

Погружение в астрофизические течения – вводный практикум по вычислительной гидродинамике

Кондратьев Илья

ИКИ РАН

II школа ЭЛАГ

Саров – НЦФМ, 04 июля 2024

Содержание

1. Описание используемого на занятии кода Piastra2D.
2. Различные модели сплошных сред как законы сохранения.
3. Метод конечного объема.
4. Линейное уравнение переноса как простейшая модель сплошной среды.
5. Программная реализация вычислительных кодов.
6. Разные способы решения уравнения переноса. Устойчивость. Апроксимация. Методы высокого порядка и монотонность.
7. Уравнения газовой динамики. Методы Годуновского типа.
8. Магнитная гидродинамика.

Рекомендуемая литература

1. M. Zingale “Introduction to Computational Astrophysical Hydrodynamics”, 2015+ (http://open-astrophysics-bookshelf.github.io/numerical_exercises/CompHydroTutorial.pdf)
2. Д.В. Бисикало, А.Г. Жилкин, А.А. Боярчук «Газодинамика тесных двойных звезд» — М.: Физматлит, 2013
3. А.Г. Куликовский, Н.В. Погорелов, А.Ю. Семенов «Математические вопросы численного решения гиперболических систем уравнений» — М.: Физматлит, 2001
4. E.F. Toro “Riemann solvers and numerical methods for fluid dynamics – Springer, Berlin, 2009
5. D.S. Balsara “Higher-order accurate space-time schemes for computational astrophysics—Part I: finite volume methods”, Living Rev Comput Astrophys 3:2 (2017), <https://doi.org/10.1007/s41115-017-0002-8>
6. G. Toth “The $\nabla \cdot \mathbf{B} = 0$ Constraint in Shock-Capturing Magnetohydrodynamics Codes”, J Comput Phys, 161, 605–652 (2000) <https://doi.org/10.1006/jcph.2000.6519>
7. A. Mignone “High-order conservative reconstruction schemes for finite volume methods in cylindrical and spherical coordinates”, J Comput Phys 270:784–814 (2014) <https://doi.org/10.1016/j.jcp.2014.04.001>
8. J.M. Martí, E. Müller “Grid-based methods in relativistic hydrodynamics and magnetohydrodynamics”, Living Rev Comput Astrophys 1:3 (2015) <https://link.springer.com/article/10.1007/lrca-2015-3>
9. A. Mignone et al “PLUTO: A Numerical Code for Computational Astrophysics”, Astrophys J Suppl, 170, 228 (2007) <https://iopscience.iop.org/article/10.1086/513316>

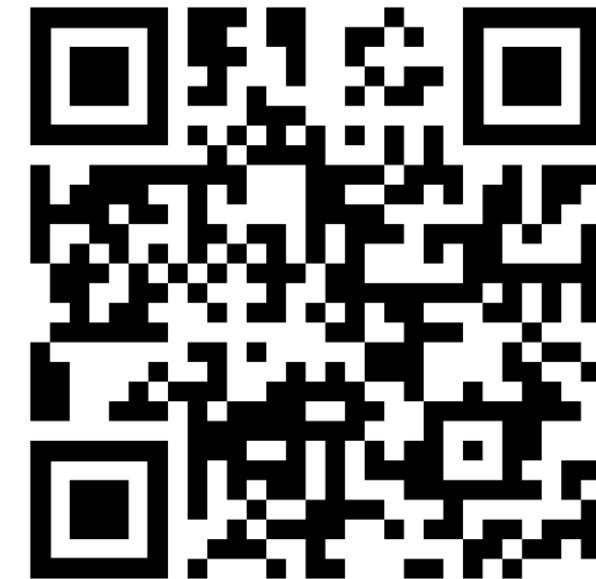
Piastra2D – учебный код для сжимаемых течений в 2D

- Архив можно загрузить по ссылке
<https://github.com/mrkondratyev/Piastra2D>
- Для работы необходимо предварительно установить **Python** с библиотеками **NumPy**, **matplotlib** (обычно эти библиотеки установлены по умолчанию) и **IPython** + **jupyter notebook** (всё можно установить при помощи anaconda или pip, я рекомендую anaconda)
- Запускать модели можно с помощью ipynb-файлов
“*_solver_easy_run.ipynb”**



Piastra2D – учебный код для сжимаемых течений в 2D

- Код «умеет» решать уравнения линейного переноса, газовой динамики и магнитной гидродинамики в 1D и 2D в декартовой геометрии (см. ниже на данном занятии)
- Имеет все основные детали, которые есть в «настоящих» больших комплексах программ для моделирования плазменных/астрофизических течений
- Для тестирования кода можно запустить файл **fluid_solver_easy_run.ipynb** с любой из тестовых моделей
- На выходе получится анимация + в конце будет сообщение об успешном завершении



Использование готовых пакетов программ

Очень много МГД-кодов с учетом разной дополнительной физики:

1. **PLUTO** (<https://plutocode.ph.unito.it/>)
2. **Athena++** (<https://www.athena-astro.app/>)
3. **FLASH** (<https://flash.rochester.edu/site/flashcode/>)
4. + many-many more!

- «+»: Не нужно писать/отлаживать код (**однако всегда нужно знать, как работает используемый вами метод!!!**); крупные пакеты уже подготовлены для использования на многопроцессорных системах
- «-»: Функционала под нужную именно вам задачу может не быть; в крупных пакетах может быть сложно разобраться; проблема “черного ящика”

Гидродинамический закон сохранения

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot \mathbf{F} = \mathbf{R}$$

- Изменение сохраняющейся величины внутри ограниченного объема во времени равно потоку этой величины через границы ячеек

$$\frac{\partial \int_V \mathbf{U} dV}{\partial t} + \oint_{\partial V} \mathbf{F} \cdot d\mathbf{S} = \int_V \mathbf{R} dV$$

Большинство кодов решает уравнения сплошной среды в такой форме (Методы конечного объема / Методы типа Годунова)

Различные модели сплошных сред как законы сохранения

- Линейное уравнение переноса (адвекция) – простейшая **линейная** модель сплошной среды – начальный профиль переносится в пространстве с постоянной скоростью

$$\frac{\partial U}{\partial t} + a_x \frac{\partial U}{\partial x} + a_y \frac{\partial U}{\partial y} = 0$$

- Уравнение Бюргерса (Хопфа) – простейшая **нелинейная** модель сплошной среды – возможны аналоги волн разрежения и ударных волн

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} = 0 \quad \rightarrow \quad \frac{\partial u}{\partial t} + \frac{\partial}{\partial x} \left(\frac{u^2}{2} \right) = 0$$

Уравнения газовой динамики

- Стандартная форма уравнений (Ландау & Лифшиц, том 6):

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0$$

- ρ - плотность,
- \mathbf{v} - вектор скорости,
- p - давление,
- ϵ - внутренняя энергия на единицу массы,

$$\rho \left(\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla) \mathbf{v} \right) = -\nabla p$$

$$\frac{\partial \epsilon}{\partial t} + \mathbf{v} \cdot \nabla \epsilon = -\frac{p}{\rho} \nabla \cdot \mathbf{v}$$

+ нач и гран условия
+ уравнение состояния

**Перепишем их в форме
законов сохранения**

Различные модели сплошных сред как законы сохранения – Газовая динамика

1. Уравнение сохранения массы (непрерывности):

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho v_x)}{\partial x} + \frac{\partial(\rho v_y)}{\partial y} = 0 \quad (1)$$

2. Уравнение сохранения импульса в направлении x :

$$\frac{\partial(\rho v_x)}{\partial t} + \frac{\partial(\rho v_x^2 + p)}{\partial x} + \frac{\partial(\rho v_x v_y)}{\partial y} = 0 \quad (2)$$

3. Уравнение сохранения импульса в направлении y :

$$\frac{\partial(\rho v_y)}{\partial t} + \frac{\partial(\rho v_x v_y)}{\partial x} + \frac{\partial(\rho v_y^2 + p)}{\partial y} = 0 \quad (3)$$

4. Уравнение сохранения энергии:

$$\frac{\partial E}{\partial t} + \frac{\partial((E + p)v_x)}{\partial x} + \frac{\partial((E + p)v_y)}{\partial y} = 0 \quad (4)$$

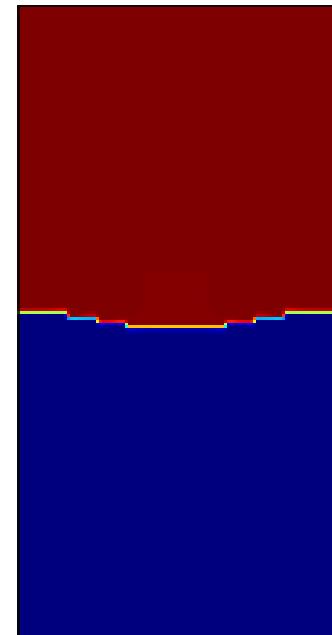
где ρ - плотность, v и v - компоненты скорости в направлениях x и y соответственно, p - давление, E - полная энергия на единицу объёма, связанная с плотностью, скоростью и внутренней энергией следующим образом:

$$E = \rho e + \frac{1}{2} \rho(v_x^2 + v_y^2) \quad (5)$$

где e - внутренняя энергия на единицу массы.

- Газовая динамика – описывает процессы в сжимаемых жидкостях/газах и столкновительной плазме без магнитного поля
- Огромное количество процессов! Взрывы, турбулентность, плавучесть и т.д.

Рис.:
Моделирование
неустойчивости
Рэлея-Тейлора,
Fluid_solver_easy
_run.ipynb
С параметрами
«RTI», **Nx = 200**,
Ny = 100



Уравнения магнитной гидродинамики

- Стандартная форма уравнений (Ландау & Лифшиц, том 8):

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0$$

$$\rho \left(\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla) \mathbf{v} \right) = -\nabla p + \mathbf{J} \times \mathbf{B} + \mathbf{f}$$

$$\frac{\partial \mathbf{B}}{\partial t} = \nabla \times (\mathbf{v} \times \mathbf{B})$$

$$\frac{\partial s}{\partial t} + \mathbf{v} \cdot \nabla s = 0$$

$$\mathbf{J} = \nabla \times \mathbf{B}$$

- Магнитная гидродинамика (МГД) – еще больше процессов!
- Аккреционные диски, некоторые виды сверхновых, солнечно-земные связи, лабораторные и астрофизические джеты, термояд, плазменные ускорители и много-много других объектов

**Перепишем их в форме
законов сохранения**

+ нач и гранич условия
+ уравнение состояния

Различные модели сплошных сред как законы сохранения – магнитная гидродинамика

1. Уравнение сохранения массы:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0 \quad (1)$$

2. Уравнение сохранения импульса:

$$\frac{\partial(\rho \mathbf{v})}{\partial t} + \nabla \cdot \left[\rho \mathbf{v} \mathbf{v} + \left(p + \frac{\mathbf{B}^2}{8\pi} \right) \mathbf{I} - \frac{\mathbf{B} \mathbf{B}}{4\pi} \right] = 0 \quad (2)$$

3. Уравнение сохранения энергии:

$$\frac{\partial E}{\partial t} + \nabla \cdot \left[\left(E + p + \frac{\mathbf{B}^2}{8\pi} \right) \mathbf{v} - \frac{(\mathbf{v} \cdot \mathbf{B}) \mathbf{B}}{4\pi} \right] = 0 \quad (3)$$

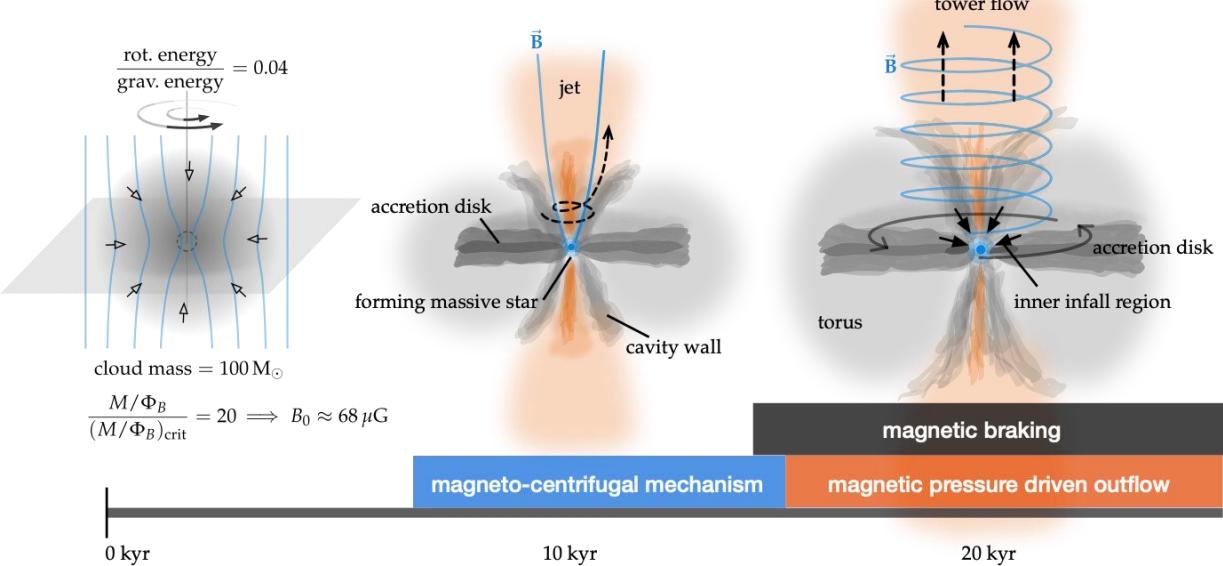
4. Уравнение индукции магнитного поля:

$$\frac{\partial \mathbf{B}}{\partial t} + \nabla \cdot (\mathbf{v} \mathbf{B} - \mathbf{B} \mathbf{v}) = 0 \quad (4)$$

где ρ - плотность, \mathbf{v} - вектор скорости, p - давление, \mathbf{B} - магнитное поле, E - полная энергия, включающая внутреннюю энергию ϵ , кинетическую энергию и магнитную энергию:

$$E = \rho \epsilon + \frac{1}{2} \rho \mathbf{v}^2 + \frac{\mathbf{B}^2}{8\pi} \quad (5)$$

Рис.: Генерация джетов от протозвезд [REF: <https://www.gandreoliva.org/posters/ringberg23/index.html>]



Другие модели, которые представимы в аналогичной (дивергентной) форме

- Релятивистская газовая динамика (сверхновые, акреция на НЗ и ЧД, АЯГ)
- Релятивистская магнитная гидродинамика (все выше + КГВ)
- Уравнения переноса излучения (и нейтрино!) в двухмоментной форме (поток энергии и поток импульса излучения)
- Двухжидкостная гидродинамика плазмы

Все упомянутые системы уравнений являются **гиперболическими** моделями среды (матрица $A = \frac{\partial F}{\partial U}$ имеет вещественные собственные значения при любых входных данных), и к ним могут быть применены схожие подходы



В гиперболических моделях можно рассмотреть волновые возмущения, движущиеся с конечными скоростями

Дивергентная форма записи уравнений

- В дифференциальной форме

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot \mathbf{F} = \mathbf{R}$$

Изменение во времени объемной плотности сохраняющейся величины равно дивергенции от потока данной величины

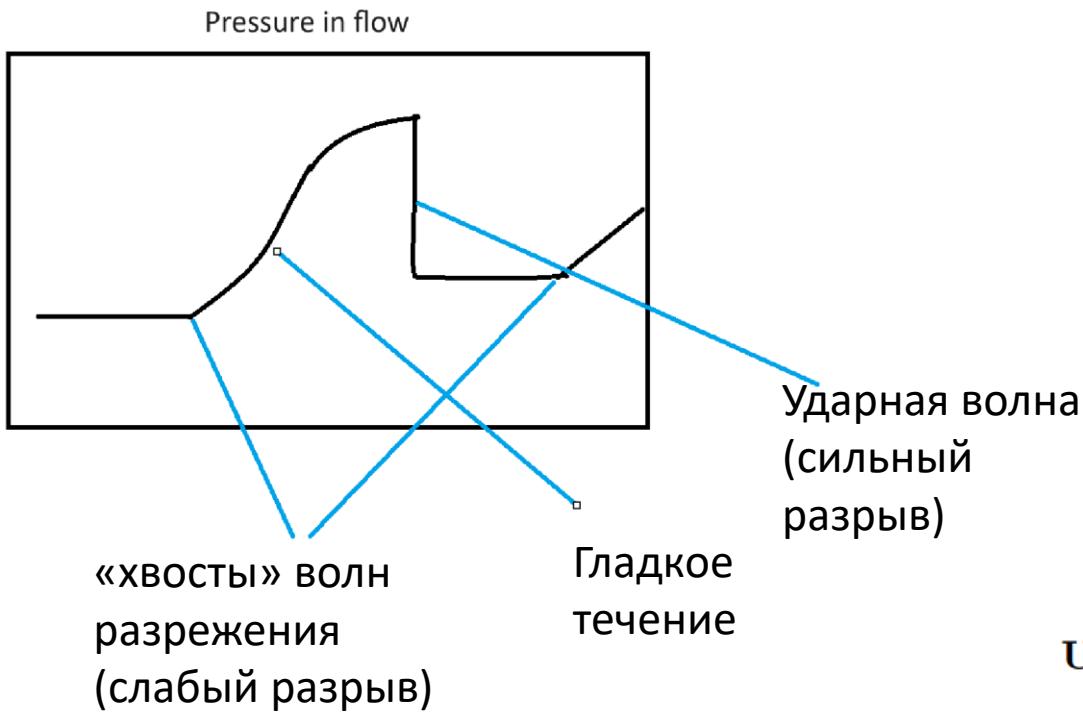
- В интегральной форме (интеграл по объему => изменение самой величины + теорема Остроградского-Гаусса для дивергенции потоков)

$$\frac{\partial \int_V \mathbf{U} dV}{\partial t} + \oint_{\partial V} \mathbf{F} \cdot d\mathbf{S} = \int_V \mathbf{R} dV$$

- «Работает» для любой из рассмотренных систем уравнений

Интегральная форма уравнений лучше дифференциальной?

