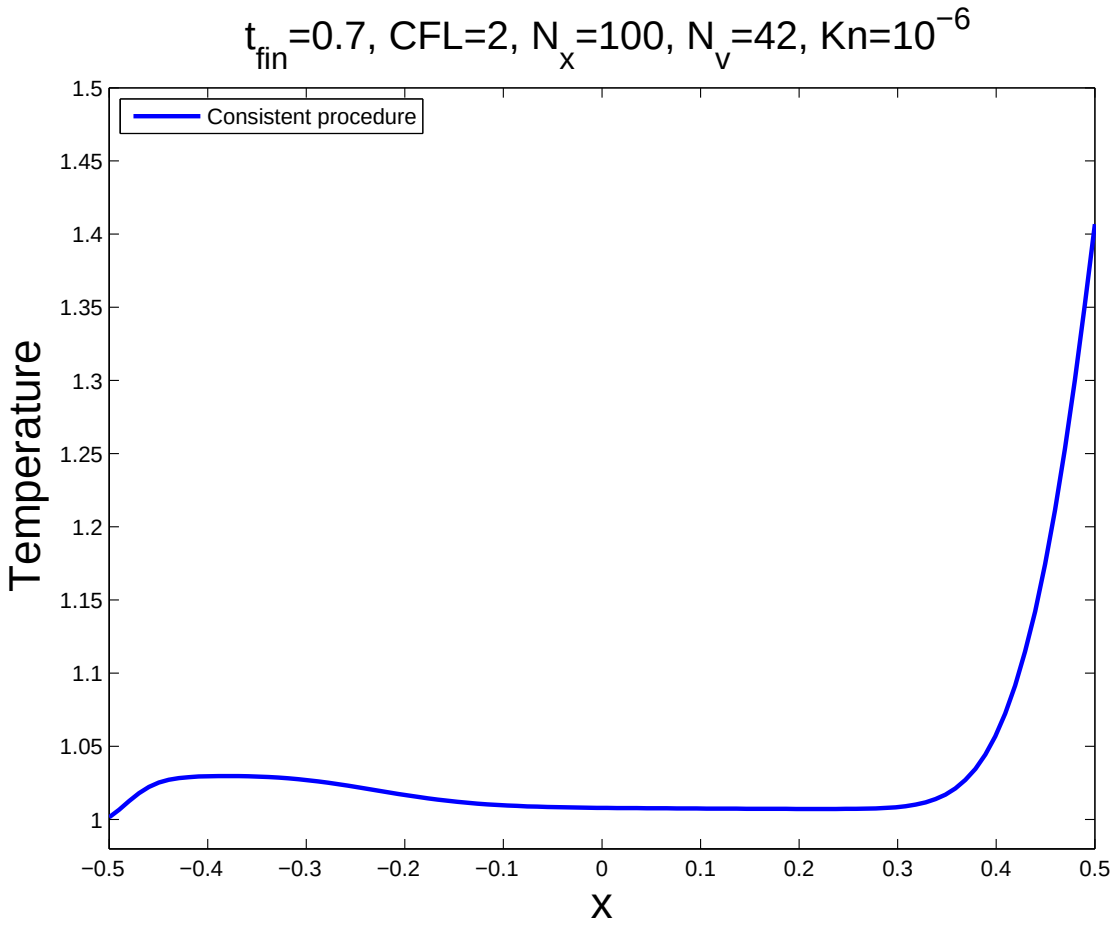
# Temperature, $Kn = 10^{-6}$. IP.
# $t = 0.5$



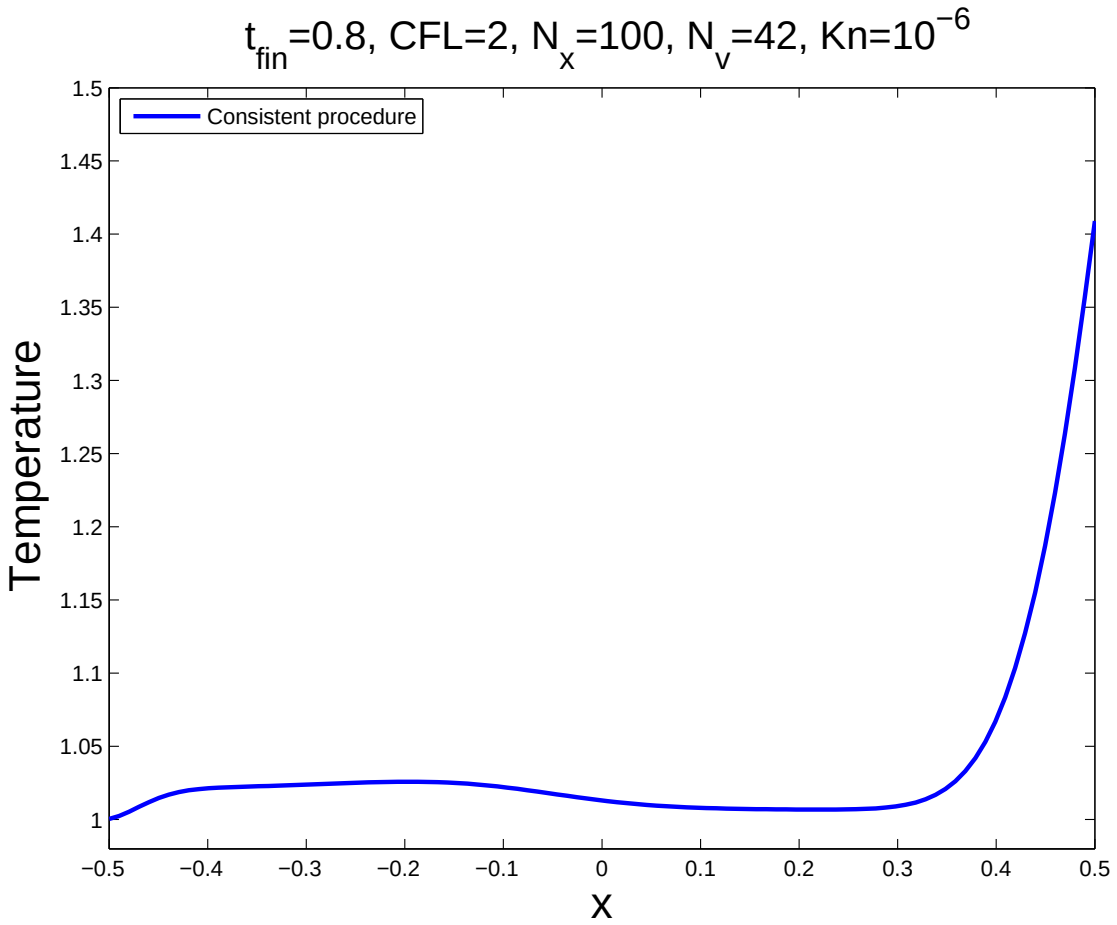
# Temperature, $Kn = 10^{-6}$. IP.
# $t = 0.6$



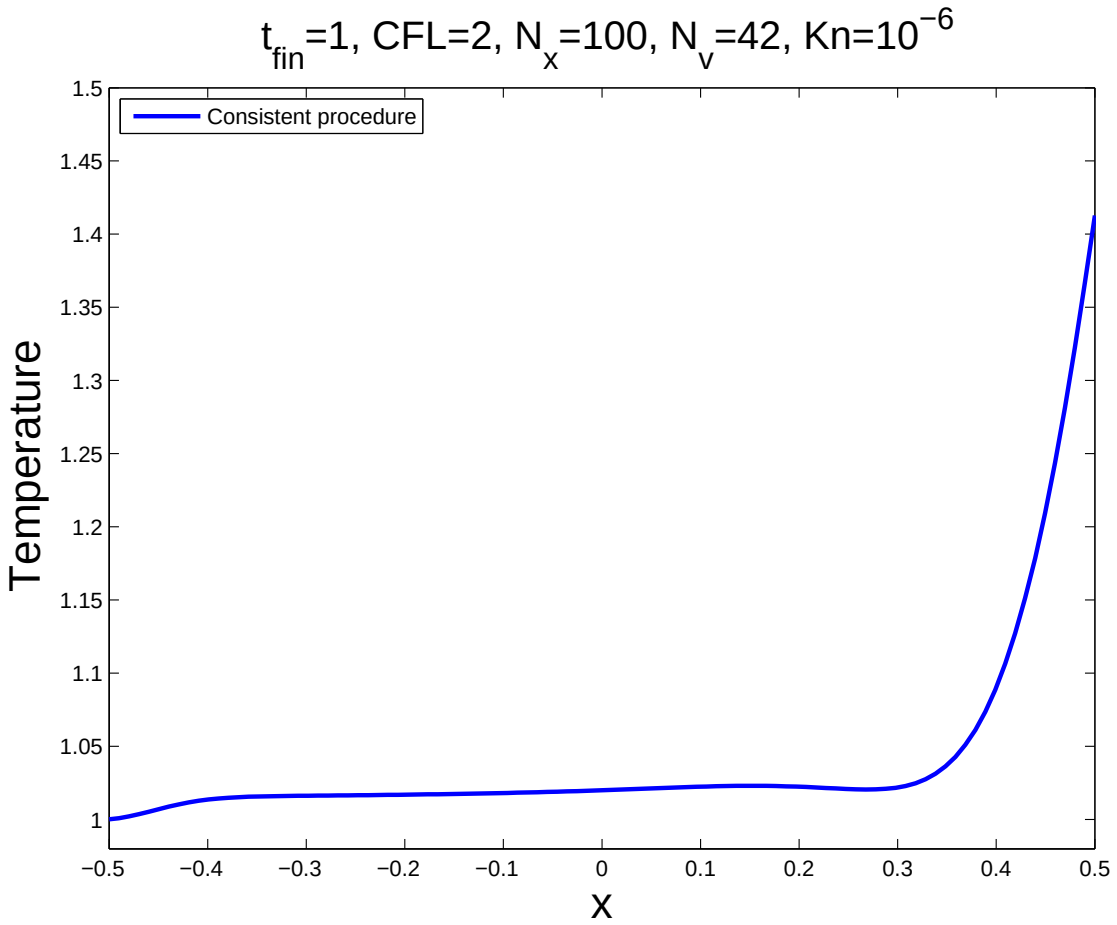
# Temperature, $Kn = 10^{-6}$. IP.
$t = 0.7$