- Пусть есть течение с разрывами



Дифференциальные уравнения могут не существовать, однако профиль функции – интегрируем! Интегральная форма позволяет описать большее количество решений

$$\mathbf{U} = \begin{pmatrix} \rho \\ \rho v_x \\ \rho v_y \\ E \end{pmatrix}, \quad \mathbf{F}_x = \begin{pmatrix} \rho v_x \\ \rho v_x^2 + p \\ \rho v_x v_y \\ (E + p)v_x \end{pmatrix}, \quad \mathbf{F}_y = \begin{pmatrix} \rho v_y \\ \rho v_x v_y \\ \rho v_y^2 + p \\ (E + p)v_y \end{pmatrix}$$

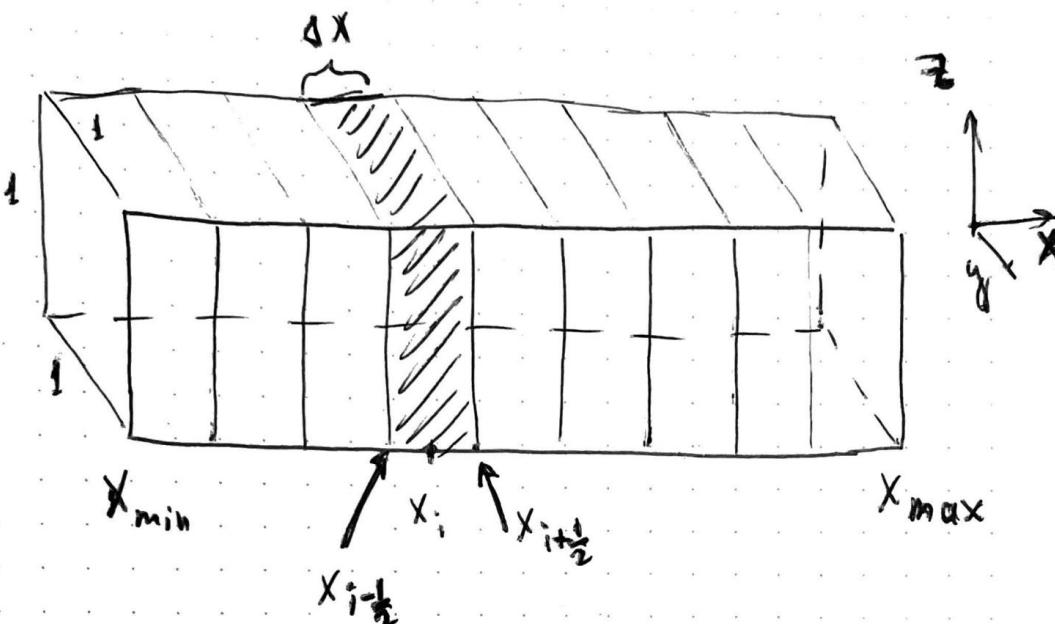
Метод конечного объема (МКО) для уравнений сплошной среды в форме законов сохранения

- Рассмотрим 1D систему законов сохранения в дивергентной форме

$$\frac{\partial U}{\partial t} + \frac{\partial F}{\partial x} = R$$

- Рассмотрим для простоты призму единичного сечения (см. рис.) и введем на ней сетку

$$S_{face} = 1 * 1 - \text{площадь грани}$$
$$dV = 1 * 1 * dx - \text{объем ячейки}$$



Метод конечного объема (МКО) для уравнений сплошной среды в форме законов сохранения

- Проинтегрируем закон сохранения по времени (1 шаг) и «объему» ячейки

$$\int_{t^n}^{t^{n+1}} dt \int_{x_{i-1/2}}^{x_{i+1/2}} \left(\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} - \mathbf{R} \right) dx = \\ \int_{x_{i-1/2}}^{x_{i+1/2}} [\mathbf{U}^{n+1} - \mathbf{U}^n] dx + \int_{t^n}^{t^{n+1}} [\mathbf{F}_{i+1/2} - \mathbf{F}_{i-1/2}] dt - \int_{t^n}^{t^{n+1}} dt \int_{x_{i-1/2}}^{x_{i+1/2}} dx \mathbf{R} = 0$$

$$\bar{\mathbf{U}}_i^n = \frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_{i+1/2}} \mathbf{U}^n dx$$

Введем средние значения для сохраняющихся величин, правой части и потоков

$$\bar{\mathbf{R}}_i^{n+1/2} = \frac{1}{\Delta x} \frac{1}{\Delta t} \int_{t^n}^{t^{n+1}} dt \int_{x_{i-1/2}}^{x_{i+1/2}} \mathbf{R} dx$$

$$\bar{\mathbf{F}}_{i+1/2}^{n+1/2} = \frac{1}{\Delta t} \int_{t^n}^{t^{n+1}} \mathbf{F}_{i+1/2} dt$$

МКО для уравнений сплошной среды в форме законов сохранения

- После подстановки получили следующую аппроксимацию:

$$\frac{\bar{\mathbf{U}}_i^{n+1} - \bar{\mathbf{U}}_i^n}{\Delta t} + \frac{\bar{\mathbf{F}}_{i+1/2}^{n+1/2} - \bar{\mathbf{F}}_{i-1/2}^{n+1/2}}{\Delta x_i} = \bar{\mathbf{R}}_i^{n+1/2}$$

- Умножим на объем ячейки и «проинтегрируем» по всей сетке: $\sum_i f \Delta x_i$:

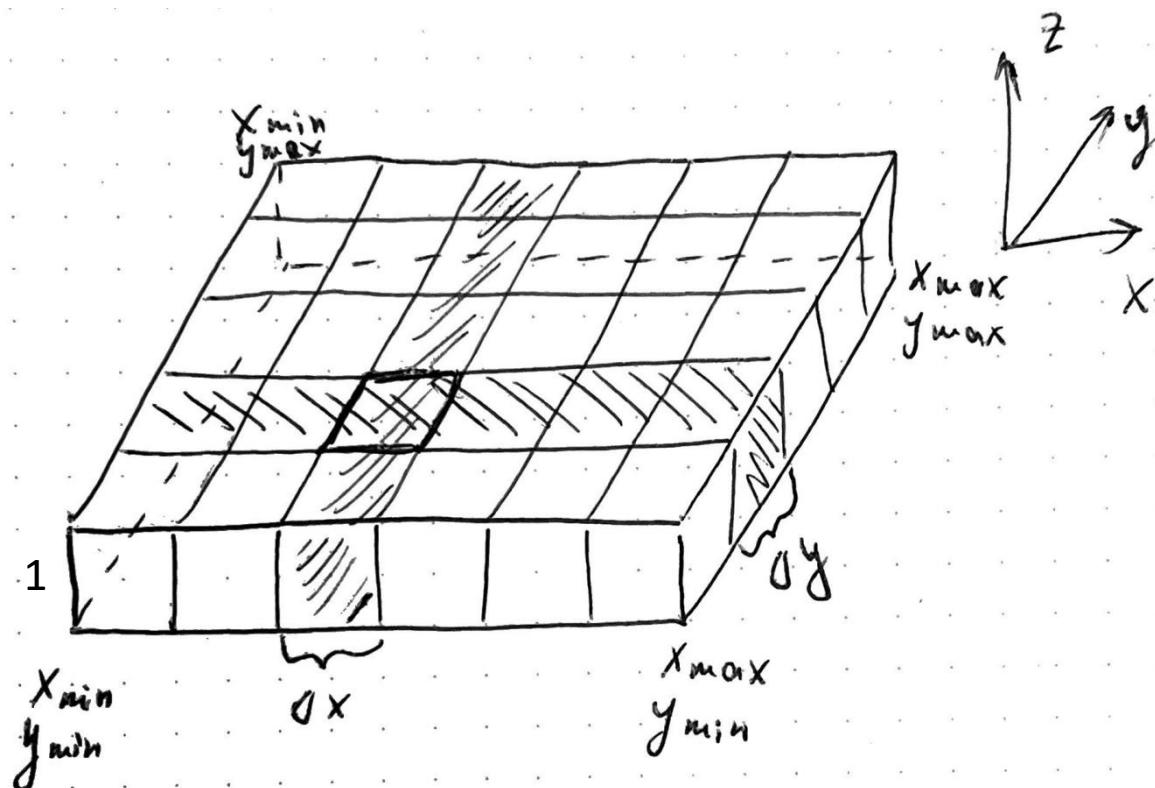
$$\frac{\left(\sum_i \Delta x_i \bar{\mathbf{U}}_i^{n+1} \right) - \left(\sum_i \Delta x_i \bar{\mathbf{U}}_i^n \right)}{\Delta t} + \bar{\mathbf{F}}_R^{n+1/2} - \bar{\mathbf{F}}_L^{n+1/2} - \sum_i \Delta x_i \bar{\mathbf{R}}_i^{n+1/2} = 0$$

Законы сохранения выполняются точно и в дискретном виде! -
консервативность

МКО для уравнений сплошной среды в форме законов сохранения – многомерный случай

- Рассмотрим теперь двумерный случай (2D)

$$\frac{\partial U}{\partial t} + \frac{\partial F_x}{\partial x} + \frac{\partial F_y}{\partial y} = R$$



Наша двумерная сетка –
призма с единичной высотой
вдоль оси z

$S_{face,x} = 1 * \Delta y$ – площадь
грани вдоль оси x

$S_{face,y} = 1 * \Delta x$ – площадь
грани вдоль оси y

$\Delta V = 1 * \Delta y * \Delta x$ – объем
ячейки

МКО для уравнений сплошной среды в форме законов сохранения – многомерный случай

Опять запишем интеграл

$$\int_{t^n}^{t^{n+1}} \int_{x_{i-1/2}}^{x_{i+1/2}} \int_{y_{j-1/2}}^{y_{j+1/2}} \left(\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}_x}{\partial x} + \frac{\partial \mathbf{F}_y}{\partial y} \right) dy dx dt = \int_{t^n}^{t^{n+1}} \int_{x_{i-1/2}}^{x_{i+1/2}} \int_{y_{j-1/2}}^{y_{j+1/2}} \mathbf{R} dy dx dt$$

Используем теорему Гаусса

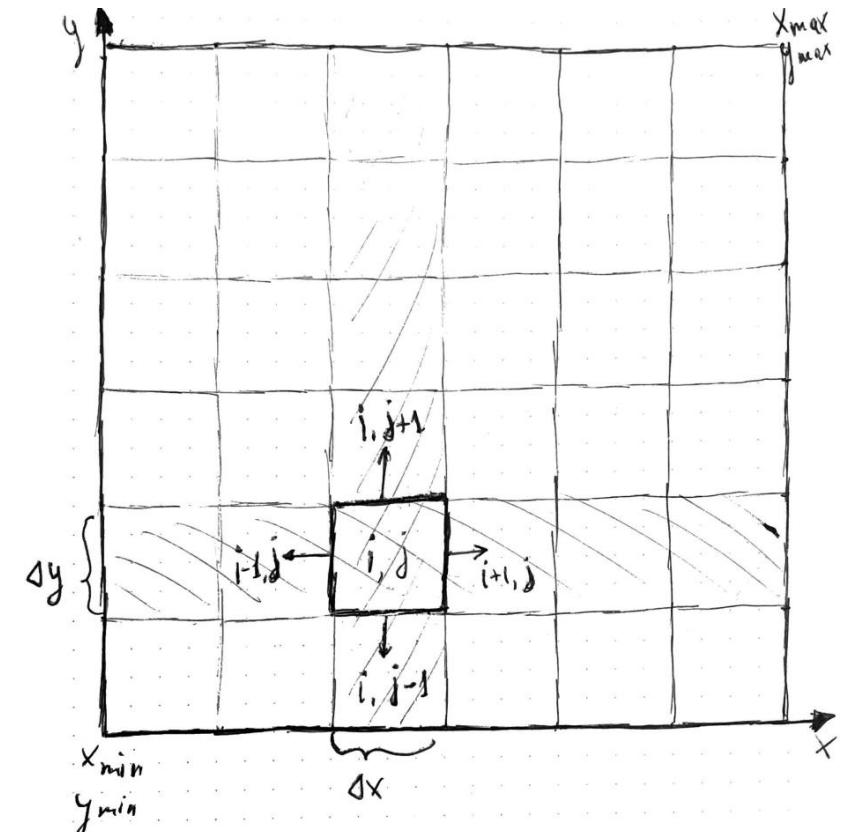
$$\int_{t^n}^{t^{n+1}} \int_{x_{i-1/2}}^{x_{i+1/2}} \int_{y_{j-1/2}}^{y_{j+1/2}} \frac{\partial \mathbf{U}}{\partial t} dy dx dt + \int_{t^n}^{t^{n+1}} \oint_{\partial V_{ij}} (\mathbf{F}_x \hat{n}_x + \mathbf{F}_y \hat{n}_y) dS dt =$$

$$\int_{t^n}^{t^{n+1}} \int_{x_{i-1/2}}^{x_{i+1/2}} \int_{y_{j-1/2}}^{y_{j+1/2}} \mathbf{R} dy dx dt$$

Интегрируем по прямоугольной ячейке

$$\int_{t^n}^{t^{n+1}} \oint_{\partial V_{ij}} (\mathbf{F}_x \hat{n}_x + \mathbf{F}_y \hat{n}_y) dS dt = \int_{t^n}^{t^{n+1}} \left(\int_{y_{j-1/2}}^{y_{j+1/2}} \mathbf{F}_x(x_{i+1/2}, y, t) dy - \int_{y_{j-1/2}}^{y_{j+1/2}} \mathbf{F}_x(x_{i-1/2}, y, t) dy \right) dt$$

$$+ \int_{t^n}^{t^{n+1}} \left(\int_{x_{i-1/2}}^{x_{i+1/2}} \mathbf{F}_y(x, y_{j+1/2}, t) dx - \int_{x_{i-1/2}}^{x_{i+1/2}} \mathbf{F}_y(x, y_{j-1/2}, t) dx \right) dt$$



МКО для уравнений сплошной среды в форме законов сохранения – многомерный случай

Интегрируем по времени

$$\int_{t^n}^{t^{n+1}} \int_{x_{i-1/2}}^{x_{i+1/2}} \int_{y_{j-1/2}}^{y_{j+1/2}} \frac{\partial \mathbf{U}}{\partial t} dy dx dt = \int_{x_{i-1/2}}^{x_{i+1/2}} \int_{y_{j-1/2}}^{y_{j+1/2}} (\mathbf{U}(x, y, t^{n+1}) - \mathbf{U}(x, y, t^n)) dy dx$$

Определим средние значения как и в 1D

$$\bar{\mathbf{U}}_{ij}^n = \frac{1}{\Delta x \Delta y} \int_{x_{i-1/2}}^{x_{i+1/2}} \int_{y_{j-1/2}}^{y_{j+1/2}} \mathbf{U}(x, y, t^n) dy dx$$

$$\bar{\mathbf{F}}_{x,i+1/2,j}^{n+1/2} = \frac{1}{\Delta y \Delta t} \int_{t^n}^{t^{n+1}} \int_{y_{j-1/2}}^{y_{j+1/2}} \mathbf{F}_x(x_{i+1/2}, y, t) dy dt$$

$$\int_{t^n}^{t^{n+1}} \int_{x_{i-1/2}}^{x_{i+1/2}} \int_{y_{j-1/2}}^{y_{j+1/2}} \mathbf{R} dy dx dt = \Delta x \Delta y \Delta t \bar{\mathbf{R}}_{ij}^{n+1/2}$$

МКО для уравнений сплошной среды в форме законов сохранения

- С учетом средних

$$\int_{t^n}^{t^{n+1}} \oint_{\partial V_{ij}} (\mathbf{F}_x \hat{n}_x + \mathbf{F}_y \hat{n}_y) \, dS \, dt = \Delta t \left(\Delta y \left(\bar{\mathbf{F}}_{x,i+1/2,j}^{n+1/2} - \bar{\mathbf{F}}_{x,i-1/2,j}^{n+1/2} \right) + \Delta x \left(\bar{\mathbf{F}}_{y,i,j+1/2}^{n+1/2} - \bar{\mathbf{F}}_{y,i,j-1/2}^{n+1/2} \right) \right)$$

- В итоге получается следующее выражение для МКО в двумерной геометрии

$$\frac{\bar{\mathbf{U}}_{ij}^{n+1} - \bar{\mathbf{U}}_{ij}^n}{\Delta t} + \frac{\bar{\mathbf{F}}_{x,i+1/2,j}^{n+1/2} - \bar{\mathbf{F}}_{x,i-1/2,j}^{n+1/2}}{\Delta x} + \frac{\bar{\mathbf{F}}_{y,i,j+1/2}^{n+1/2} - \bar{\mathbf{F}}_{y,i,j-1/2}^{n+1/2}}{\Delta y} = \bar{\mathbf{R}}_{ij}^{n+1/2}$$

Трехмерный случай (3D) полностью аналогичен 2D, но является существенно более громоздким вычислительно

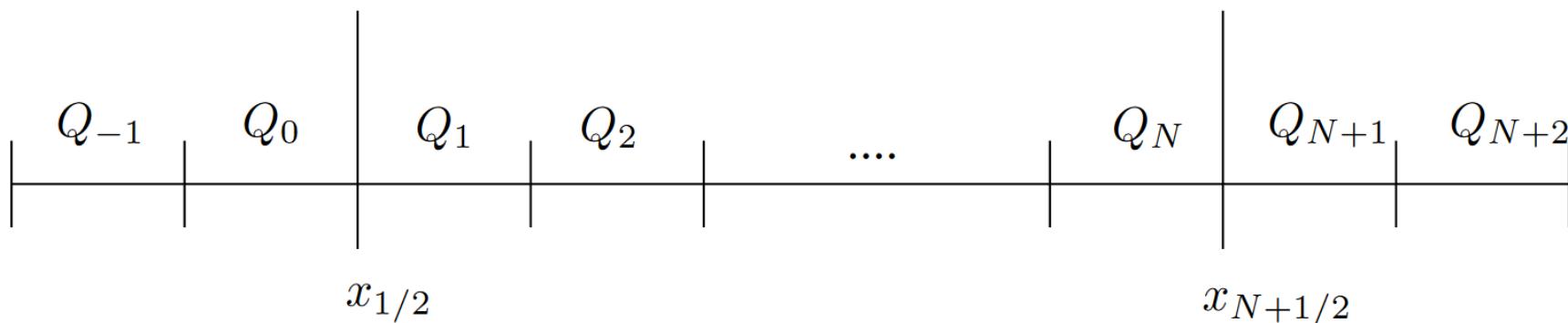
МКО – общий случай

$$\frac{\mathbf{U}_{cell}^{n+1} - \mathbf{U}_{cell}^n}{\Delta t} + \frac{1}{V_{cell}} \sum_{faces \in cell} \mathbf{F}_{face} \cdot \mathbf{n}_{face} S_{face} = \mathbf{R}_{cell}$$

- Работает в произвольной многомерной геометрии
- Все отличия между разными методами/системами лежат в выражении для потока \mathbf{F} и вида нормированной площади/объема
- Декартовые координаты в 3D: $V = \Delta x * \Delta y * \Delta z$, $S_{f,x} = \Delta y * \Delta z$, $S_{f,y} = \Delta x * \Delta z$, $S_{f,z} = \Delta x * \Delta y$
- Цилиндрические координаты в 3D: $V = \frac{r_{i+1/2}^2 - r_{i-1/2}^2}{2} * \Delta z * \Delta \varphi$,
 $S_{f,r_{i+1/2}} = r_{i+1/2} * \Delta z * \Delta \varphi$, $S_{f,z} = \frac{r_{i+1/2}^2 - r_{i-1/2}^2}{2} * \Delta \varphi$, $S_{f,\varphi} = \Delta r * \Delta z$,

Учет граничных условий в МКО – ghost cells

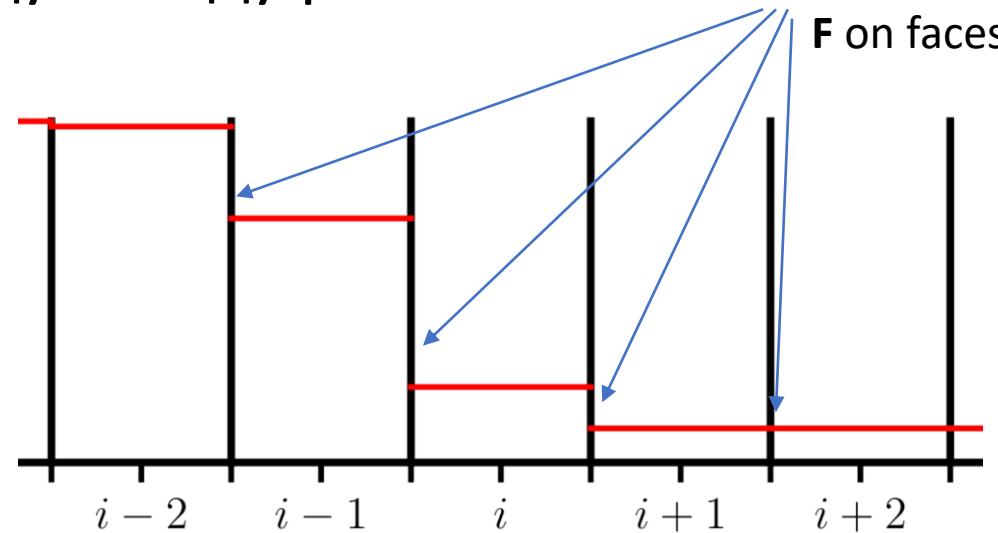
- Подход “ghost cells” – во всех массивах добавляются элементы в фиктивных ячейках за расчетной областью



- Заполняя эти элементы величинами из «реальных» ячеек в соответствии с заданными граничными условиями, можно легко моделировать краевые задачи

МКО для уравнений сплошной среды в форме законов сохранения

- В итоге получаем, что **решение уравнения определено только формой потока** через границу между расчетными ячейками
- Но каким его выбрать?



Рассмотрим пример для уравнения переноса (для него $\mathbf{F} = \mathbf{a}\mathbf{U}$) но сначала обсудим, как вообще устроено его решение, а также как устроены вычислительные коды для моделирования гидродинамических систем

Линейное уравнение переноса как простейшая модель сплошной среды

- Линейное уравнение переноса (адвекции)

$$\frac{\partial U}{\partial t} + a \frac{\partial U}{\partial x} = 0, \quad U(x, 0) = U_0(x)$$

- Характеристики (вдоль них решение распространяется без изменений)

$$\frac{dx}{dt} = a, \quad \zeta = x - at, \quad \frac{dU}{d\zeta} = 0$$

- Получили решение в форме бегущей волны

$$U(x, t) = U_0(x - at)$$

Линейное уравнение переноса как простейшая модель сплошной среды

- Метод Фурье-гармоник:

$$\frac{\partial U}{\partial t} + a \frac{\partial U}{\partial x} = 0, \quad U(x, t) \sim \exp(ik(x - at))$$

- Решение уравнения при наличии диссипации

$$\frac{\partial U}{\partial t} + a \frac{\partial U}{\partial x} = \nu \frac{\partial^2 U}{\partial x^2}, \quad U(x, t) \sim \exp(ik(x - at) - \nu k^2 t)$$

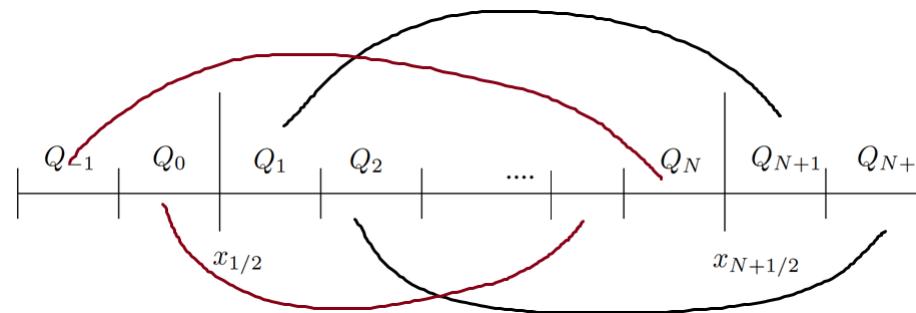
- Решение уравнения при наличии дисперсии

$$\frac{\partial U}{\partial t} + a \frac{\partial U}{\partial x} = \mu \frac{\partial^3 U}{\partial x^3}, \quad U(x, t) \sim \exp(ik(x - at) - i\mu k^3 t)$$

Границные условия для адвекции – ghost cells

- Зададим периодические граничные условия в следующем виде (рис.)
- Такие условия часто используются когда моделируется бесконечная область и/или турбулентные/нерегулярные течения

```
#задаем граничные условия с использованием
#фиктивных ячеек (в данном примере они
#периодические)
for i in range(0,Ngc):
    #внутренняя граница
    U0[i] = U0[Nx+i]
    #внешняя граница
    U0[Nx+Ngc+i] = U0[Ngc+i]
```



Пример. Как устроены гидродинамические коды.

- Файл **Advection_1D.ipynb** – простая 1D линейная модель адвекции

Структура практически любого FV гидродинамического кода:

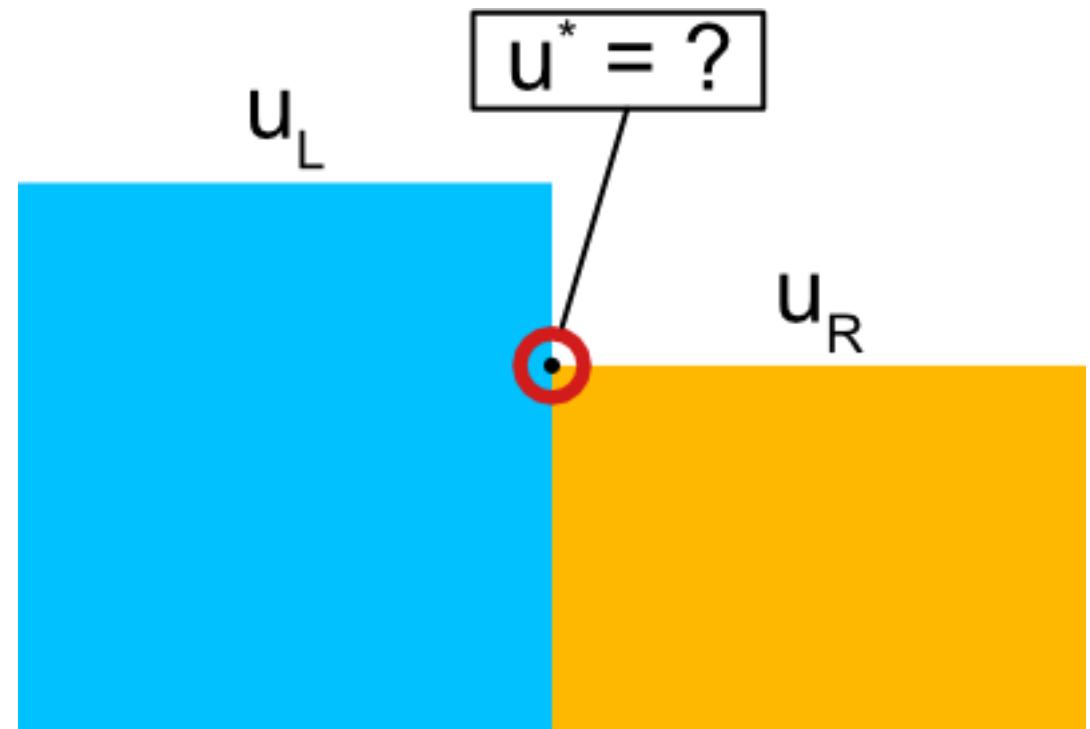
1. Задание расчетной сетки
2. Задание параметров модели и начальных\границых условий
3. Цикл по времени: (пока Time < finalTime)
 1. Расчет допустимого шага во времени (для явных схем - из условия CFL)
 2. Учет граничных условий (через заполнение «фиктивных ячеек»)
 3. Уточнение расчетных величин для расчета в потоках (опционально, порядок точности ++)
 4. Расчет потоков на границах между ячейками
 5. Дополнительные процедуры (опционально) (учет гравитации, удаление дивергенции магнитного поля, учет вязкости/проводимости, etc.....)
 6. Обновление сохраняющихся (консервативных) переменных методом конечного объема
 7. Восстановление физических (примитивных) переменных из сохраняющихся

(некоторые методы (с высоким порядком по времени) – многостадийные и требуют повторения процедур (2)-(7) несколько раз!)

Расчет потока. Задача Римана.

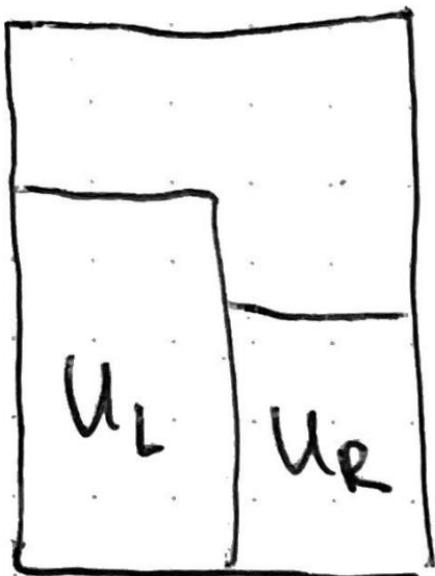
- Поток рассчитывается исходя из физики модели
- Можем трактовать решения справа и слева от границы между ячейками как разрывы в газе (или в другой системе вроде уравнений переноса излучения)
- Из физики моделируемой системы возникает задача Римана
- Является ключевым элементом для гидродинамического кода

Bernhard Riemann (1860): В начальный момент $t = 0$ есть перегородка между двумя независимыми состояниями среды. Перегородку убирают. Что произойдет со средой при $t > 0$?

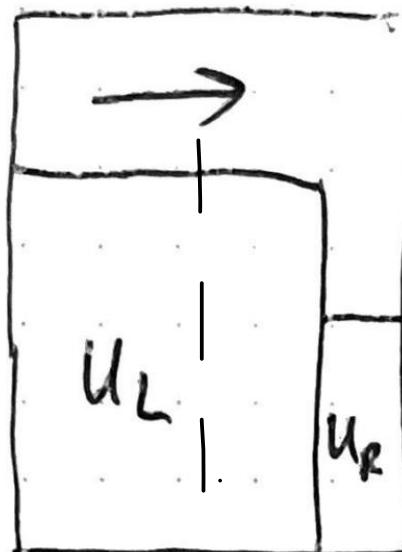


Задача Римана для уравнения адвекции

- Это очень просто!