# Temperature, $Kn = 10^{-6}$. IP.
$t = 0.8$

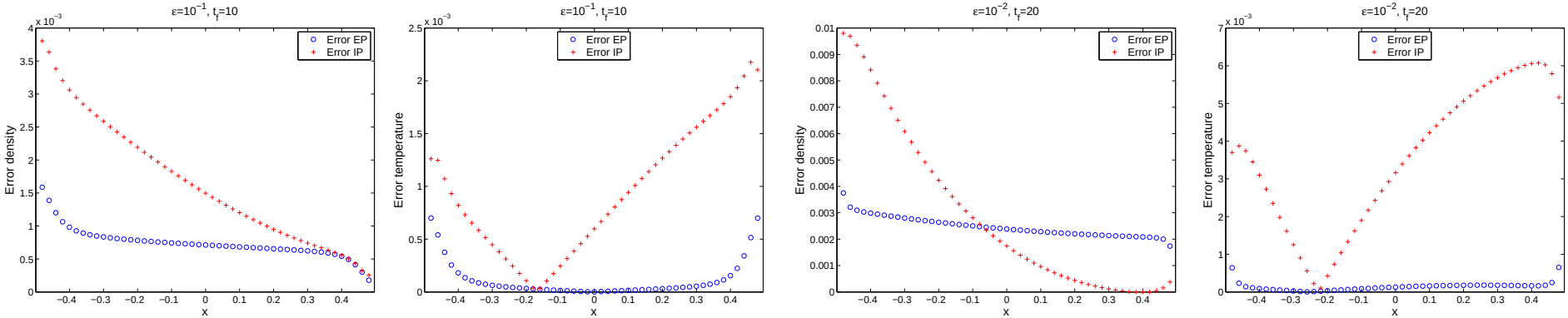


# Temperature, $Kn = 10^{-6}$. IP.
$t = 1$



# Comparison between IP and EP

In general EP is more efficient, especially for small Knudsen number, when IP requires several iterations.
In some cases EP turns out to be even more accurate. We show just one test.




Error in density and temperature profiles generated by a temperature gradient obtained from IP and EP methods (with $N_x = 50$), at $t_f = 10$ if $\varepsilon = 10^{-1}$, and at $t_f = 20$ if $\varepsilon = 10^{-2}$,CFL=1, with respect to a reference solution obtained by ILW technique (with $N_x = 200$).

# Application to BGK models for mixture

Mixture composed of $r$ species.
$f_s$, $s = 1, ..., r$: distribution function of each component.
A consistent BGK model for gas mixtures is given by the following system of kinetic equations (Andries, Aoki, Perthame, 2002):

$$\frac{\partial f_s}{\partial t} + \underline{v} \cdot \nabla_x f_s = \frac{1}{\epsilon_s}\left(M_s - f_s\right),\ s = 1, ..., r \tag{8}$$

where $M_s$ is a local *auxiliary* Maxwellian, depending on auxiliary fields $\tilde{n}_s$, $\underline{\tilde{u}}_s$ e $\tilde{T}_s$ which can be made explicit in terms of the actual macroscopic fields $n_s$, $\underline{u}_s$ and $T_s$ of the distribution functions $f_s$.

$$M_s = \tilde{n}_s\left(\frac{m_s}{2\pi K\tilde{T}_s}\right)^{3/2} exp\left(-\frac{m_s}{2K\tilde{T}_s}(\underline{v} - \underline{\tilde{u}}_s)^2\right),\quad s = 1, ..., r.$$

# Numerical Test for gas mixture

We considered a mixture of four gases with the following values of the molecular masses:

$$m_1 = 58.5, \quad m_2 = 18, \quad m_3 = 40, \quad m_4 = 36.5.$$

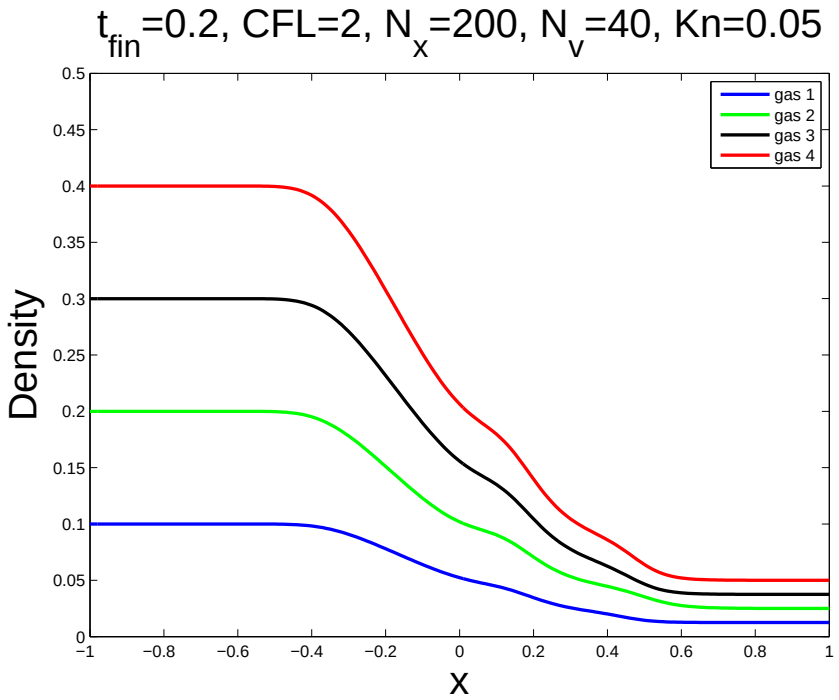
As initial data we have chosen Maxwellians reproducing the following moments (Riemann problem):

$$(\rho_0, u_0, p_0) = \begin{cases} (1, 0, 5/3), & x < 0.5, \\ (1/8, 0, 1/6), & x > 0.5, \end{cases}$$