$$t = 0$$



$$t > 0$$

 $a > 0$



$$t > 0$$

 $a < 0$

Численный поток для уравнения адвекции

- Решение задачи Римана приводит к следующему выражению для потока

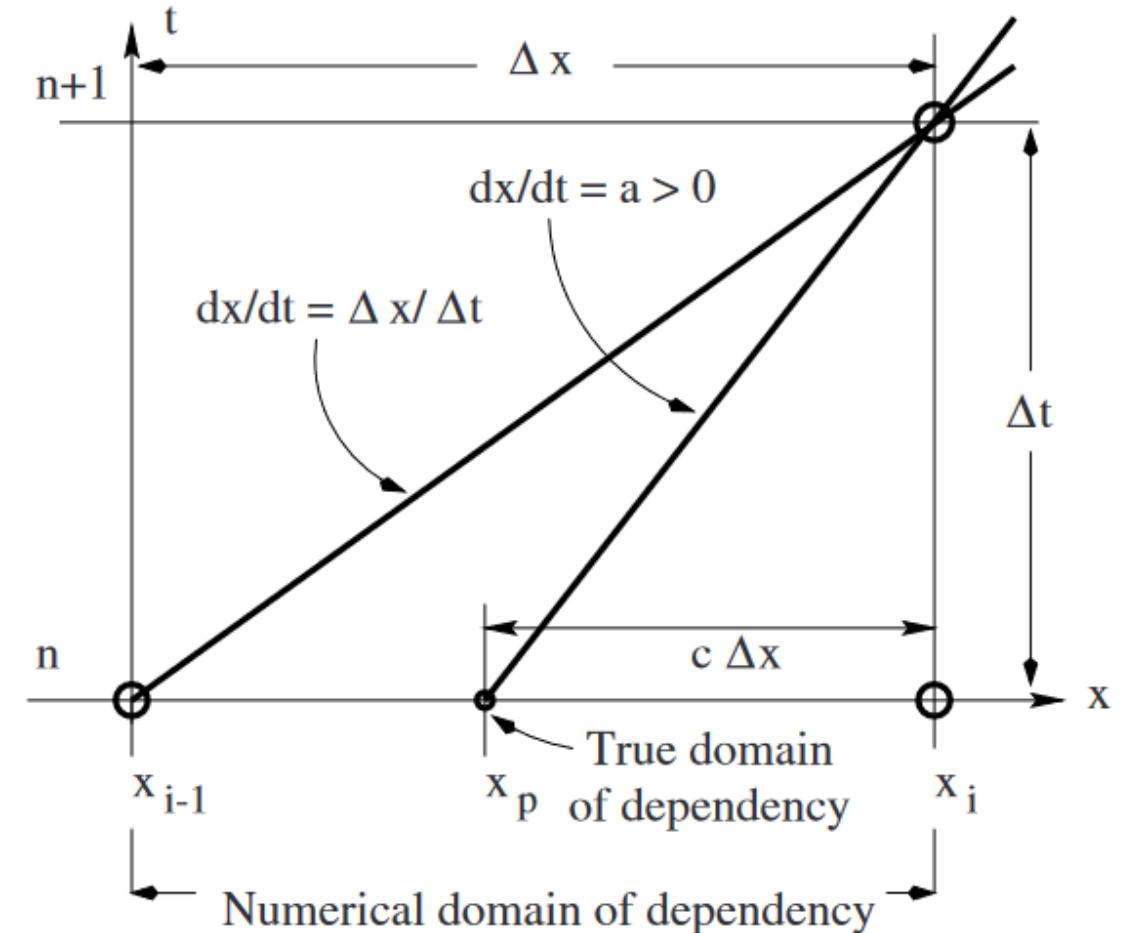
$$F_{i+1/2} = \frac{aU_{i+1} + aU_i}{2} - |a| \frac{U_{i+1} - U_i}{2} \quad \longleftrightarrow \quad \frac{u_i^{n+1} - u_i^n}{\Delta t} = -\frac{a}{\Delta x} \begin{cases} u_i^n - u_{i-1}^n, & a \geq 0, \\ u_{i+1}^n - u_i^n, & a < 0. \end{cases}$$

- Условие устойчивости метода можно трактовать как независимость задач Римана на каждой грани

$$\Delta t = CFL * \min_{i,j} \left(\frac{\Delta x}{|a_x|}, \frac{\Delta y}{|a_y|} \right)$$

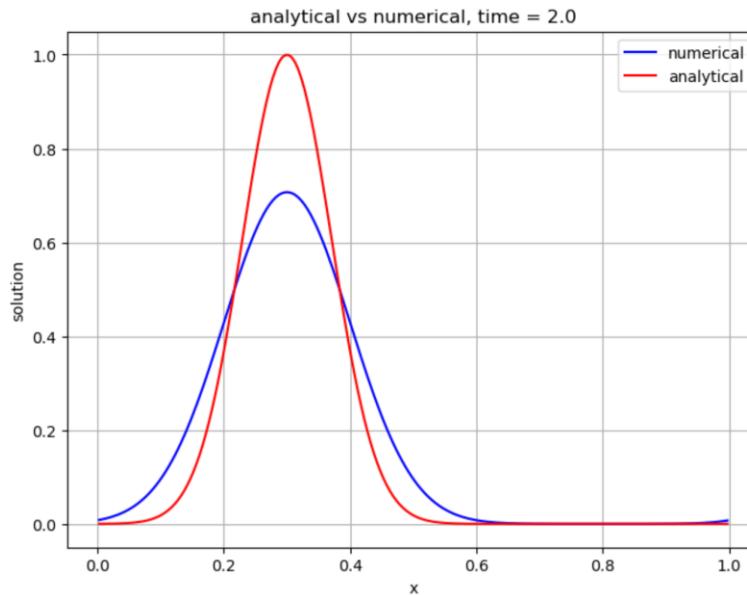
Courant-Friedrichs-Lowy (1928) condition

- Условие CFL является неотъемлемой частью **явных** схем для моделирования гидродинамических течений
- При $CFL > 1$ мы выходим физически разрешимую область влияния модели
- Более «обыденное» определение:
За один шаг по времени возмущение не должно проходить более одной ячейки



Численная диффузия

- Мы получили решение с первым порядком точности
- В рамках такого подхода решение «размазывается» со временем
- Это можно трактовать как решение «реального» уравнения с учетом члена в форме вязкости $\sim \Delta x$

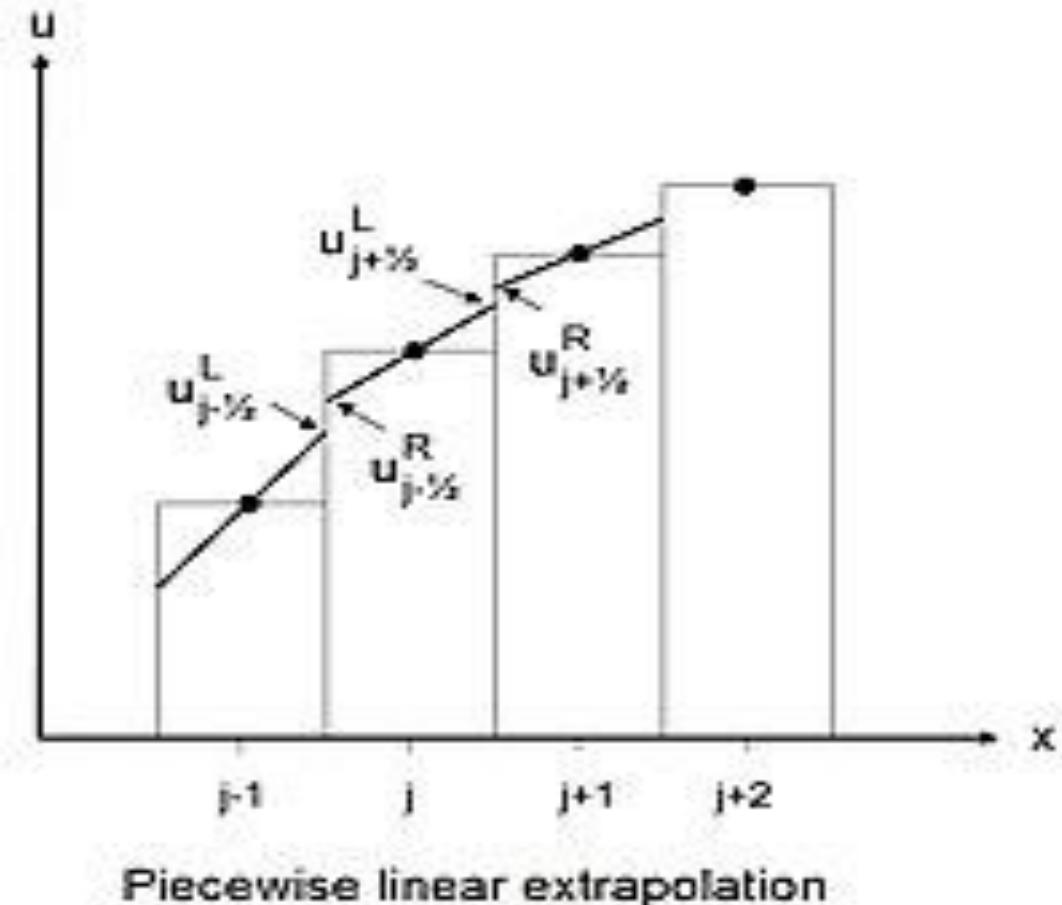


$$\frac{\partial U}{\partial t} + a \frac{\partial U}{\partial x} = \nu \frac{\partial^2 U}{\partial x^2}, \quad U(x, t) \sim \exp(i k(x - at) - \nu k^2 t)$$

- Попробуем построить схему с более высоким порядком точности

Схемы второго порядка точности

- Мы использовали кусочно-постоянное решение для вычисления потока F (**PCM – Piecewise constant method**)
- Попробуем взять линейную реконструкцию с учетом данных о соседних ячейках по времени и по пространству (**PLM**)



Схемы второго порядка точности

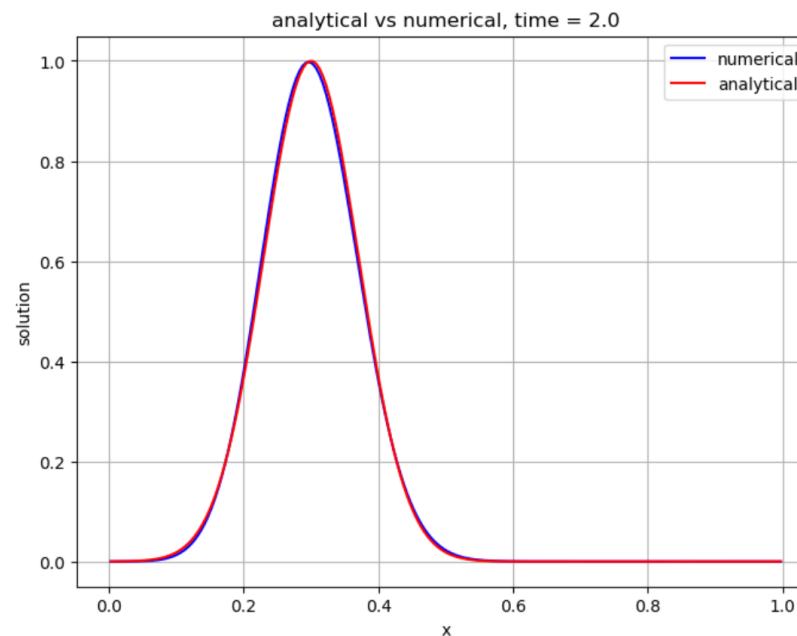
- Уточним величины справа и слева от разрыва:
- $U_L = U_i^n + \frac{\Delta x}{2} \left(\frac{\partial U}{\partial x} \right)_i^n + \frac{\Delta t}{2} \left(\frac{\partial U}{\partial t} \right)_i^n, \quad U_R = U_{i+1}^n - \frac{\Delta x}{2} \left(\frac{\partial U}{\partial x} \right)_{i+1}^n + \frac{\Delta t}{2} \left(\frac{\partial U}{\partial t} \right)_{i+1}^n$
- Используем уравнение для исключения производной по времени
- $U_L = U_i^n + \frac{\Delta x}{2} \left(\frac{\partial U}{\partial x} \right)_i^n - a \frac{\Delta t}{2} \left(\frac{\partial U}{\partial x} \right)_i^n, \quad U_R = U_{i+1}^n - \frac{\Delta x}{2} \left(\frac{\partial U}{\partial x} \right)_{i+1}^n - a \frac{\Delta t}{2} \left(\frac{\partial U}{\partial x} \right)_{i+1}^n$
- Получили $(+O(\Delta t^2 + \Delta x^2))$
- $U_L = U_i^n + \frac{\Delta x - a \Delta t}{2} \left(\frac{\partial U}{\partial x} \right)_i^n; \quad U_R = U_{i+1}^n - \frac{\Delta x + a \Delta t}{2} \left(\frac{\partial U}{\partial x} \right)_{i+1}^n$
- Пусть $\left(\frac{\partial U}{\partial x} \right)_i^n = \left(\frac{\partial U}{\partial x} \right)_{i+1}^n = \frac{U_{i+1}^n - U_i^n}{\Delta x}$ - получим схему Лакса-Вендроффа (1960)

Схема Лакса-Вендроффа (1960)

- Поток схемы LW равен:
- $F_{i+1/2}^{n+1/2} = \frac{aU_L + aU_R}{2} - |a| \frac{U_R - U_L}{2}$
- В исходных обозначениях (см. файл **advection_1D.ipynb**) :
- $F_{i+1/2}^{n+1/2} = \frac{aU_{i+1}^n + aU_i^n}{2} + a^2 \frac{\Delta t}{2\Delta x} (U_i^n - U_{i+1}^n)$
- Данная схема имеет второй порядок точности по времени и пространству

Схема Лакса-Вендроффа для гладкого профиля

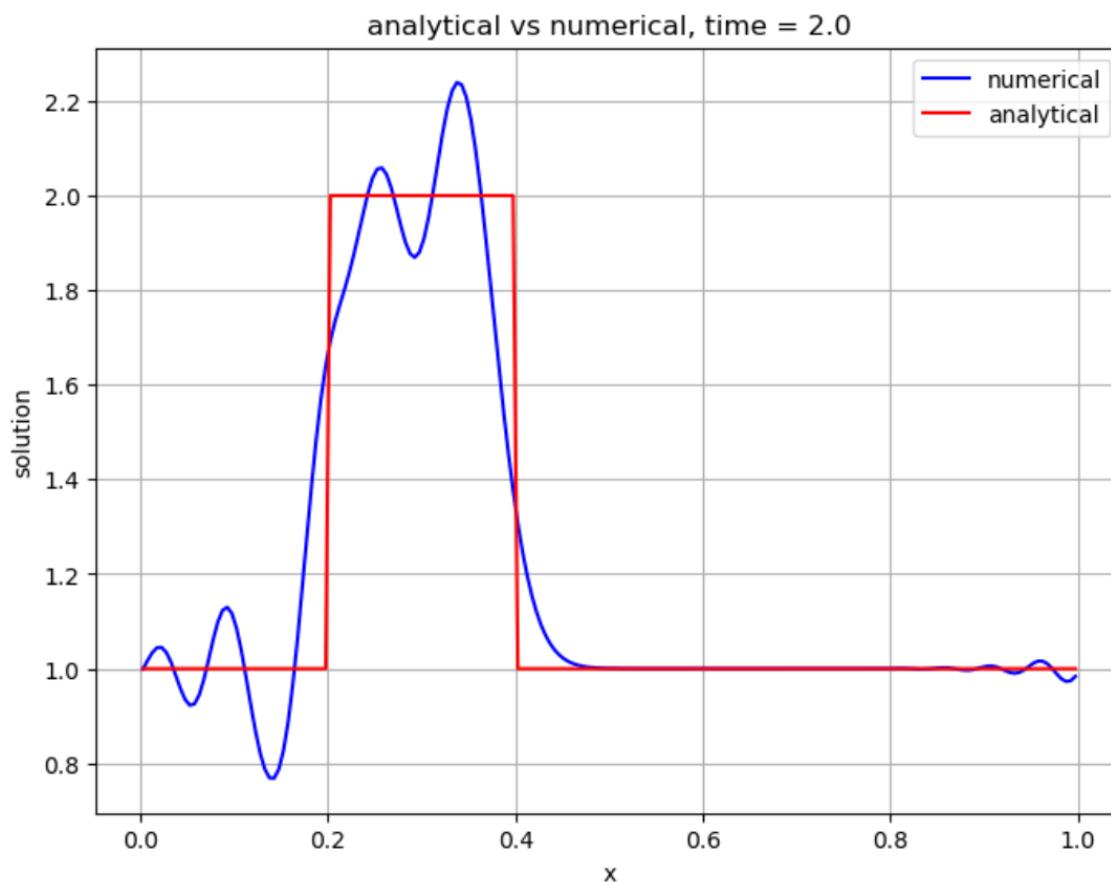
- Можно показать, что такая схема будет иметь меньший уровень численной диссипации (порядок аппроксимации =**2** по времени и пространству вместо =**1** для решения методом задачи Римана)
- Для гладкого решения решение почти идеально совпадает с аналитическим ('smooth')



- Посмотрим, что происходит с разрывным профилем ('disc')

Схема Лакса-Вендроффа для разрывного профиля

- Получили что-то странное...

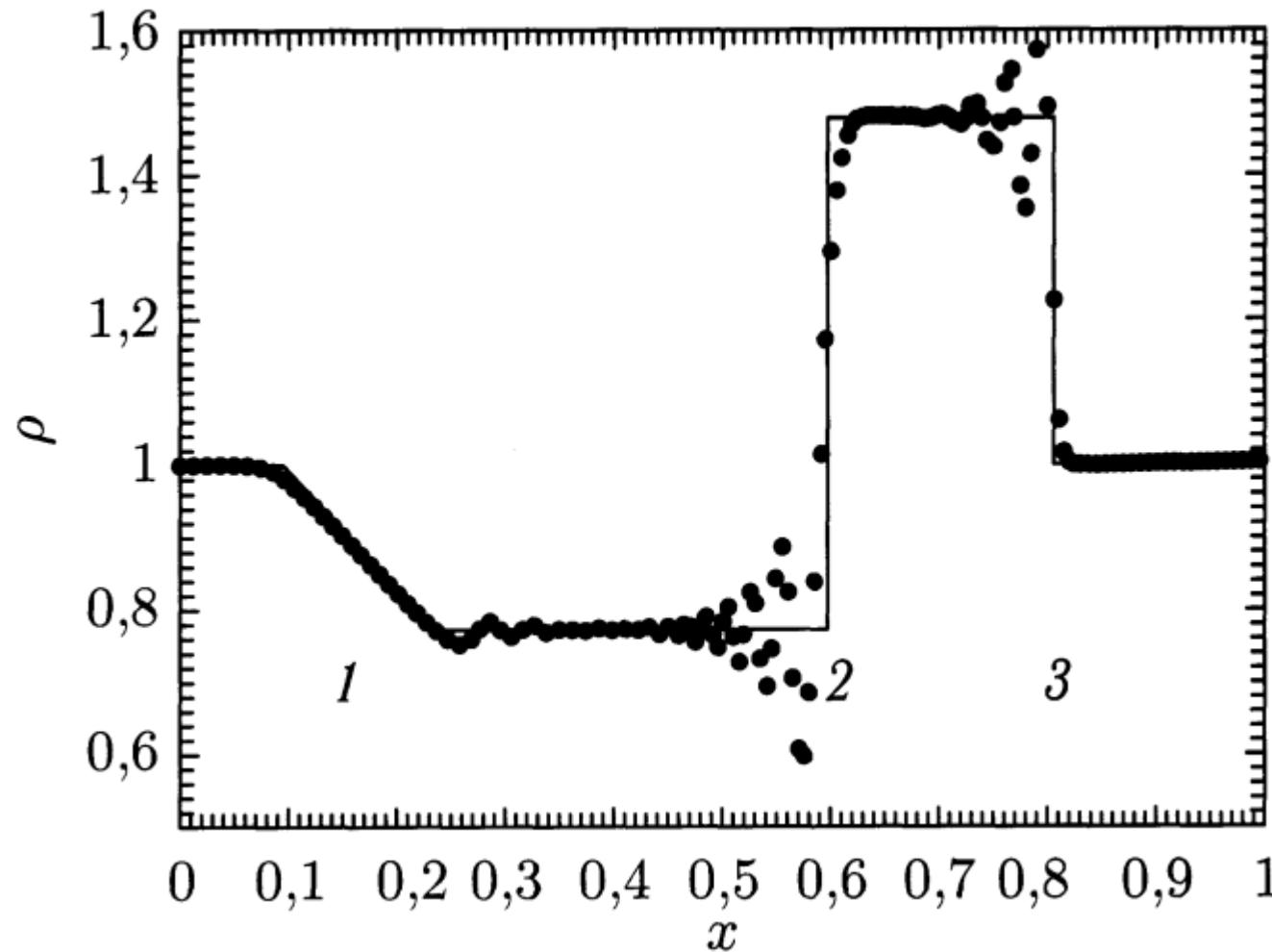


Для коротких волн (разрыв)
имеем дисперсию с
коэффициентом $\sim \Delta x^2$,
т.е. скорость зависит от
длины волны

$$\frac{\partial U}{\partial t} + a \frac{\partial U}{\partial x} = \mu \frac{\partial^3 U}{\partial x^3}, \quad U(x, t) \sim \exp(i k(x - at) - i \mu k^3 t)$$

Получается, что схемы с первым
порядком эффективно решают
уравнения с вязкостью, а со
вторым – с дисперсией

Пример из газовой динамики



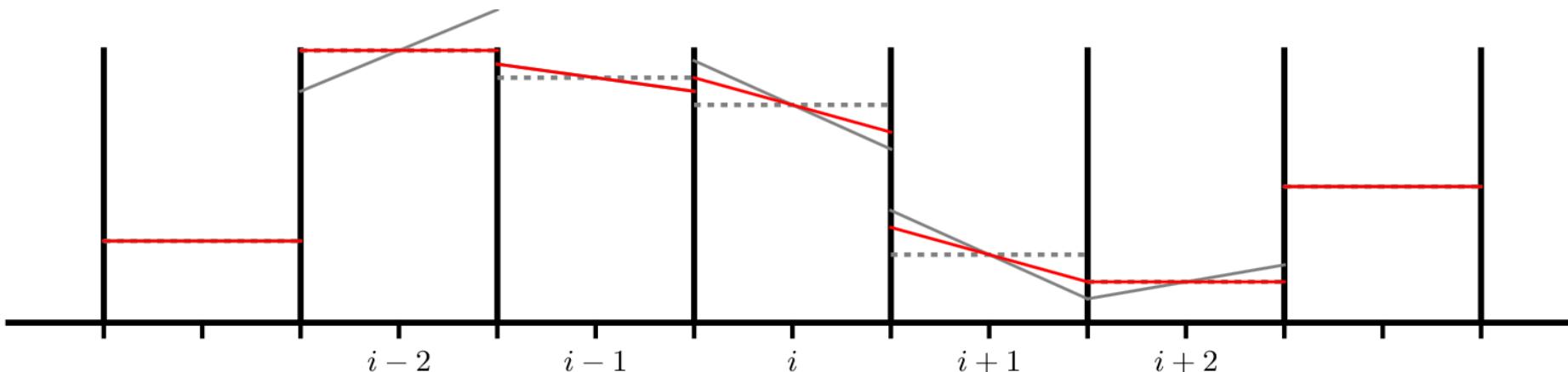
Видим
«overshooting»
для схемы Лакса-
Вендроффа

Д.В. Бисикало, А.Г. Жилкин,
А.А. Боярчук «Газодинамика
тесных двойных звезд» — М.:
Физматлит, 2013

Осцилляции, немонотонность и TVD-принцип

- К уравнению адвекции применим принцип максимума, решение всегда должно лежать внутри минимального и максимального значений
- Также у нас перестала сохраняться монотонность решения!
- Численная дисперсия совместно с немонотонностью приводит к возникновению осцилляций
- В более общем случае, говорят о полной вариации решения (A. Harten (1983)):

$$TV(\phi) = \sum_{j=1}^{N-1} |\phi_{j+1} - \phi_j|$$



Свойство TVD – total variation diminishing

- С.К. Годунов (1959) показал, что любая **линейная** схема для уравнения адвекции (справедливо и для других систем) не является монотонной, если ее **порядок точности выше первого**
- Как можно обойти это условие? Сделать схему **нелинейной!**
- Введем «анализатор» гладкости:

$$R = \frac{U_{i+1} - U_i}{U_i - U_{i-1}}$$

- $R \gg 1, R \ll 1 \Rightarrow$ находимся в области сильного градиента (разрыва)
- $R < 0 \Rightarrow$ находимся в точке экстремума

Ограничение наклонов при уточнении переменных (MUSCL-подход, vanLeer 1977,1982)

- Основная идея: Попробуем возвращаться к монотонной схеме с первым порядком там, где вариация нарастает
- В точках с сильным градиентом будем ограничивать наклон, а на экстремумах будем отключать второй порядок точности
- $U_L = U_i^n + \frac{\Delta x - a\Delta t}{2} \left(\frac{\partial U}{\partial x} \right)_i^n$; $U_R = U_{i+1}^n - \frac{\Delta x + a\Delta t}{2} \left(\frac{\partial U}{\partial x} \right)_{i+1}^n$ - градиенты надо ограничить!

$$\left(\frac{\partial U}{\partial x} \right)_i^n = \text{limiter} \left(\frac{U_i^n - U_{i-1}^n}{\Delta x}, \frac{U_{i+1}^n - U_i^n}{\Delta x} \right) = \frac{U_i^n - U_{i-1}^n}{\Delta x} * \text{limiter}(1, 1/R)$$

$$\left(\frac{\partial U}{\partial x} \right)_{i+1}^n = \text{limiter} \left(\frac{U_{i+1}^n - U_i^n}{\Delta x}, \frac{U_{i+2}^n - U_{i+1}^n}{\Delta x} \right)$$

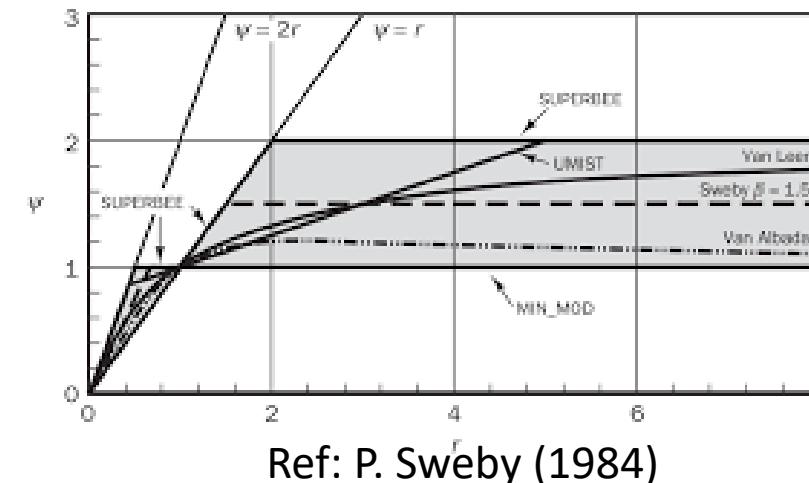
И снова считаем поток по уточненным значениям

$$F_{i+1/2}^{n+1/2} = \frac{aU_L + aU_R}{2} - |a| \frac{U_R - U_L}{2}$$

Виды ограничителей наклона (limiter'ы)

- Множество разных авторов и вариантов, наиболее популярные (см. также `reconstruction.py`, в нем функции '`PLM_rec`' и '`limiter`'):

```
def limiter(x, y, limiter_type):  
  
    # Smoothness analyzer  
    r = (y + 1e-14) / (x + 1e-14)  
  
    if limiter_type == 'VL':  
        #vanLeer limiter  
        df = x * (np.abs(r) + r) / (1.0 + np.abs(r))  
  
    elif limiter_type == 'MM':  
        # minmod limiter -- the most diffusive one (but the most stable)  
        df = 0.5 * x * (1.0 + np.sign(r)) * np.minimum(1.0, np.abs(r))
```



Ref: P. Sweby (1984)

- При использовании готовых пакетов зачастую на при вводе параметров симуляции также можно выбрать тип ограничителя

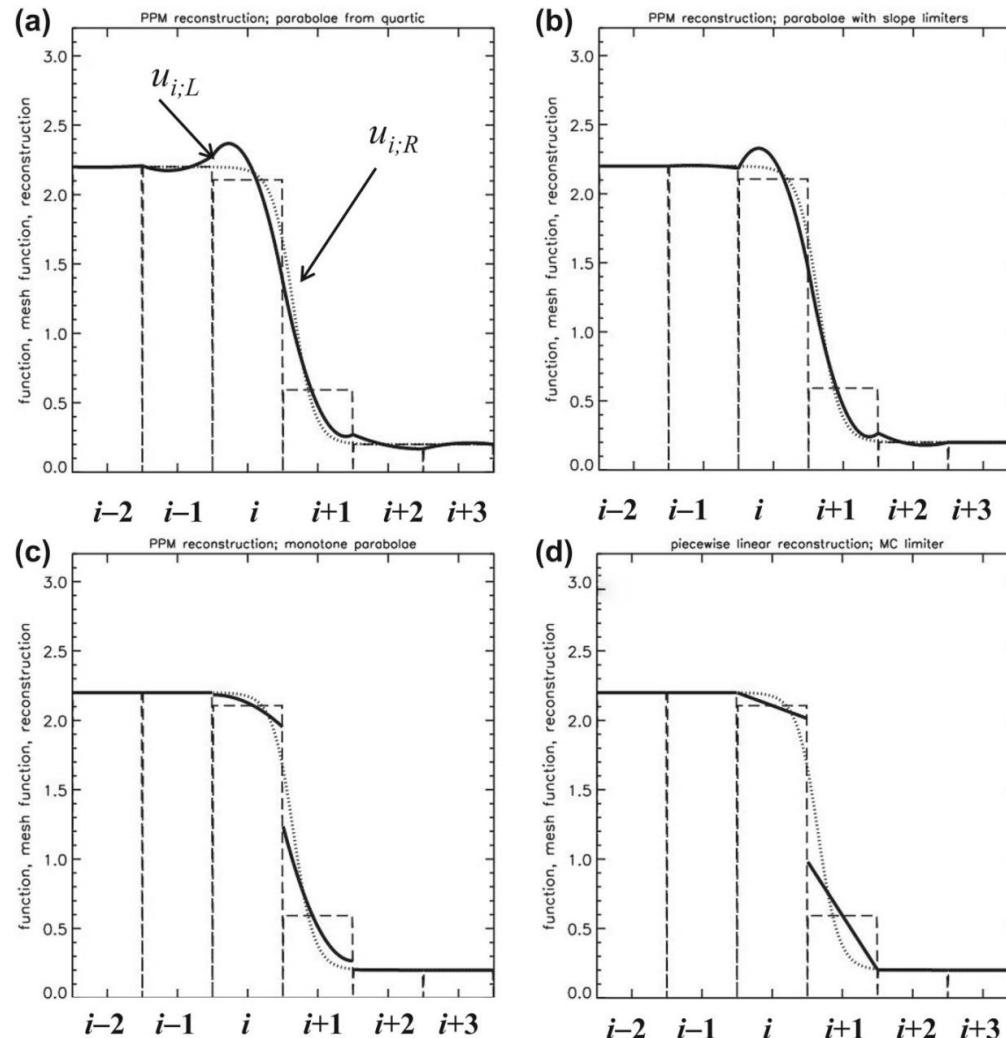
Квазимонотонные методы высокого порядка

- В итоге мы получили второй порядок везде кроме областей с большими градиентами/разрывами или экстремумами (**PLM**)
- Для **2D/3D** задач все делается **аналогично** – используются те же формулы, но в другом направлении
- Можно попробовать построить метод с третьим порядком – вместо линейных полиномов возьмем параболы (+ процедура «лимитирования») – такой подход будет называться кусочно-параболическим методом (**PPM**) [**Colella & Woodward 1984**]:

$$u_i(x) = \bar{u}_i + \hat{u}_x x + \hat{u}_{xx} \left(x^2 - \frac{1}{12} \right)$$

Piecewise parabolic method - PPM

- Детали можно найти в файле **reconstruction.py**



- Внутри ячейки аппроксимация функции имеет третий порядок, на границах может иметь любой порядок ≥ 3
- Выберем опцию «PPM» в файле **advection_solver_easy_run.ipynb** и сравним с «PLM»

$$u_{i;R} = \frac{7}{12} (\bar{u}_i + \bar{u}_{i+1}) - \frac{1}{12} (\bar{u}_{i-1} + \bar{u}_{i+2})$$

$$\hat{u}_x = u_{i;R} - u_{i;L}; \quad \hat{u}_{xx} = 3u_{i;R} - 6\bar{u}_i + 3u_{i;L}.$$

$$u_i(x) = \bar{u}_i + \hat{u}_x x + \hat{u}_{xx} \left(x^2 - \frac{1}{12} \right)$$

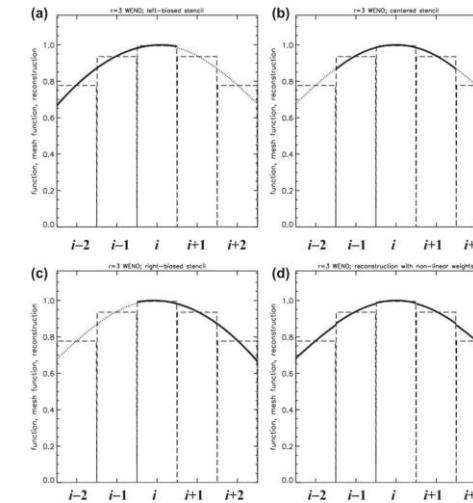
[D.S. Balsara – *liv rev comp astrophys* 3:2 (2017)]

Weighted Essentially Non-Oscillating method (WENO)

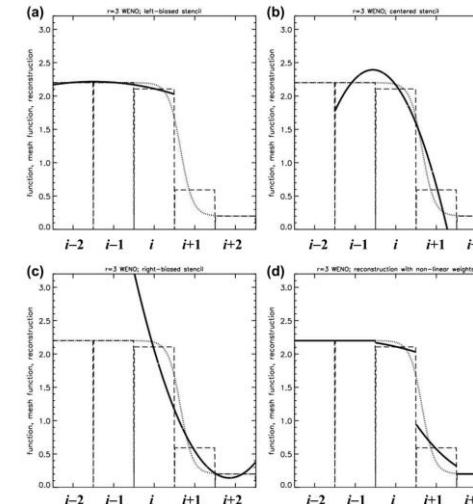
- Для более высоких порядков точности используют ENO/WENO подходы
- Рассматриваются три несколько разных вариантов уточнения, и из них выбирается наиболее гладкий вариант (ENO), либо их взвешенная комбинация (WENO)
- В последнем подходе точность может достигать 13 порядка!