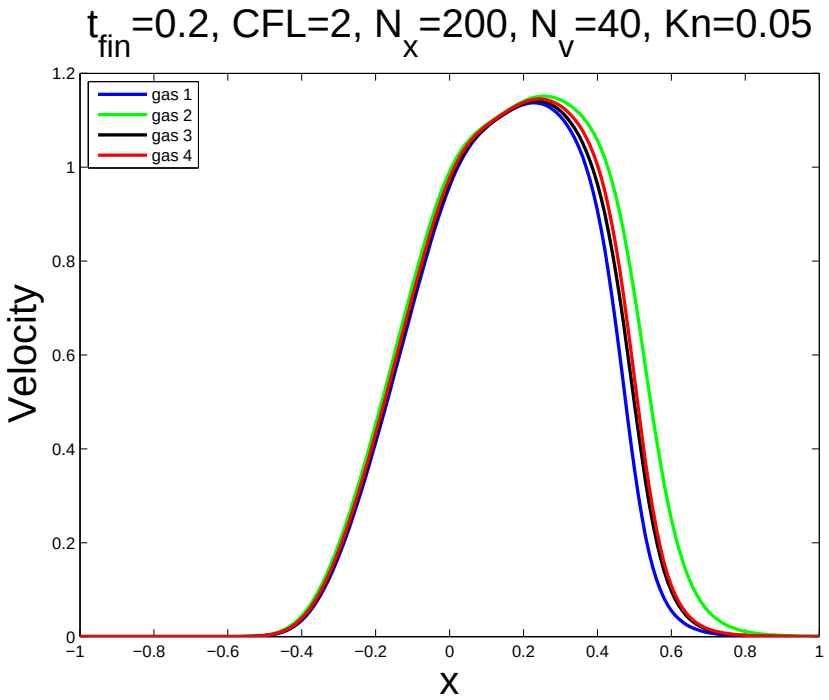
$$(\rho_{01}, \rho_{02}, \rho_{03}, \rho_{04}) = \begin{cases} (1/10, 2/10, 3/10, 4/10), & x < 0.5, \\ (1/80, 2/80, 3/80, 4/80), & x > 0.5, \end{cases},$$