[D.S. Balsara – *liv rev comp astrophys* 3:2 (2017)]

Детали в `reconstruction.py`



Гладкий профиль



Сильно переменный профиль

Методы Рунге-Кутты для интегрирования по времени (method-of-lines)

- Данный подход позволяет очень просто развивать схемы на высокий порядок по времени путем учета дополнительных стадий [Shu and Osher 1988 – TVD-методы] – удобно для 2D/3D подходов и схем высокого порядка

$$\frac{d\bar{\mathbf{U}}_i}{dt} + \frac{\mathbf{F}_{i+1/2} - \mathbf{F}_{i-1/2}}{\Delta x} = \bar{\mathbf{R}}_i$$

Получим систему ОДУ во времени
после аппроксимации по пространству

- RK2 – второй порядок

$$\begin{aligned}\mathbf{U}^{(1)} &= \mathbf{U}^n + \Delta t L(\mathbf{U}^n) \\ \mathbf{U}^{n+1} &= \frac{1}{2}\mathbf{U}^n + \frac{1}{2}\mathbf{U}^{(1)} + \frac{1}{2}\Delta t L(\mathbf{U}^{(1)})\end{aligned}$$

$L(\mathbf{U})$ – это наш
оператор с потоками
и уточнением по
пространству

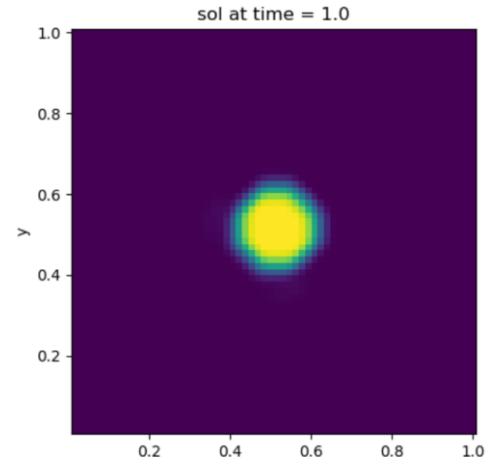
- RK3 – третий порядок

$$\begin{aligned}\mathbf{U}^{(1)} &= \mathbf{U}^n + \Delta t L(\mathbf{U}^n) \\ \mathbf{U}^{(2)} &= \frac{3}{4}\mathbf{U}^n + \frac{1}{4}\mathbf{U}^{(1)} + \frac{1}{4}\Delta t L(\mathbf{U}^{(1)}) \\ \mathbf{U}^{n+1} &= \frac{1}{3}\mathbf{U}^n + \frac{2}{3}\mathbf{U}^{(2)} + \frac{2}{3}\Delta t L(\mathbf{U}^{(2)})\end{aligned}$$

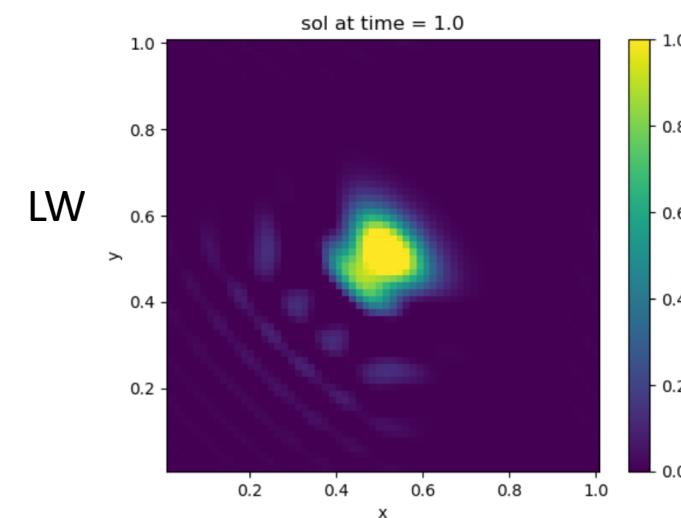
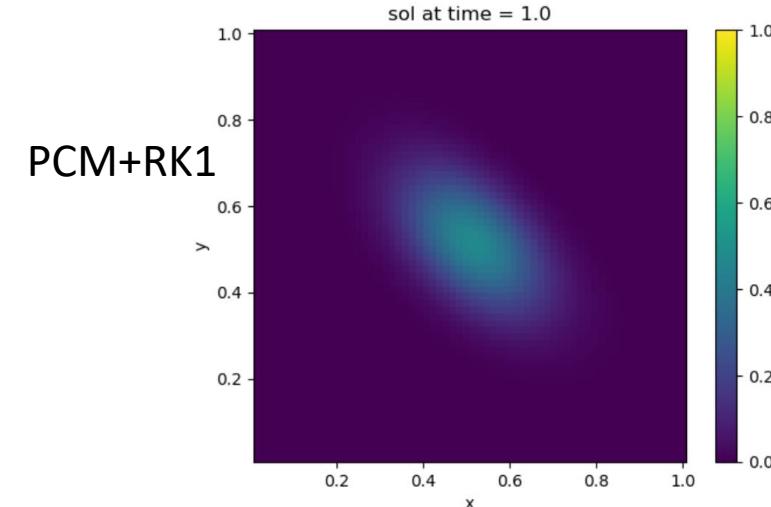
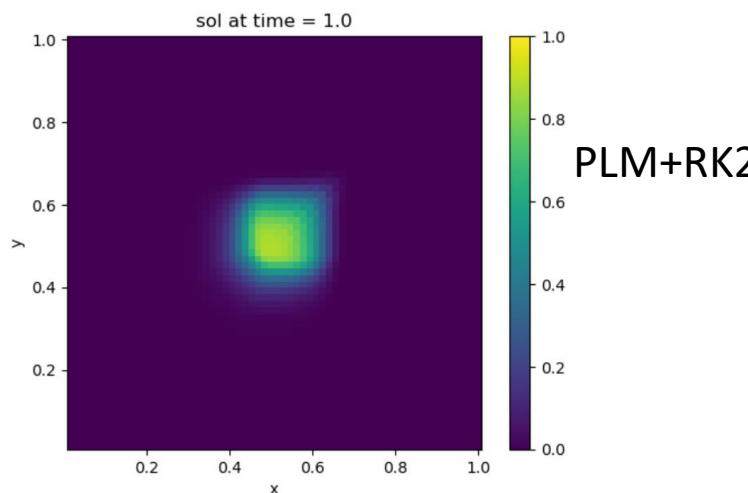
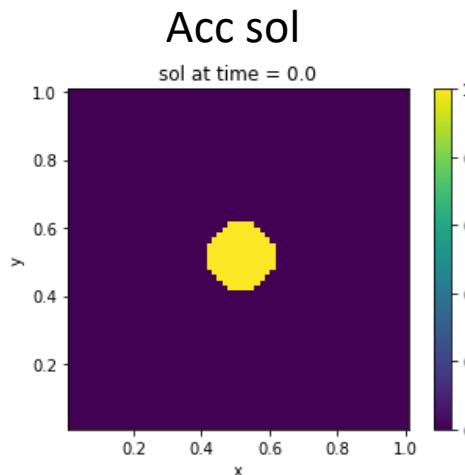
Рассмотрим многомерную адвекцию с разными порядками точности

- Файл **advection_solver_easy_run.ipynb**
- Рассмотрим пример ‘2D’ и методы **PPM + RK3 + adv**, **PLM + RK2 + adv**, **PCM + RK1 + adv**, а также метод **LW**
- PPM почти не изменяет решение, PLM слегка размазывает профиль, а PCM приводит к тому, что профиль полностью размазывается. Как и в 1D, схема LW вносит осцилляции

Рассмотрим многомерную адвекцию с разными порядками точности



Сетка 64x64



Теперь мы почти все знаем про
адвекцию и можем двигаться
далее!

- ✓ Метод конечного объема и сетка
 - ✓ Задача Римана
 - ✓ Устойчивость
- ✓ Методы высокого порядка

Газовая динамика

- Уравнения имеют схожую с адвекцией форму

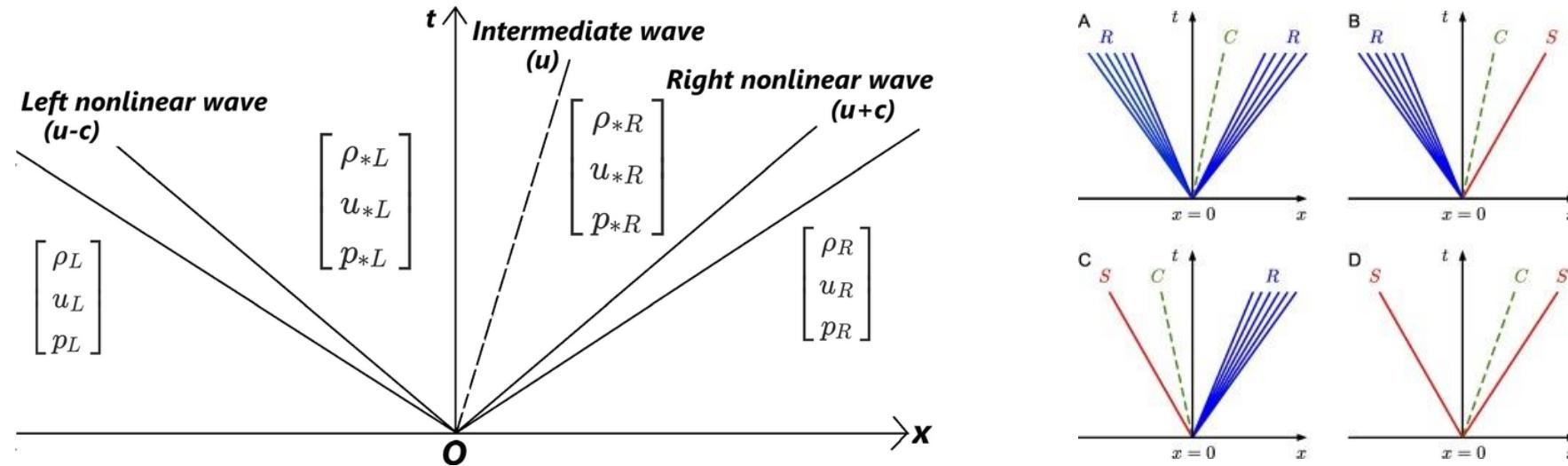
$$\mathbf{U} = \begin{pmatrix} \rho \\ \rho v_x \\ \rho v_y \\ E \end{pmatrix}, \quad \mathbf{F}_x = \begin{pmatrix} \rho v_x \\ \rho v_x^2 + p \\ \rho v_x v_y \\ (E + p)v_x \end{pmatrix}, \quad \mathbf{F}_y = \begin{pmatrix} \rho v_y \\ \rho v_x v_y \\ \rho v_y^2 + p \\ (E + p)v_y \end{pmatrix}$$

- Г.Д. – квазилинейная гиперболическая система дифференциальных уравнений в частных производных
- Три вида характеристик - $v_x \pm c_s$ (звуковые) и v_x (энтропийная)

По аналогии с адвекцией, можно определить задачу Римана вблизи каждой грани (ее также называют задачей о распаде произвольного разрыва) – решив эту задачу *точно*, мы получим метод Годунова [С.К. Годунов 1959]

Метод Годунова

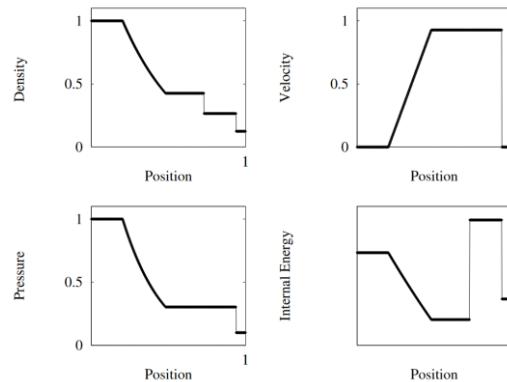
- В задаче о распаде произвольного разрыва могут возникать три вида волн – ударные волны, волны разрежения (простые волны) и контактные разрывы, соответствующие трем характеристикам



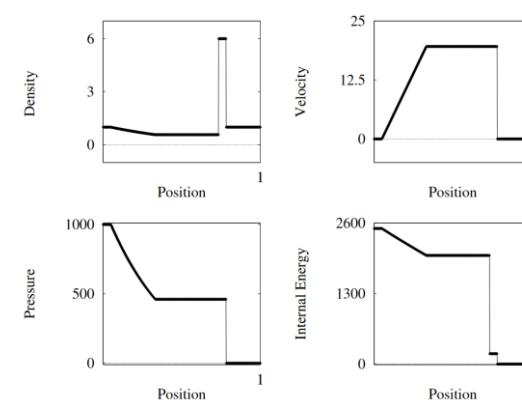
- Условие устойчивости (условие CFL) – задачи Римана не должны зависеть друг от друга – газ в ячейке должен «чувствовать» только локальные волны, которые успевают дойти из соседних ячеек (Более простая формулировка: **самое быстрое возмущение в моделируемой среде не должно распространяться дальше чем на одну ячейку за один шаг по времени**)
- Имеем следующее **условие устойчивости**: $\Delta t = CFL * \min_{i,j} \left(\frac{\Delta x}{|v_x| + c_s}, \frac{\Delta y}{|v_y| + c_s} \right)$, $CFL < 1$
for 1D and $CFL < 0.5$ for 2D in PLM methods with RK2 and ~ 0.6 and ~ 0.4 for 2D PPM with RK3

Пример – разные варианты задачи о распаде разрыва в газовой динамике

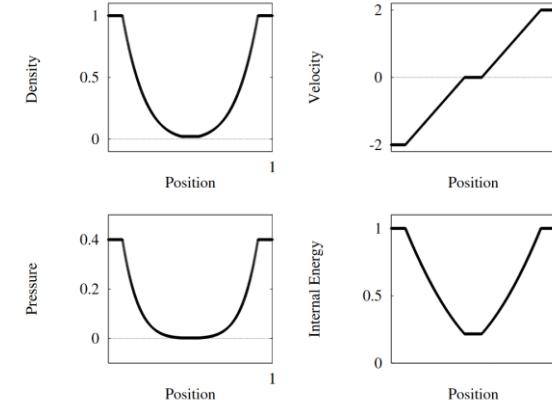
- Файл `fluid_solver_easy_run.ipynb`



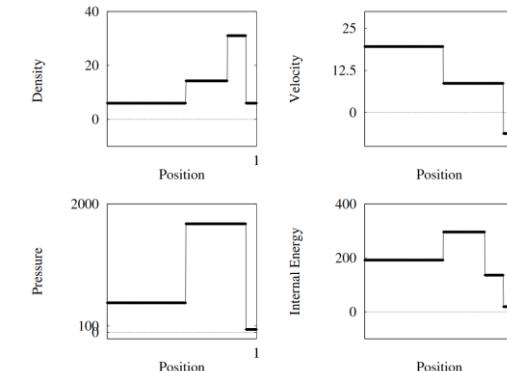
Test problem: Sod1D



Test problem:
strong1D



Две волны разрежения



Две УВ

Кстати, а почему разрыв вообще распадается? ☺

Метод Годунова

- Задача о распаде разрыва решается точно (**[Кочин 1928]**), однако решение данной задачи является очень громоздким и требует решения нелинейного алгебраического уравнения на каждой грани (см. ссылку 4, там очень подробно описано решение!)
- «+»: позволяет достаточно хорошо разрешать течения с разными разрывами, имеет минимальную численную вязкость
- «-»: вычислительно громоздкий, сложно развить на другие модели (напр., МГД), учесть неидеальное уравнение состояния и пр.

Более того, мы знаем сами состояния газа слева и справа от разрыва лишь приближенно, т.к. мы решаем наши уравнения с конечной точностью! Поэтому решать точно задачу Римана не всегда имеет смысл. Поищем способы попроще!

Методы Годуновского типа

Давайте решим задачу Римана приближенно!