$$u_{0i} = 0, \ i = 1, ..., 4.$$

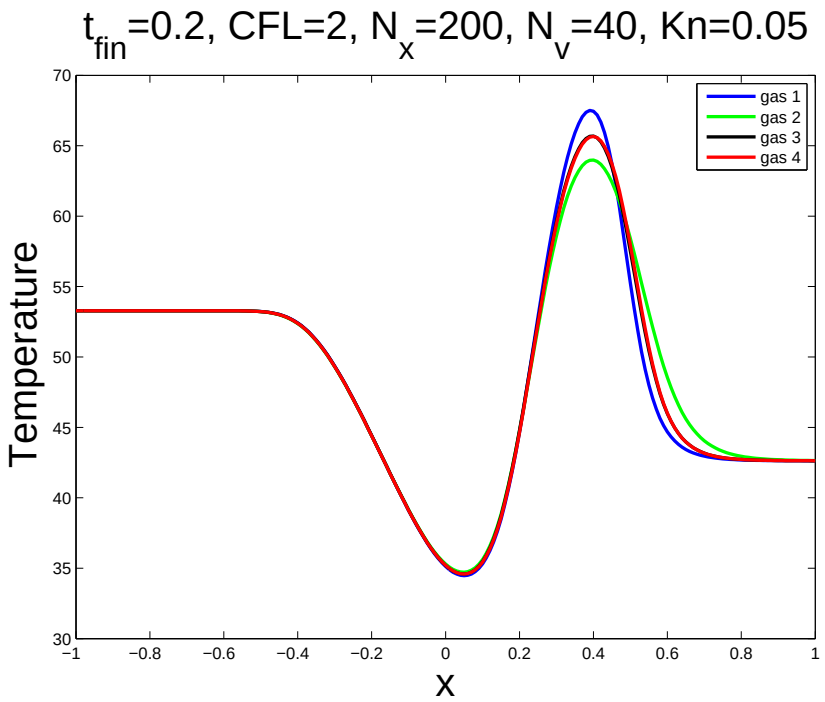
# Density, Kn=0.05



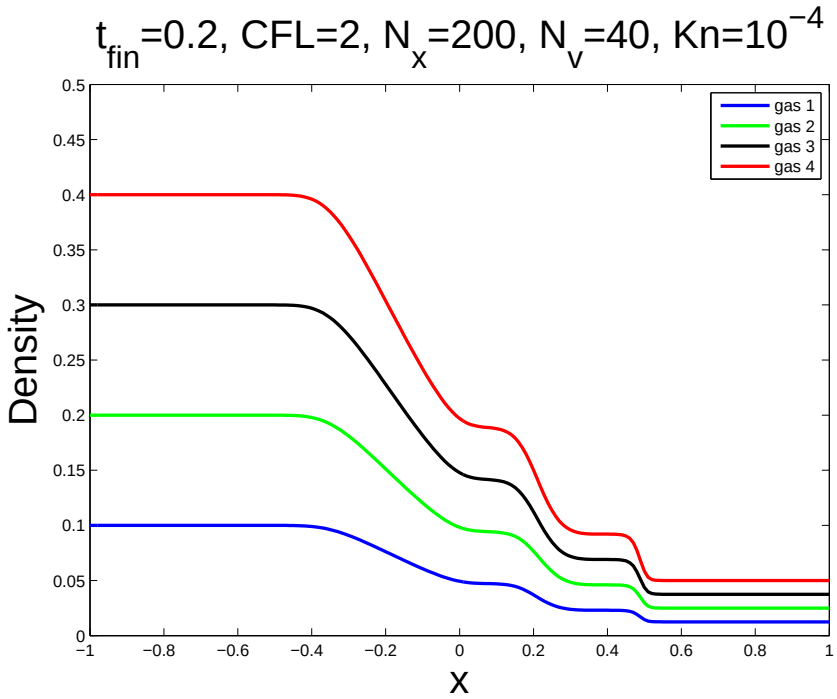
# Velocity, $Kn = 0.05$



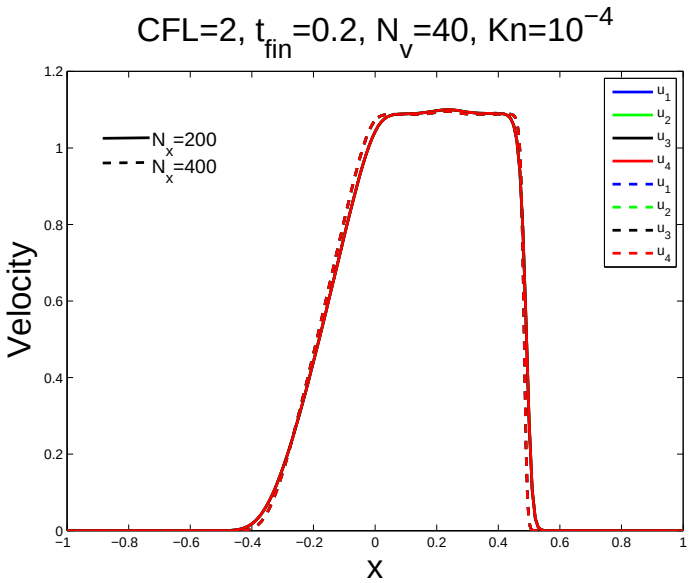
# Temperature, $Kn = 0.05$

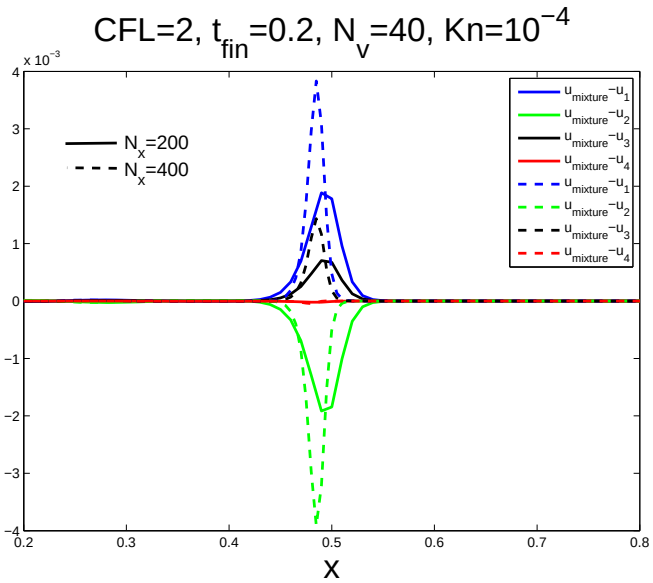


# Density, $Kn = 10^{-4}$

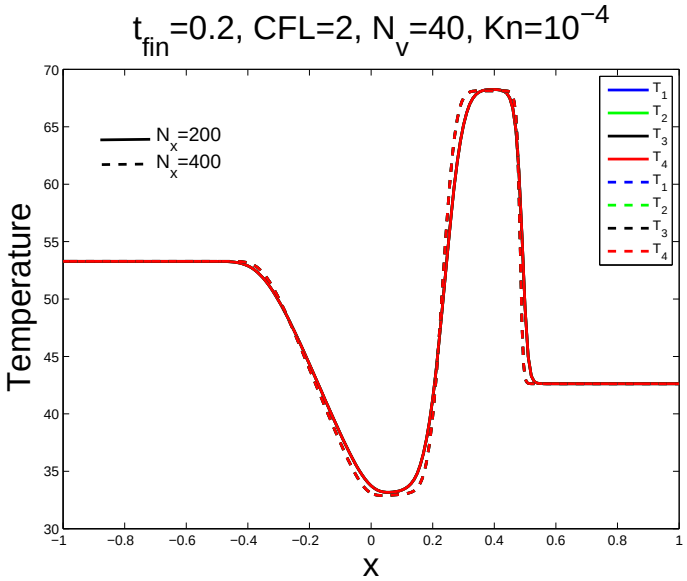


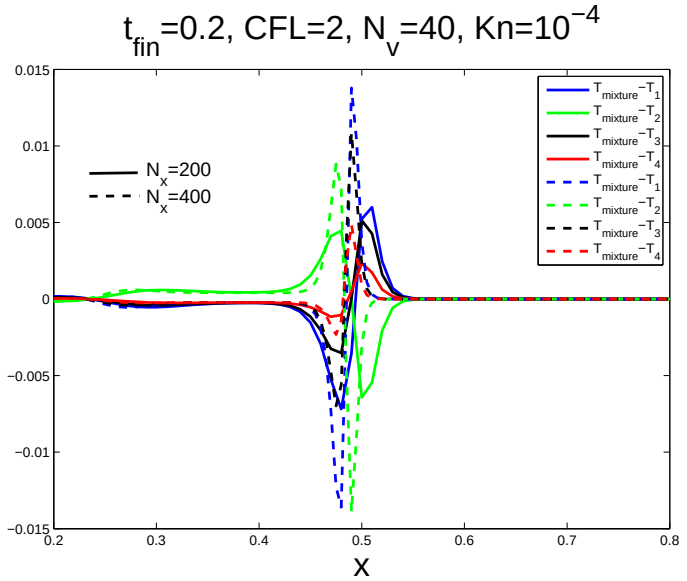
# Velocity, $Kn = 10^{-4}$



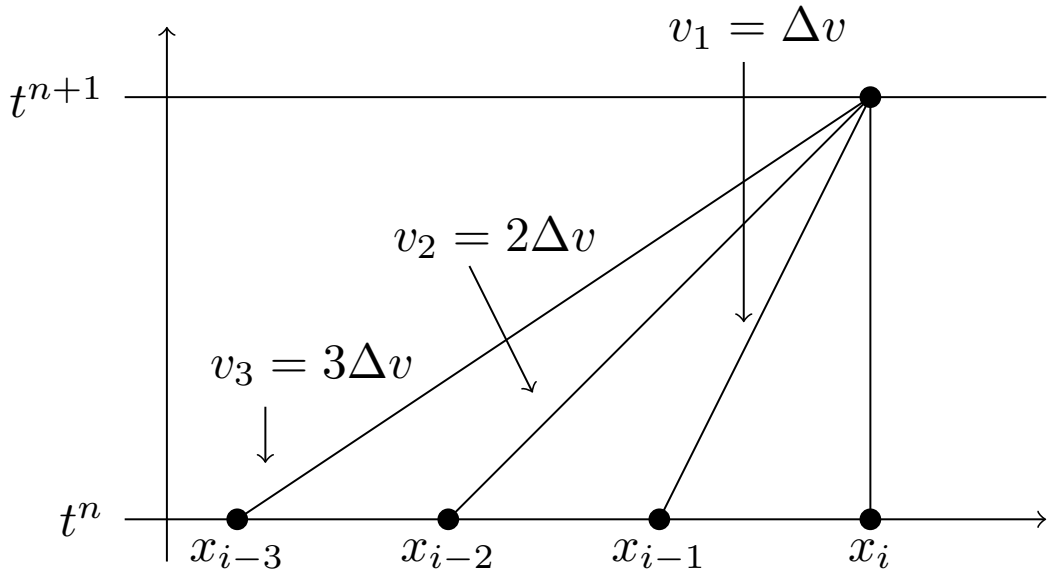
# Temperature, $Kn = 10^{-4}$



# SL schemes without interpolation

- Computational cost of the schemes is mainly due to the interpolation;
- To reduce the cost: look for schemes that avoid interpolation;
- First order: choosing $\Delta v \Delta t = \Delta x$ the characteristics connect grid points in space (see figure).



# SL schemes without interpolation

- Accuracy order can be increased using BDF or RK time integration;
- With the choice $\Delta v \Delta t = \Delta x$, which implies $CFL = N_v$, BDF schemes can be easily adapted in this setting;
- RK schemes: $\Delta v \Delta t = s\Delta x$, $s \in \mathbb{N}$ and each component of $\underline{c}$ must be multiple of $1/s$.
- The restriction on the $c$ coefficients, and stability requirements rule out second order RK schemes with $s = 2$ and third order schemes with $s = 3$.

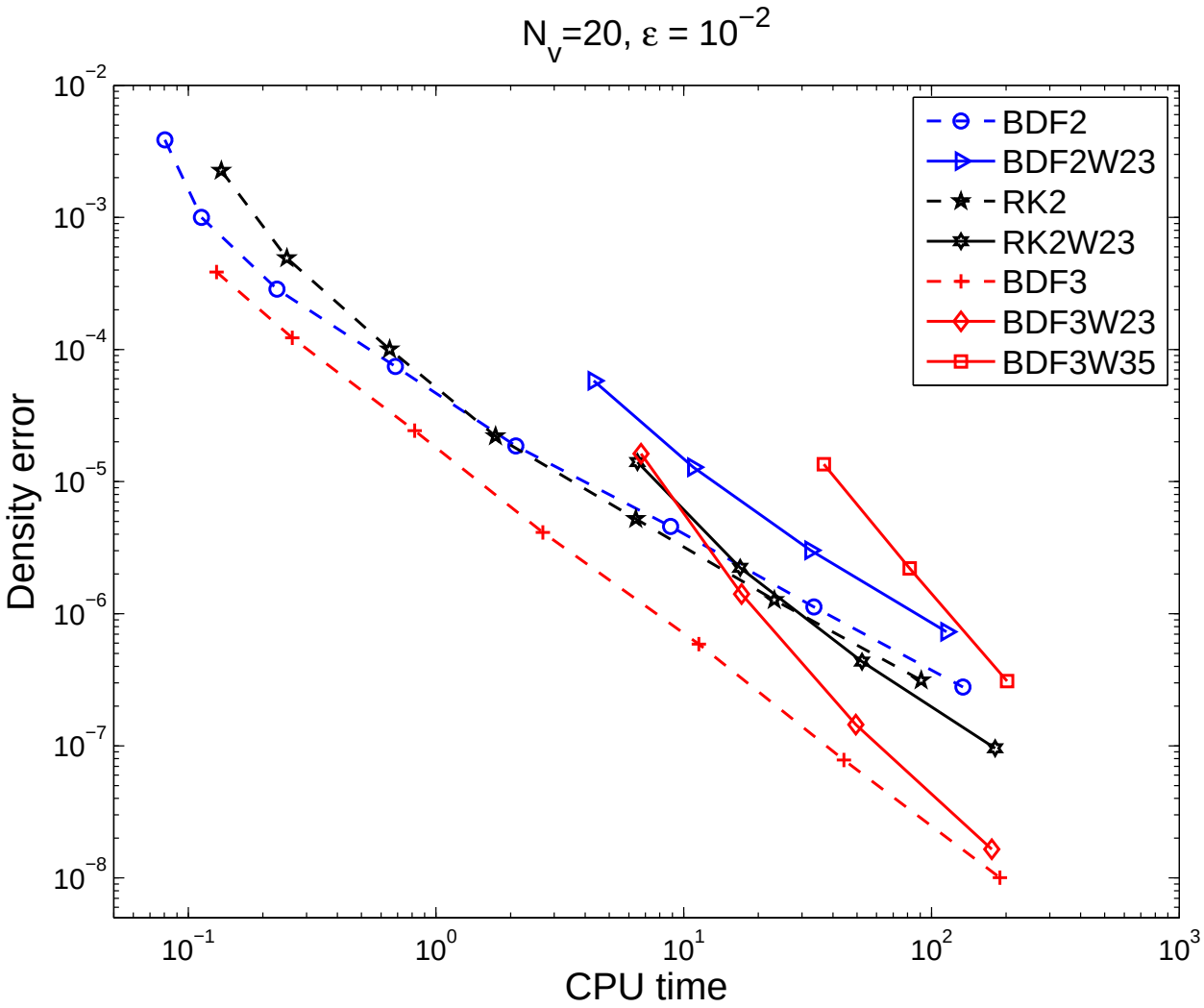
A second order $s = 3$ and third order $s = 4$ $L$-stable schemes:

$$T_2 = \begin{array}{c|cc} 1/3 & 1/3 & 0 \\ 1 & 3/4 & 1/4 \\ \hline & 3/4 & 1/4 \end{array} \qquad A_3 = \begin{array}{c|ccc} 1/2 & 1/2 & 0 & 0 \\ 3/4 & 1/2 & 1/4 & 0 \\ 1 & 5/3 & -4/3 & 2/3 \\ \hline & 5/3 & -4/3 & 2/3 \end{array}$$

# Remarks:

- The bad:
  - DIRK schemes avoiding interpolation require CFL number fixed to $sN_v$ therefore space may be over-resolved.
  - For non rectangular geometry interpolation is needed near the boundady (ghost points).
- The good:
  - Extremely efficient and simple to code in rectangular geometry, especially if few points in velocity space are needed.
  - Even for complex geometry the schemes without interpolation could be used far from the boundary.