- **Возможны два варианта:**

1. Предположение о более грубой волновой структуре в задаче Римана (напр., возможны только ударные волны) [**Harten, Lax and van Leer (1983)**] – часто используется физиками и астрофизиками
2. Вместо газодинамической нелинейной задачи решается линеаризованная (акустика), предполагая, что в каждой точке можно локально линеаризовать уравнения [**Roe (1981)**]

Методы Годуновского типа. Метод HLL (Harten, Lax and van Leer (1983))

- Детали см. в файле **Riemann_fluxes.py**

- Две ударные волны

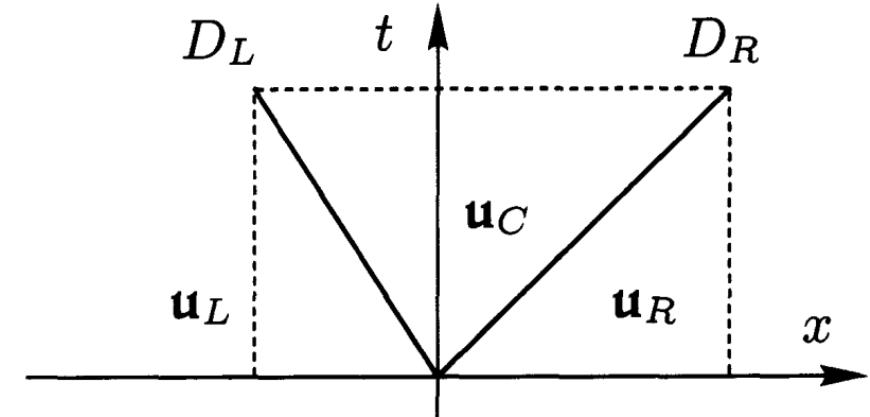
- Проводится оценка скоростей УВ:

$$D_L = \min(v_{x,L}; v_{x,R}) - \max(c_{s,L}, c_{s,R})$$

$$D_R = \max(v_{x,L}; v_{x,R}) + \max(c_{s,L}, c_{s,R})$$

- Записываются потоки газовых величин на УВ слева и справа от границы

- По характерным скоростям УВ выбирается поток (левый, центральный или правый)



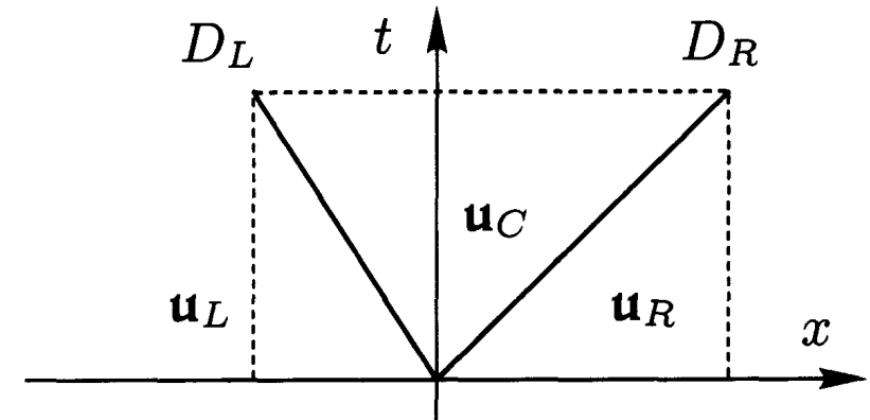
$$\mathbf{F}_* = \begin{cases} \mathbf{F}_L, & D_L \geq 0, \\ \mathbf{F}_C, & D_L \leq 0 \leq D_R, \\ \mathbf{F}_R, & D_R \leq 0. \end{cases}$$

Методы Годуновского типа. Метод HLL (Harten, Lax and van Leer (1983))

- Условия Гюгонио для ударных волн

$$D_L (\mathbf{u}_C - \mathbf{u}_L) = \mathbf{F}_C - \mathbf{F}_L,$$

$$D_R (\mathbf{u}_R - \mathbf{u}_C) = \mathbf{F}_R - \mathbf{F}_C.$$



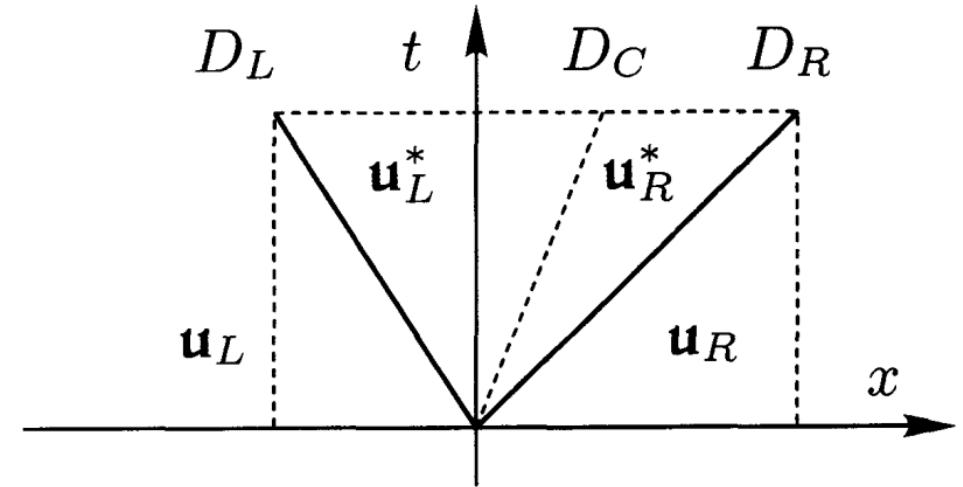
- Получаем решение для состояния газа потока внутри области между УВ

$$\mathbf{u}_C = \frac{D_R \mathbf{u}_R - D_L \mathbf{u}_L - \mathbf{F}_R + \mathbf{F}_L}{D_R - D_L},$$

$$\mathbf{F}_C = \frac{D_R \mathbf{F}_L - D_L \mathbf{F}_R + D_L D_R (\mathbf{u}_R - \mathbf{u}_L)}{D_R - D_L}.$$

Методы Годуновского типа. Метод HLLC (Toro, Spruce and Pierce (1994))

- Детали см. в файле **Riemann_fluxes.py**
- Предполагает наличие двух ударных волн и контактного разрыва – 4 состояния газа
- Записывая условия Рэнкина-Гюгонио для контактного разрыва и связывая их с условиями на УВ, получим более точную аппроксимацию
- Фактически данный метод является стандартом в настоящий момент

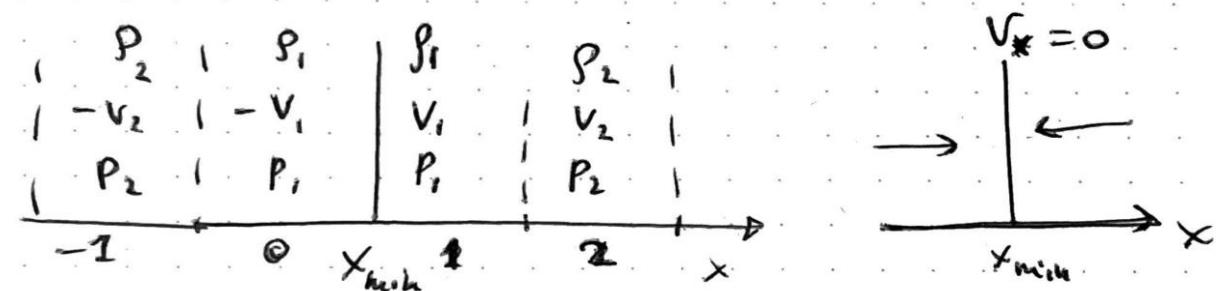


Учет контактных разрывов позволяет лучше разрешать турбулентные/неустойчивые течения

Границные условия – ghost cells

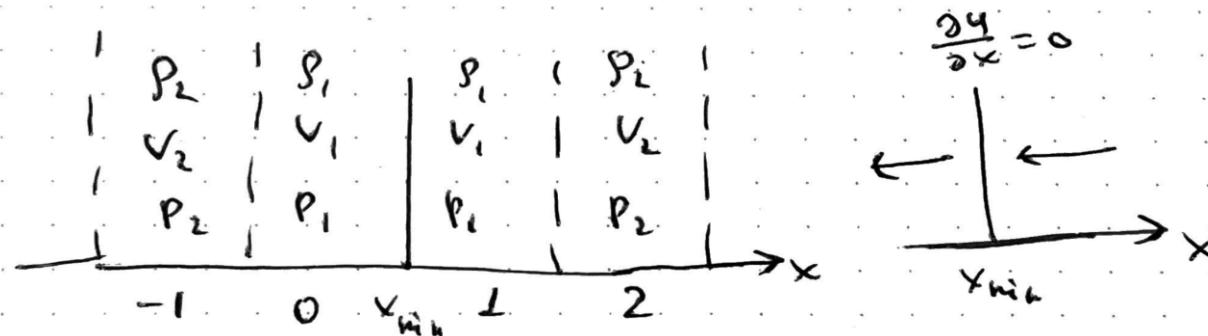
- Периодические – все как для адвекции, только для всех величин газа
- Условие «Стенка»

$$\rho_0 = \rho_1, v_{x,0} = -v_{x,1}, v_{y,0} = v_{y,1}, p_0 = p_1$$



- Условие «Свободная граница» (нулевой градиент)

$$\rho_0 = \rho_1, v_{x,0} = v_{x,1}, v_{y,0} = v_{y,1}, p_0 = p_1$$

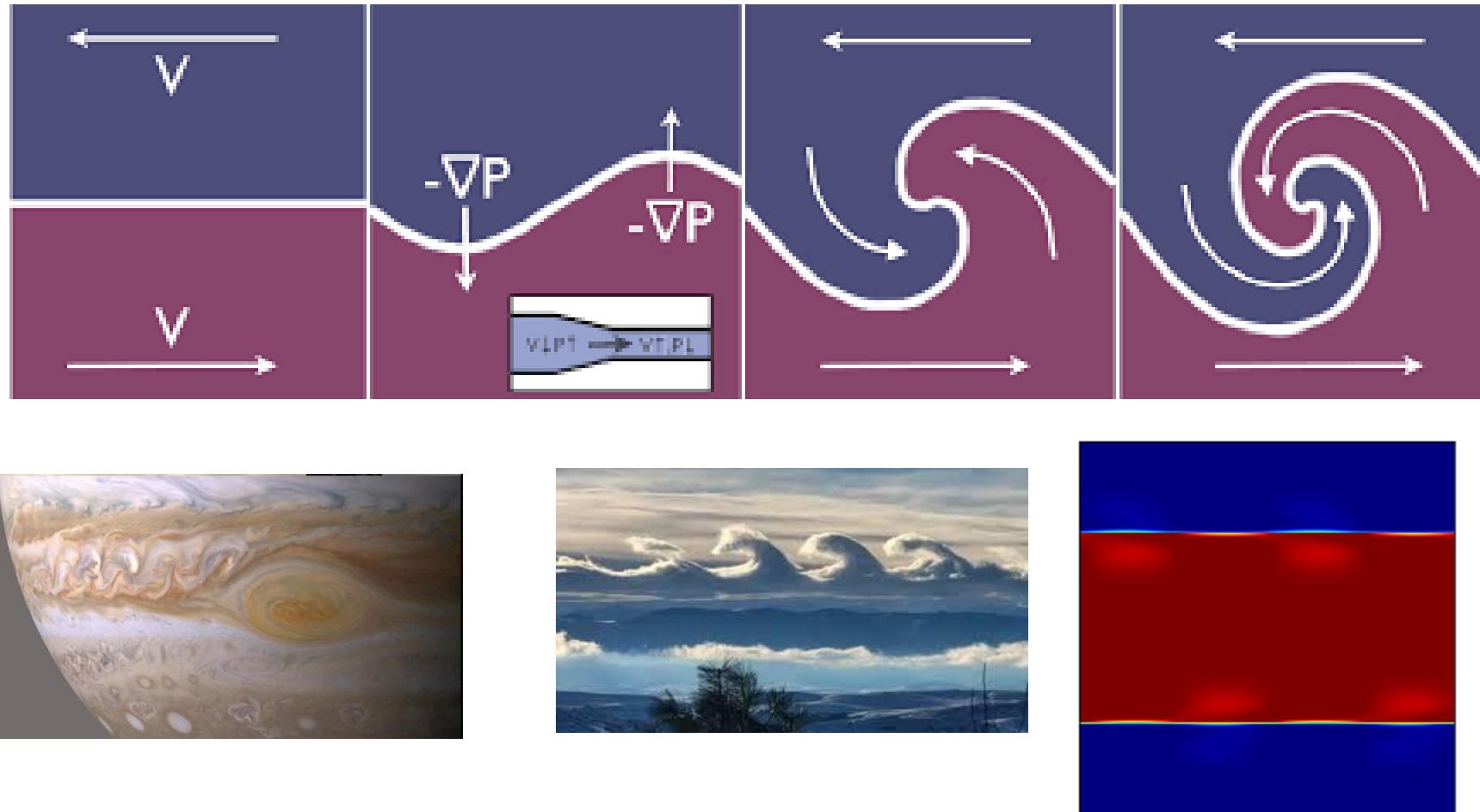


Как быть с высоким порядком и 2D?

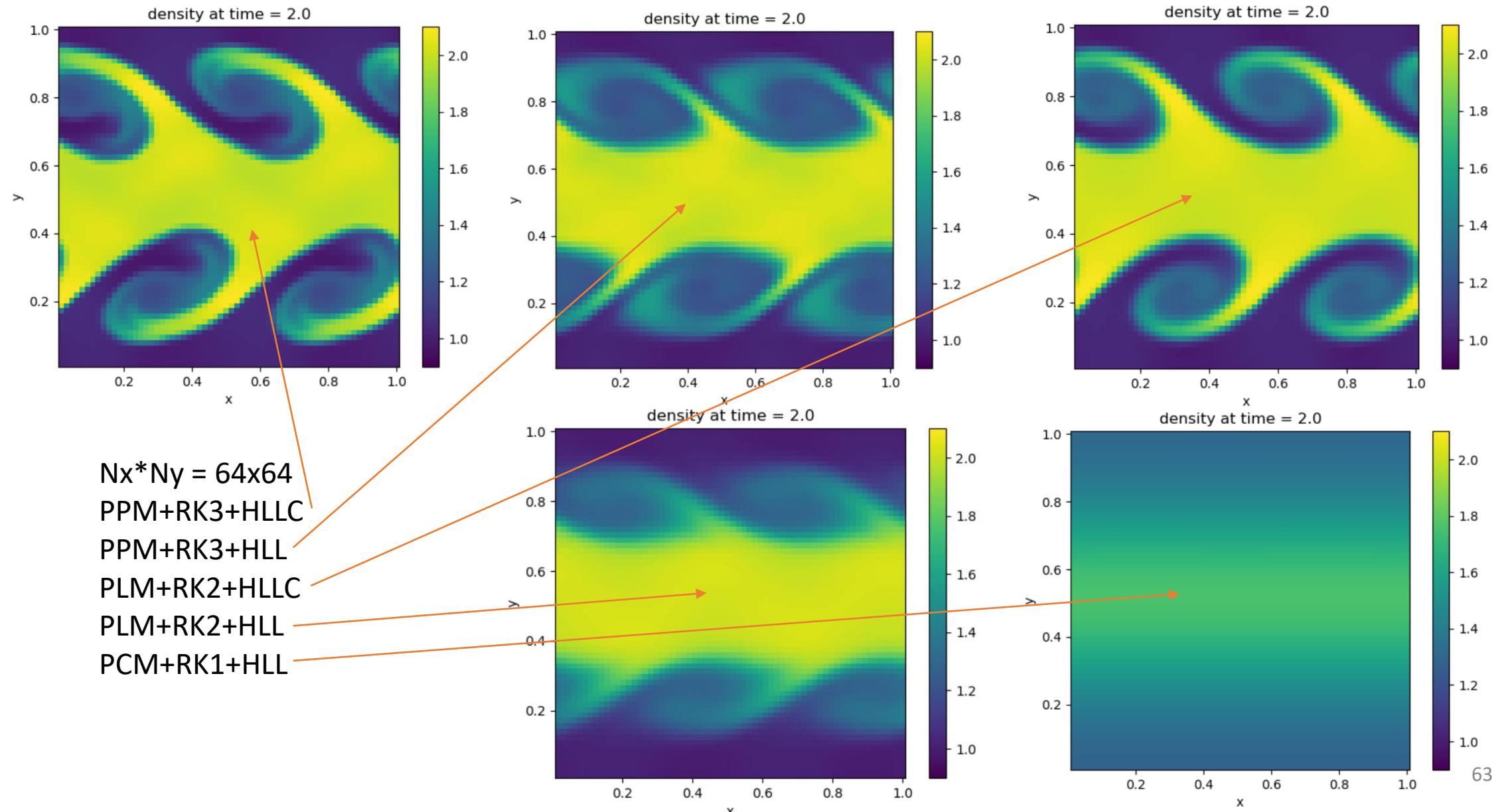
- Точно также! Однако, можно уточнять физические величины (плотность, скорость, давление), либо же сохраняющиеся (массу, импульс, энергию), а также «характеристические» переменные – инварианты Римана
- Выбирается метод аппроксимации по времени, по пространству, а также метод для нахождения потока (см. `Fluid_solver_easy_run.ipynb`)
- Потоки считаются вдоль x и вдоль y независимо друг от друга, для повышения порядка по времени используем методы `RK_X`
- В качестве сравнения методов рассмотрим генерацию неустойчивости Кельвина-Гельмгольца в `fluid_solver_easy_run.ipynb`
- Детали - см. файлы `one_step_***.py`

Пример: неустойчивость Кельвина-Гельмольца

- Неустойчивость сдвигового течения в жидкости
- Встречается в огромном количестве космических и земных объектов
- Для запуска можно задать “КНІ” + методы



Неустойчивость Кельвина-Гельмгольца



Магнитная гидродинамика

- Вид в форме законов сохранения:

$$\mathbf{U} = \begin{pmatrix} \rho \\ \rho v_x \\ \rho v_y \\ E \\ B_x \\ B_y \end{pmatrix}, \quad \mathbf{F}_x = \begin{pmatrix} \rho v_x \\ \rho v_x^2 + p + \frac{B_x^2 + B_y^2}{2} - B_y^2 \\ \rho v_x v_y - B_x B_y \\ \left(E + p + \frac{B_x^2 + B_y^2}{2}\right) v_x - \mathbf{v} \cdot \mathbf{B} B_x \\ 0 \\ B_y v_x - B_x v_y \end{pmatrix}, \quad \mathbf{F}_y = \begin{pmatrix} \rho v_y \\ \rho v_x v_y - B_x B_y \\ \rho v_y^2 + p + \frac{B_x^2 + B_y^2}{2} - B_y^2 \\ \left(E + p \frac{B_x^2 + B_y^2}{2}\right) v_y - \mathbf{v} \cdot \mathbf{B} B_y \\ B_x v_y + B_y v_x \\ 0 \end{pmatrix}$$