# Cost-accuracy comparison between schemes with and without interpolation (smooth solution)




CPU time and $L_\infty$ error varying $N$

# Convergence proofs

A convergence proof for a first order version of the semilagrangian scheme has been obtained in collaboration with S.B.Yun and P.Santagati (G.R., P. Santagati, S.-B. Yun, SIAM J. Numer. Anal. 2012).
BGK equation in dimensions $d$ in space and velocity.
Initial value problem in periodic domain in space.
Discretize phase space in time, space, and velocity (by $\Delta t$, $\Delta x$, and $\Delta v$).
The first order scheme can be written as

$$f^{n+1}_{i,j,R} = \frac{\kappa}{\kappa + \triangle t}\ \tilde{f}^{n}_{i,j,R} + \frac{\triangle t}{\kappa + \triangle t}\ \mathcal{M}^{n}_{i,j}(\tilde{f}^{n}_{R}), \quad (n = 1, ..., N_x - 1) \tag{9}$$

where $\kappa =$ Knudsen number, and $\tilde{f}$ and $\tilde{M}$ are suitably defined. The discrete function is extended as piecewise constant function in the whole phase space.

The main result of the paper is an error estimate of the form

$$\|f(\cdot,\cdot,T_f) - f_R^{\triangle}(\cdot,\cdot,T_f)\|_{L^1_q} \leq C\left(\Delta x + \Delta v + \Delta t + \frac{\Delta x + \Delta v}{\Delta t} + \frac{1}{(1+R)^{q+1}}\right)$$

where

$$\|f\|_{L^1_q} \equiv \int_{\mathbb{T}^d \times \mathbb{R}^d} f(x,v,t)(1+|v|)^q \, dv \, dx$$

and $R$ is a parameter denoting the truncation of the domain in velocity space. The proof is rather technical. The starting point is to write a recurrence relation from time $t$ to time $t + \Delta t$, both for the exact solution and for the numerical solution, and to estimate term by term, until a local estimate is obtained. From the local estimate, by iteration, one obtains the global estimate.

Result is not optimal since it requires $\Delta x, \Delta v \to 0$ faster than $\Delta t$. It also requires a restriction on $\Delta t$ on the Knudsen number $\kappa$.

Work in progress: SL scheme for the ES-BGK skip

# Work in progress: SL scheme for the ES-BGK model

We plan to work both on a computational and theoretical point of view.

$$\begin{aligned} \partial_t f + v \cdot \nabla f &= \frac{1}{\kappa} A_\nu (\mathcal{M}_\nu(f) - f), \\ f(x, v, 0) &= f_0(x, v), \end{aligned} \tag{10}$$

Phase point $(x, v) \in \mathbb{T}^{d_1} \times \mathbb{R}^{d_2}$ $(d_1 \leq d_2)$ at time $t \in \mathbb{R}_+$.
Collision frequency: $A_\nu = (1-\nu)^{-1}$. The ellipsoidal Gaussian $\mathcal{M}_\nu(f)$ reads:

$$\mathcal{M}_\nu(f) = \frac{\rho}{\sqrt{\det(2\pi \mathcal{T}_\nu)}} \exp\left( -\frac{1}{2}(v-U)^\top \mathcal{T}_\nu^{-1} (v-U) \right),$$

The temperature tensor $\mathcal{T}_\nu$ is given by a convex combination of $T$ and $\Theta$:

$$\mathcal{T}_\nu = (1-\nu) T I_d + \nu \Theta,$$

where $I_d$ is the $d_2 \times d_2$ identity matrix, and

$$\rho(x,t)\Theta(x,t) = \int_{\mathbb{R}^{d_2}} f(x,v,t)(v-U) \otimes (v-U)\, dv$$

# Solution of the relaxtion operator

The first order semilagrangian scheme reads:

$$\frac{f_{i,j}^{n+1} - \tilde{f}_{i,j}^{n}}{\triangle t} = \frac{1}{\kappa} A_\nu \{ \mathcal{M}_j(f_i^{n+1}) - f_{i,j}^{n+1} \}.$$

Here $\tilde{f}_{s,j}^{n}$ denotes the linear interpolation:

$$\widetilde{f}_{i,j}^{n} = \frac{x(i,j) - x_s}{\triangle x} \, f_{s+1,j}^{n} + \frac{x_{s+1} - x(i,j)}{\triangle x} \, f_{s,j}^{n}. \tag{11}$$

The relaxation step can be explicitly solved, with a small error,

$$\Theta_i^{n+1} \approx \frac{\triangle t}{\kappa + \triangle t} \widetilde{T}_i^n I_d + \frac{\kappa}{\kappa + \triangle t} \widetilde{\Theta}_i^n,$$

Therefore $\mathcal{M}_j(f_i^{n+1})$ can be explicitly approximated, and $f_{i,j}^{n+1}$ computed:

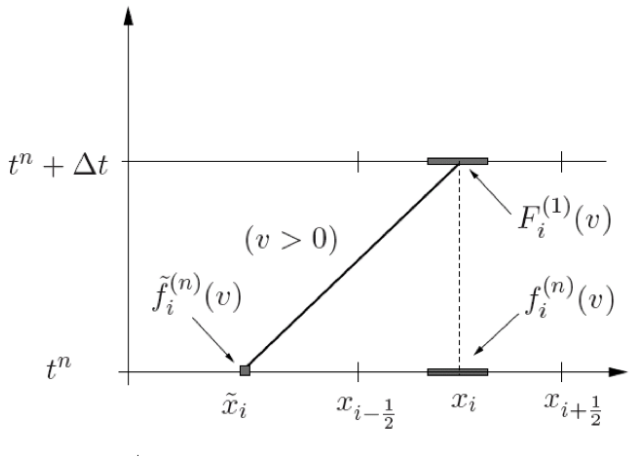
$$f_{i,j}^{n+1} = \frac{\kappa}{\kappa + A_\nu \triangle t} \widetilde{f}_{i,j}^{n} + \frac{A_\nu \triangle t}{\kappa + A_\nu \triangle t} \mathcal{M}_{\widetilde{\nu},j}(\tilde{f}_i^n).$$

# Remarks

- The limitation on the support in velocity is removed for simplicity of the analysis
- Similar limitation on the space step and on the time step are present. A finer analysis is needed to in order to remove them (if possible)
- Proof of convergence (with S.B.Yun) and high order extensions (with S.B.Yun and S.Boscarino) are in progress .
- We are investigating the use of methods that do not require interpolation, which should simplify the analysis and perhaps allow to obtain sharper results, and possibly simplify the analysis for higher order schemes in time.

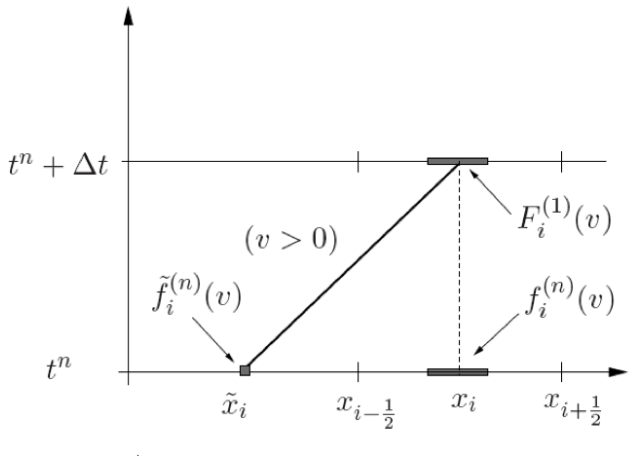
# Conservative correction

General technique to construct conservative schemes starting from non-conservative schemes. Can be adopted both at *finite volume* or *conservative finite difference* level



# Conservative correction

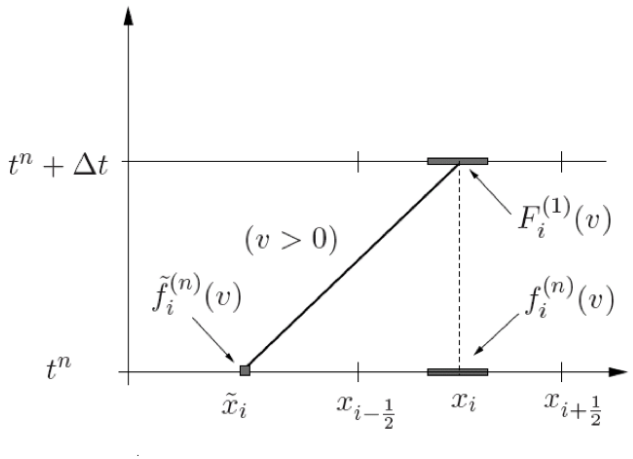
General technique to construct conservative schemes starting from non-conservative schemes. Can be adopted both at *finite volume* or *conservative finite difference* level




1. compute a predictor value at the center of the cell

# Conservative correction

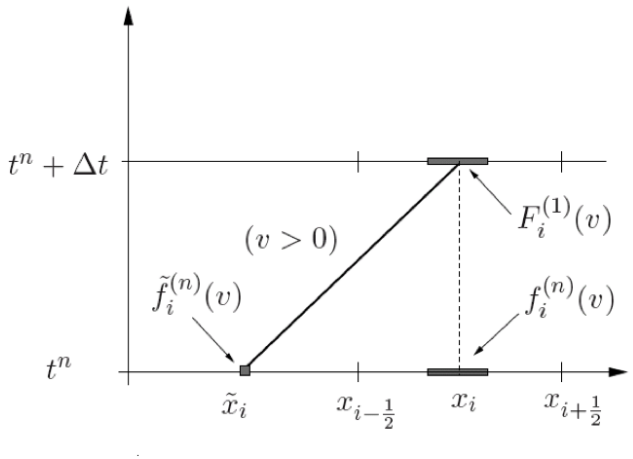
General technique to construct conservative schemes starting from non-conservative schemes. Can be adopted both at *finite volume* or *conservative finite difference* level




1. compute a predictor value at the center of the cell
2. use such a predictor to perform reconstruction of the fluxes, at cell edges

# Conservative correction

General technique to construct conservative schemes starting from non-conservative schemes. Can be adopted both at *finite volume* or *conservative finite difference* level




1. compute a predictor value at the center of the cell
2. use such a predictor to perform reconstruction of the fluxes, at cell edges
3. evolve the conservative values according to the computed fluxes

# Application to conservation laws

Consider a system of conservation laws

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

# Application to conservation laws

Consider a system of conservation laws

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

One can choose a predictor based on

# Application to conservation laws

Consider a system of conservation laws

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

One can choose a predictor based on

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

# Application to conservation laws

Consider a system of conservation laws

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

One can choose a predictor based on

$$\frac{\partial u}{\partial t} + A(u)\frac{\partial u}{\partial x} = 0 \qquad \text{or even} \qquad \frac{\partial v}{\partial t} + B(v)\frac{\partial v}{\partial x} = 0$$

# Application to conservation laws

Consider a system of conservation laws

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

One can choose a predictor based on

$$\frac{\partial u}{\partial t} + A(u)\frac{\partial u}{\partial x} = 0 \qquad \text{or even} \qquad \frac{\partial v}{\partial t} + B(v)\frac{\partial v}{\partial x} = 0$$

where $u = U(v)$ is an invertible mapping ($v = V(u)$ is the inverse) and the formulation in $v$ is somehow *simpler*.

Then one can apply a conservative correction using finite volume or finite difference discretization.

skip to stability

# Finite volume approach

# Finite volume approach

1. from $\{\bar{u}_j^n\}$ compute the pointwise values of $\{v_j^n\}$.

# Finite volume approach

1. from $\{\bar{u}_j^n\}$ compute the pointwise values of $\{v_j^n\}$.
2. evolve $v_j$ with a non conservative scheme (e.g. Runge-Kutta with $\nu$ stages)

$$v_j^{(l)} = v_j^{(1)} - \Delta t \sum_{k=1}^{l-1} a_{lk} B(v_j^{(k)})(D_x v^{(k)})_j, \quad j = 1, \ldots, N_x, \quad l = 1, \ldots, \nu,$$

$(D_x v^{(k)})_j$: numerical discretization of space derivative of $v(x, t^n + c_k \Delta t)$.

# Finite volume approach

1. from $\{\bar{u}_j^n\}$ compute the pointwise values of $\{v_j^n\}$.
2. evolve $v_j$ with a non conservative scheme (e.g. Runge-Kutta with $\nu$ stages)

$$v_j^{(l)} = v_j^{(1)} - \Delta t \sum_{k=1}^{l-1} a_{lk} B(v_j^{(k)})(D_x v^{(k)})_j, \quad j = 1, \ldots, N_x, \quad l = 1, \ldots, \nu,$$

   $(D_x v^{(k)})_j$: numerical discretization of space derivative of $v(x, t^n + c_k \Delta t)$.
3. Reconstruct (pointwise) the nonconservative variables at cell edges $v_{j+1/2}^{(k)\pm}$

# Finite volume approach

1. from $\{\bar{u}_j^n\}$ compute the pointwise values of $\{v_j^n\}$.
2. evolve $v_j$ with a non conservative scheme (e.g. Runge-Kutta with $\nu$ stages)

$$v_j^{(l)} = v_j^{(1)} - \Delta t \sum_{k=1}^{l-1} a_{lk} B(v_j^{(k)})(D_x v^{(k)})_j, \quad j = 1, \ldots, N_x, \quad l = 1, \ldots, \nu,$$

   $(D_x v^{(k)})_j$: numerical discretization of space derivative of $v(x, t^n + c_k \Delta t)$.
3. Reconstruct (pointwise) the nonconservative variables at cell edges $v_{j+1/2}^{(k)\pm}$
4. Compute the fluxes at cell edges: $f_{j+\frac{1}{2}}^{(k)} = F(u_{j+\frac{1}{2}}^{(k)-}, u_{j+\frac{1}{2}}^{(k)+}) = \tilde{F}(v_{j+\frac{1}{2}}^{(k)-}, v_{j+\frac{1}{2}}^{(k)+})$

# Finite volume approach

1. from $\{\bar{u}_j^n\}$ compute the pointwise values of $\{v_j^n\}$.
2. evolve $v_j$ with a non conservative scheme (e.g. Runge-Kutta with $\nu$ stages)

$$v_j^{(l)} = v_j^{(1)} - \Delta t \sum_{k=1}^{l-1} a_{lk} B(v_j^{(k)})(D_x v^{(k)})_j, \quad j = 1, \ldots, N_x, \quad l = 1, \ldots, \nu,$$

$(D_x v^{(k)})_j$: numerical discretization of space derivative of $v(x, t^n + c_k \Delta t)$.

3. Reconstruct (pointwise) the nonconservative variables at cell edges $v_{j+1/2}^{(k)\pm}$
4. Compute the fluxes at cell edges: $f_{j+\frac{1}{2}}^{(k)} = F(u_{j+\frac{1}{2}}^{(k)-}, u_{j+\frac{1}{2}}^{(k)+}) = \tilde{F}(v_{j+\frac{1}{2}}^{(k)-}, v_{j+\frac{1}{2}}^{(k)+})$
5. Evolve the conservative variables

$$\begin{aligned} \bar{u}_j^{n+1} &= \bar{u}_j^n - \frac{\Delta t}{\Delta x} \sum_{l=1}^{\nu} b_l K_l \\ K_l &= f_{j+\frac{1}{2}}^{(l)} - f_{j-\frac{1}{2}}^{(l)} \end{aligned}$$

# Finite difference approach

# Finite difference approach

1. compute $\{v_j^n = V(u_j)\}$.

# Finite difference approach

1. compute $\{v_j^n = V(u_j)\}$.
2. evolve $v_j$ with a non conservative scheme (e.g. Runge-Kutta with $\nu$ stages)

$$v_j^{(l)} = v_j^{(1)} - \Delta t \sum_{k=1}^{l-1} a_{lk} B(v_j^{(k)})(D_x v^{(k)})_j, \quad j = 1, \ldots, N_x, \quad l = 1, \ldots, \nu,$$

$(D_x v^{(k)})_j$: numerical discretization of space derivative of $v(x, t^n + c_k \Delta t)$.

# Finite difference approach

1. compute $\{v_j^n = V(u_j)\}$.
2. evolve $v_j$ with a non conservative scheme (e.g. Runge-Kutta with $\nu$ stages)

$$v_j^{(l)} = v_j^{(1)} - \Delta t \sum_{k=1}^{l-1} a_{lk} B(v_j^{(k)})(D_x v^{(k)})_j, \quad j = 1, \ldots, N_x, \quad l = 1, \ldots, \nu,$$

   $(D_x v^{(k)})_j$: numerical discretization of space derivative of $v(x, t^n + c_k \Delta t)$.
3. Compute the splitter fluxes $f^-$ and $f^+$ $(f^- + f^+ = f)$ at cell center $x_j$ at each stage $l$
4. Reconstruct (from cell average to pointwise) the fluxes $f^+$ and $f^-$ at cell edges $f_{j+\frac{1}{2}}^{(k)} = f^-(x_{j+1/2}^+, t^n + c_k \Delta t) + f^+(x_{j+1/2}^-, t^n + c_k \Delta t)$

# Finite difference approach

1. compute $\{v_j^n = V(u_j)\}$.
2. evolve $v_j$ with a non conservative scheme (e.g. Runge-Kutta with $\nu$ stages)

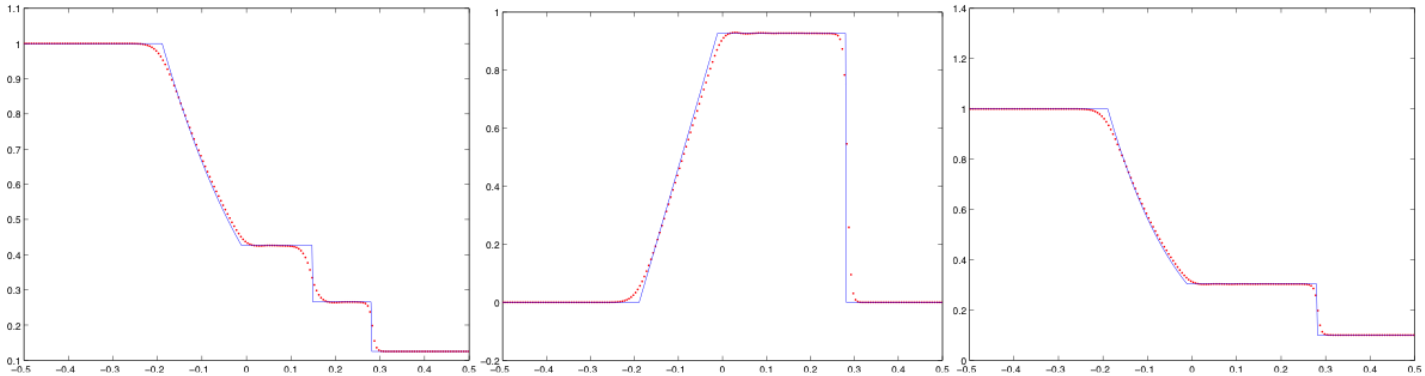
$$v_j^{(l)} = v_j^{(1)} - \Delta t \sum_{k=1}^{l-1} a_{lk} B(v_j^{(k)})(D_x v^{(k)})_j, \quad j = 1, \ldots, N_x, \quad l = 1, \ldots, \nu,$$

   $(D_x v^{(k)})_j$: numerical discretization of space derivative of $v(x, t^n + c_k \Delta t)$.
3. Compute the splitter fluxes $f^-$ and $f^+$ $(f^- + f^+ = f)$ at cell center $x_j$ at each stage $l$
4. Reconstruct (from cell average to pointwise) the fluxes $f^+$ and $f^-$ at cell edges $f_{j+\frac{1}{2}}^{(k)} = f^-(x_{j+1/2}^+, t^n + c_k \Delta t) + f^+(x_{j+1/2}^-, t^n + c_k \Delta t)$
5. Evolve the conservative pointwise variables

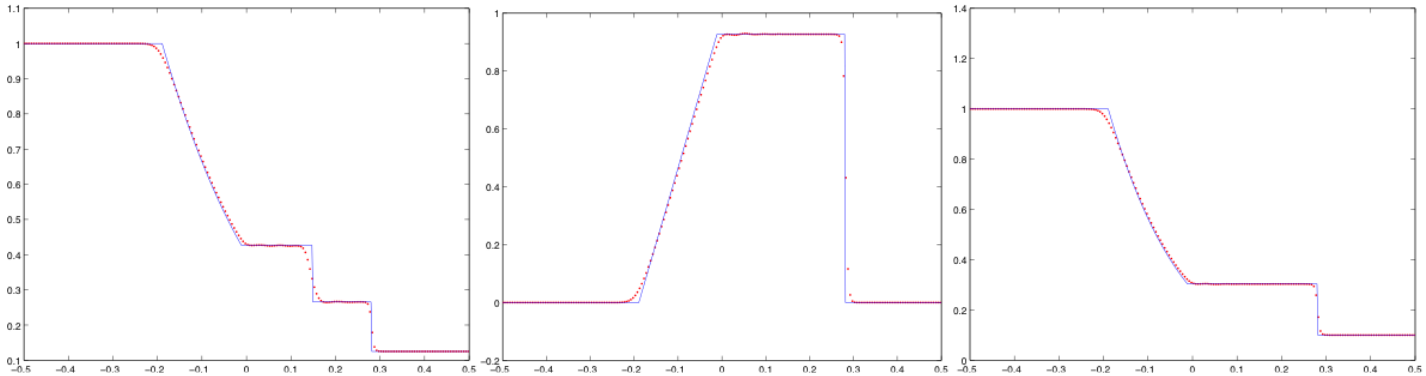
$$\begin{aligned} u_j^{n+1} &= u_j^n - \frac{\Delta t}{\Delta x} \sum_{l=1}^{\nu} b_l K_l \\ K_l &= f_{j+\frac{1}{2}}^{(l)} - f_{j-\frac{1}{2}}^{(l)} \end{aligned}$$

# Application to gas dynamics

Classical Sod problem solved using primitive variables as predictor



WENO 2-3



WENO 3-5

# Stability analysis: a (somewhat) negative result

Consider linear convective equation

$$u_t + (vu)_x = 0,$$

Evolve by conservative FD scheme:

$$\frac{du_j}{dt} = -\frac{1}{\Delta x}\left(\hat{f}_{j+\frac{1}{2}} - \hat{f}_{j-\frac{1}{2}}\right),$$

The numerical solution is computed as

$$u_j^{n+1} = u_j^n - \frac{\Delta t}{\Delta x}\sum_{\ell=1}^{s} b_\ell \left(\hat{f}^{(\ell)}_{j+\frac{1}{2}} - \hat{f}^{(\ell)}_{j-\frac{1}{2}}\right).$$

$$u_j^{(\ell)} = u^n(x_j^{(\ell)}), \quad x_j^{(\ell)} = x_j - vc_\ell\Delta t$$

Look for Fourier modes

$$u_j^n[\xi] = \rho^n e^{ij\xi},$$

# Stability analysis

Use Fourier interpolation for arbitrary $x$

$$u^n(x) = \rho^n e^{ix\xi/\Delta x}.$$

Compute the non conservative semilagrangian stages

$$u_j^{(\ell)} = u^n(x_j^{(\ell)}) = \rho^n \exp(i\xi(x_j - v\Delta t c_\ell)/\Delta x) = \rho^n e^{ij\xi} e^{-ic_\ell a\xi},$$

Where $a = v\Delta t/\Delta x$ (Courant number). From this obtain the amplification factor

$$\rho = 1 - i\xi a \sum_{\ell=1}^{s} b_\ell \exp(-ic_\ell a\xi).$$

# Analogy with A-stability

Test equation for A-stability

$$w'(t) = \lambda w(t), \qquad w(0) = 1$$

Exact solution

$$w(\Delta t) = e^{\lambda \Delta t} = e^z,$$

Identity obtained observing that $\int_0^1 e^{cz}\, dc = (e^z - 1)/z$:

$$e^z = 1 + z \int_0^1 e^{cz}\, dc.$$

Using exact Fourier interpolation, the error is due to the use of quadrature rule to compute the integral:

$$R(z) = 1 + z \sum_{\ell=1}^{s} b_\ell e^{c_\ell z} \neq e^z \qquad \left[ \rho = 1 - i\xi a \sum_{\ell=1}^{s} b_\ell \exp(-i c_\ell a \xi) \right].$$

Comparison with the expression for $\rho$ gives $\rho = R(-i\xi a)$.
It turns out that $|\rho| < 1$ cannot be satisfied unconditionally on $a$.

Optimal quadrature formulas: given $s$ stages, choose the scheme of order $s$ with the largest stability region.

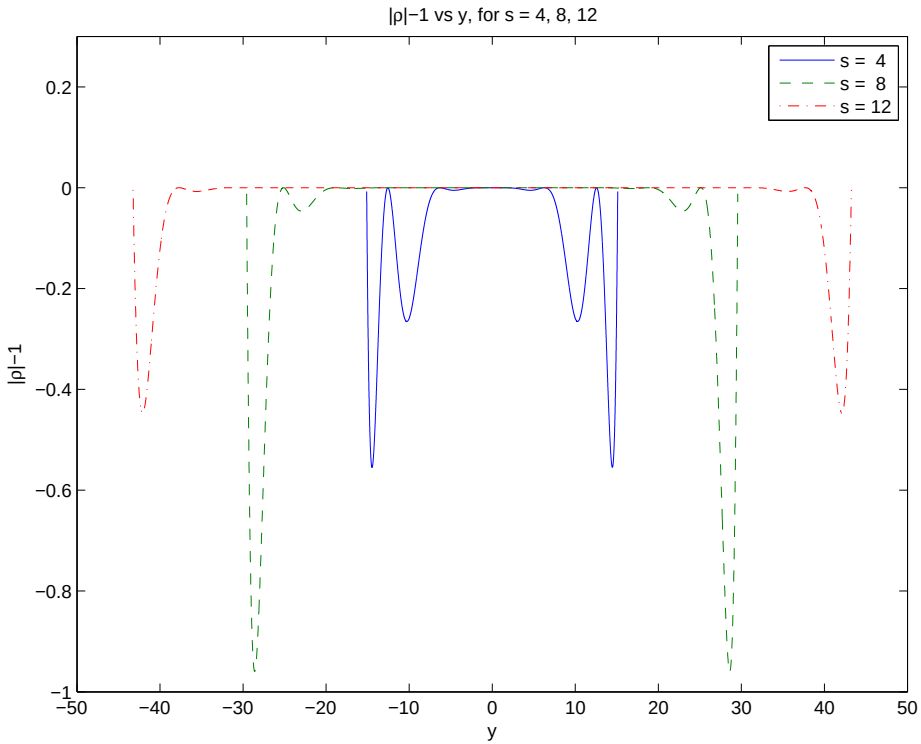
Optimal quadrature formulas: given $s$ stages, choose the scheme of order $s$ with the largest stability region. Simple optimal formulas achieve:

| |
|---|
| $s = 4,\ a^* = 4.81$ |
| $s = 8,\ a^* = 9.41$ |
| $s = 12,\ a^* = 13.77$ |

Optimal quadrature formulas: given $s$ stages, choose the scheme of order $s$ with the largest stability region. Simple optimal formulas achieve:

| |
|---|
| $s = 4,\ a^* = 4.81$ |
| $s = 8,\ a^* = 9.41$ |
| $s = 12,\ a^* = 13.77$ |

The function $|\rho| - 1$ is given by

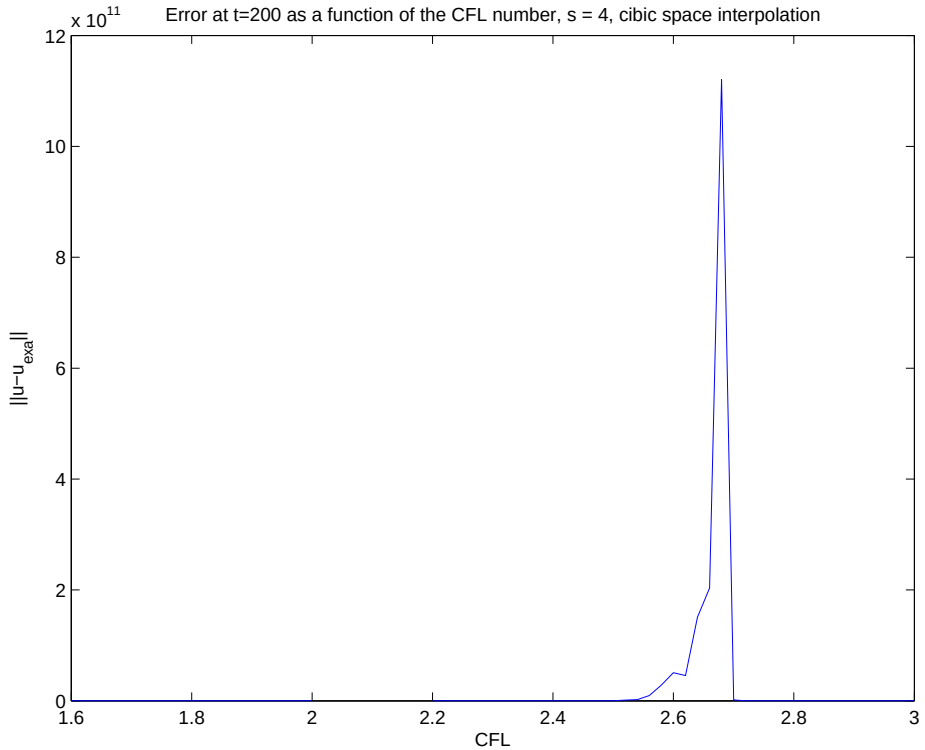


# The bad news

Numerical codes for the single scalar equation with such schemes show instabilities for some CFL numbers much smaller than $a^*$.

# The bad news

Numerical codes for the single scalar equation with such schemes show instabilities for some CFL numbers much smaller than $a^*$.
For example: using a third degree polynomial (4th order space interpolation) rather than Fourier interpolation, $s = 4$, one obtains instability in a neighborhood of $a = 2.6$ (instead of 4.81).



The instability disappears for larger values of $a$, up to about the theoretical value $a^* = 4.81$.

The instability disappears for larger values of $a$, up to about the theoretical value $a^* = 4.81$.
Possible solutions (work in progress):

- check stability of different stencils and force nonlinear reconstruction to choose the stable stencil
- reformulate the whole stability analysis replacing Fourier interpolation by polynomial interpolations

The instability disappears for larger values of $a$, up to about the theoretical value $a^* = 4.81$.
Possible solutions (work in progress):

- check stability of different stencils and force nonlinear reconstruction to choose the stable stencil
- reformulate the whole stability analysis replacing Fourier interpolation by polynomial interpolations
- Basic question: is this of any use, or there is no way to make such conservative correction stable in practice?

# Conclusions

# Conclusions

- Semilagrangian schemes are very promising for kinetic equations because:

# Conclusions

- Semilagrangian schemes are very promising for kinetic equations because:
- Non conservative schemes allow high order accuracy and large CFL numbers

# Conclusions

- Semilagrangian schemes are very promising for kinetic equations because:
- Non conservative schemes allow high order accuracy and large CFL numbers
- For BGK high accuracy in space and time can be reached, using RK or BDF.

# Conclusions

- Semilagrangian schemes are very promising for kinetic equations because:
- Non conservative schemes allow high order accuracy and large CFL numbers
- For BGK high accuracy in space and time can be reached, using RK or BDF.
- RK are more accurate, but BDF appear are more efficient.

# Conclusions

- Semilagrangian schemes are very promising for kinetic equations because:
- Non conservative schemes allow high order accuracy and large CFL numbers
- For BGK high accuracy in space and time can be reached, using RK or BDF.
- RK are more accurate, but BDF appear are more efficient.
- No formal stability restriction on CFL. In practice CFL is restricted by accuracy.

# Conclusions

- Semilagrangian schemes are very promising for kinetic equations because:
- Non conservative schemes allow high order accuracy and large CFL numbers
- For BGK high accuracy in space and time can be reached, using RK or BDF.
- RK are more accurate, but BDF appear are more efficient.
- No formal stability restriction on CFL. In practice CFL is restricted by accuracy.
- Schemes without interpolation may be more efficient (in some cases).

# Conclusions

- Semilagrangian schemes are very promising for kinetic equations because:
- Non conservative schemes allow high order accuracy and large CFL numbers
- For BGK high accuracy in space and time can be reached, using RK or BDF.
- RK are more accurate, but BDF appear are more efficient.
- No formal stability restriction on CFL. In practice CFL is restricted by accuracy.
- Schemes without interpolation may be more efficient (in some cases).
- A general technique is proposed to construct a conservative scheme starting from a non conservative one.

# Conclusions

- Semilagrangian schemes are very promising for kinetic equations because:
- Non conservative schemes allow high order accuracy and large CFL numbers
- For BGK high accuracy in space and time can be reached, using RK or BDF.
- RK are more accurate, but BDF appear are more efficient.
- No formal stability restriction on CFL. In practice CFL is restricted by accuracy.
- Schemes without interpolation may be more efficient (in some cases).
- A general technique is proposed to construct a conservative scheme starting from a non conservative one.
- Deeper analysis is needed to understand and improve the stability property of such conservative correction.

# Conclusions

- Semilagrangian schemes are very promising for kinetic equations because:
- Non conservative schemes allow high order accuracy and large CFL numbers
- For BGK high accuracy in space and time can be reached, using RK or BDF.
- RK are more accurate, but BDF appear are more efficient.
- No formal stability restriction on CFL. In practice CFL is restricted by accuracy.
- Schemes without interpolation may be more efficient (in some cases).
- A general technique is proposed to construct a conservative scheme starting from a non conservative one.
- Deeper analysis is needed to understand and improve the stability property of such conservative correction.

Thank you !


Happy Birthday to Prof. Sone and Prof. Aoki!
Costa
Pasticceria