- Имеем уже 7 волн – две быстрых и две медленных магнитозвуковых волны, две альфеновских, а также энтропийная волна

Магнитная гидродинамика

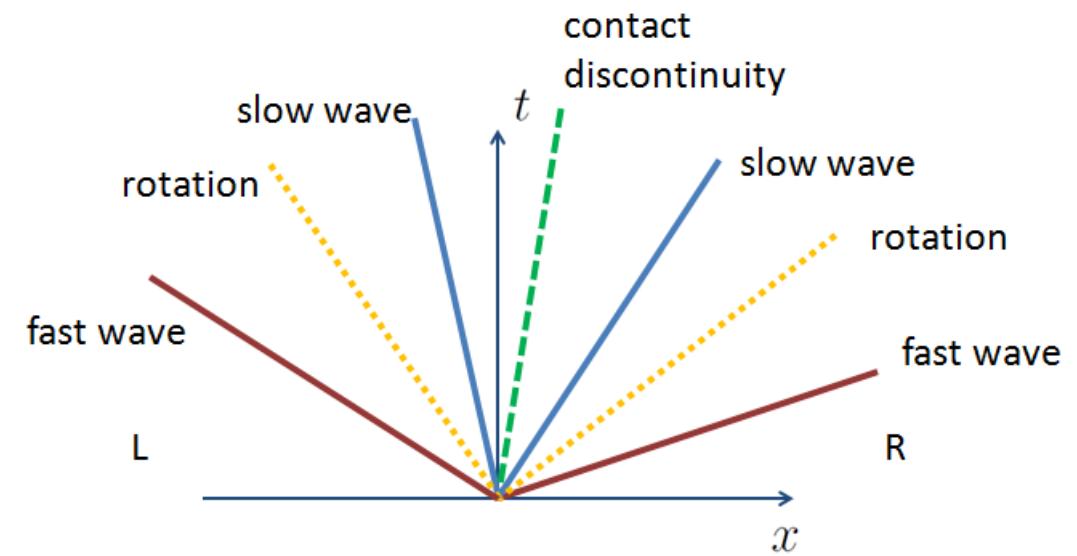
- Собственные значения и волновые решения

$$\lambda_B = 0, \quad \lambda_E = v_x, \quad \lambda_{\pm A} = v_x \pm a_x,$$

$$\lambda_{\pm S} = v_x \pm u_S, \quad \lambda_{\pm F} = v_x \pm u_F,$$

$$\mathbf{a} = \mathbf{B}/\sqrt{\rho}$$

$$u_{F,S} = \sqrt{\frac{c_s^2 + \mathbf{a}^2}{2}} \pm \frac{1}{2} \sqrt{(c_s^2 + \mathbf{a}^2)^2 - 4a_x^2 c_s^2} .$$



Методы Годуновского типа для МГД

- Точное решение задачи о распаде произвольного разрыва в МГД может иметь

400+ вариантов – DEAD END!



- Однако, мы снова можем использовать приближенные методы
- Все, что мы обсуждали про методы высокого порядка, также применимо в МГД

Пример: Метод HLL для уравнений МГД

- Детали см. в файле **Riemann_fluxes.py**

- Две быстрые МГД ударные волны

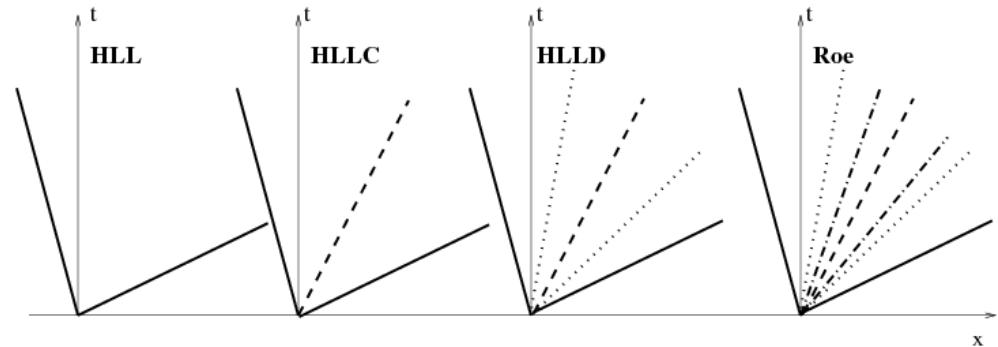
- Проводится оценка скоростей УВ:

$$D_L = \min(v_{x,L}; v_{x,R}) - \max(u_{F,L}, u_{F,R})$$

$$D_R = \max(v_{x,L}; v_{x,R}) + \max(u_{F,L}, u_{F,R})$$

- Записываются потоки МГД-величин для УВ слева и справа от границы

- По характерным скоростям УВ выбирается поток (левый, центральный или правый)

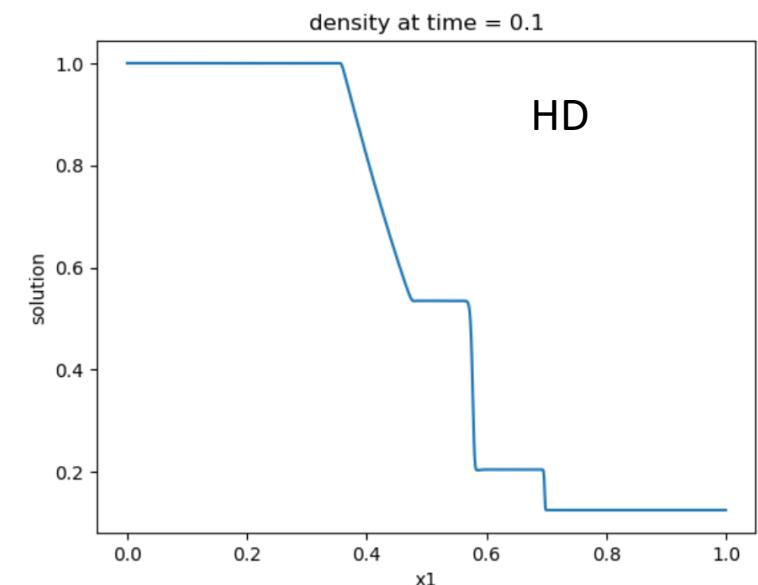
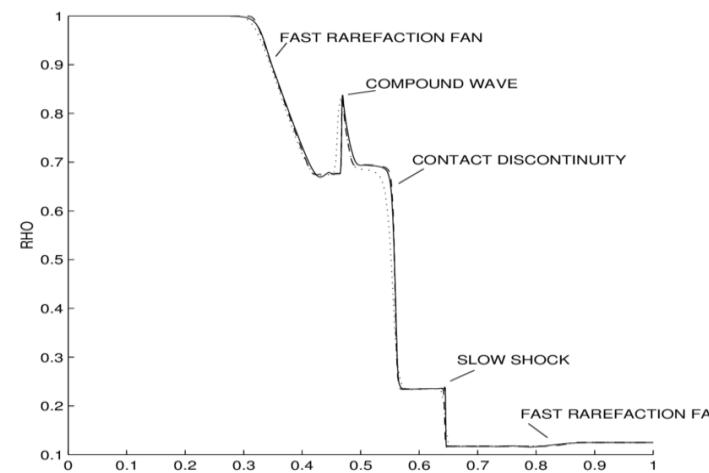
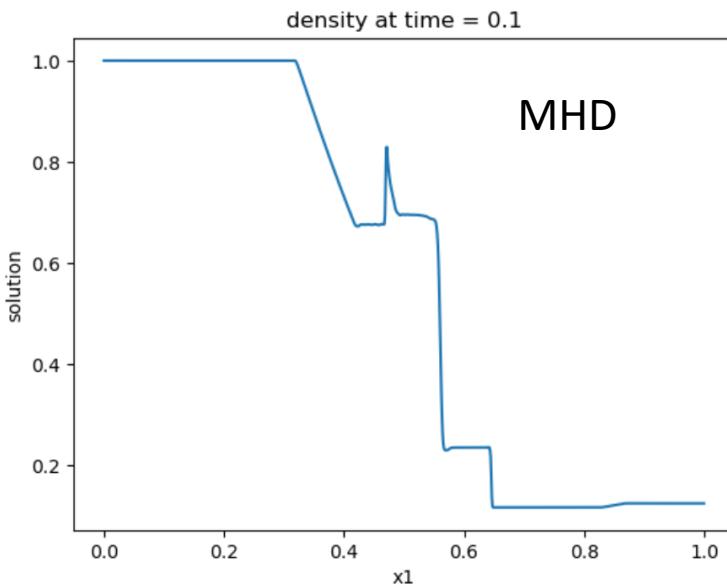


$$\mathbf{F}_* = \begin{cases} \mathbf{F}_L, & D_L \geq 0, \\ \mathbf{F}_C, & D_L \leq 0 \leq D_R, \\ \mathbf{F}_R, & D_R \leq 0. \end{cases}$$

$$\mathbf{F}_C = \frac{D_R \mathbf{F}_L - D_L \mathbf{F}_R + D_L D_R (\mathbf{u}_R - \mathbf{u}_L)}{D_R - D_L}$$

Сравнение с газовой динамикой для разных задач с разрывами

- Рассм. файл **mhd_solver_easy_run.ipynb**
- Запустим варианты BW1D и BW1D_0



Проблема возникновения ненулевой дивергенции в многомерных МГД-моделях

- 2D и 3D МГД-расчеты при помощи метода конечного объема могут приводить к численной неустойчивости
- Это происходит ввиду накопления магнитного заряда в точках сетки
- Дискретный аналог закона Фарадея не совпадает с дискретным уравнением на магнитное поле из метода конечного объема

$$\frac{\partial \mathbf{B}}{\partial t} = \nabla \times (\mathbf{v} \times \mathbf{B}) \quad \neq \quad \frac{\partial \mathbf{B}}{\partial t} + \nabla \cdot (\mathbf{vB} - \mathbf{Bv}) = 0$$

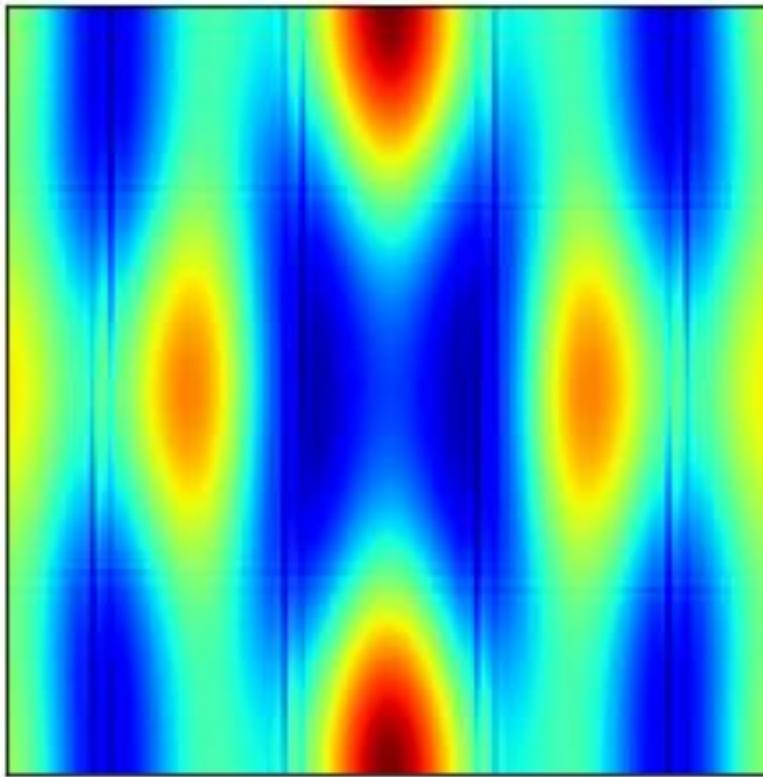
Связывает площадь и контур (магнитный поток через площадку сохраняется)

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

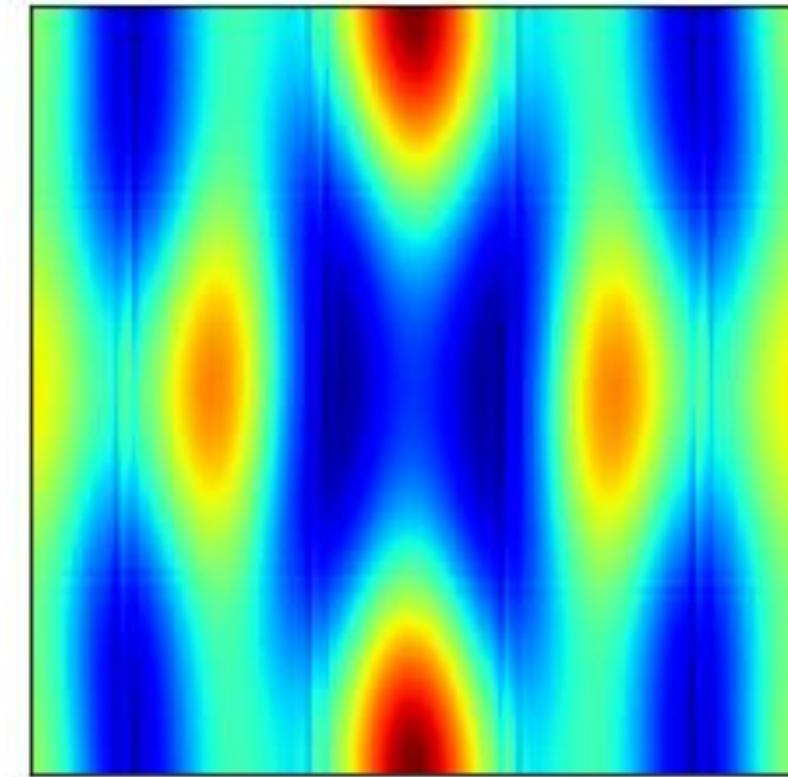
Связывает объем и площадь
(магнитный поток не сохраняется)

$$\nabla \cdot \mathbf{B} \neq 0$$

Пример – переход к МГД-турбулентности (Течение Орзага-Танга)



Есть контроль
дивергенции
(см. ниже)



Попробуйте убрать очистку
дивергенции в файле
`one_step_mhd.py` и посмотреть, что
получится с другими тестами 😊

Ничего не
делаем
(МКО)

Проблема возникновения ненулевой дивергенции в многомерных МГД-моделях

Что происходит? Пусть $\nabla \cdot \mathbf{B} = q_m \neq 0$

- Тогда по аналогии с силой, действующей на заряд вдоль электрического поля, должна возникать продольная полю «магнитодвижущая» сила

$$F \sim q_m * \mathbf{B} = \mathbf{B} \nabla \cdot \mathbf{B}$$

- Мы никак не учтываем в МКО эту силу + она является не физичной! Однако, поле эволюционирует так, что она должна быть

Методы решения

1. Ничего не делать, надеясь на лучшее (т.к. ошибка в дивергенции порядка ошибок аппроксимации)
2. Каждый шаг по времени удалять из распределения магнитного поля паразитную часть с дивергенцией (проекция на пространство бездивергентных функций)
3. Использовать метод, в котором магнитный заряд бы не возникал и на дискретном уровне
4. Модифицировать МГД-систему, вводя дополнительное уравнение на эволюцию дивергенции/магнитного заряда (чтобы $\text{div} \mathbf{B}$ затухала, «вымывалась» потоком из расчетной области и т.п.)

Варианты 2-4 активно используются

Очистка дивергенции

- Пусть получено поле \mathbf{B}^* после МКО, его можно разложить на вихревое и потенциальное поле:

$$\mathbf{B}^* = \nabla \times \mathbf{A} + \nabla \phi,$$

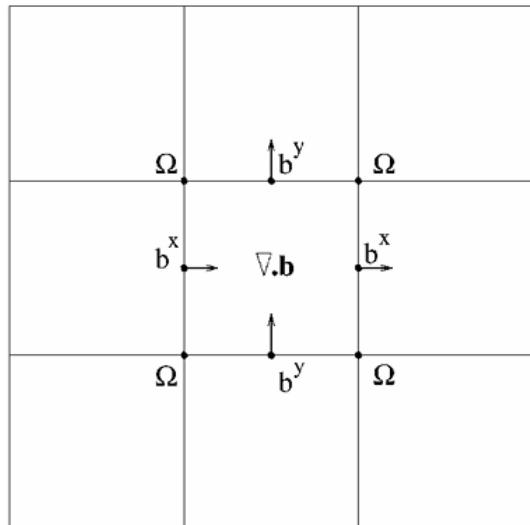
- Возьмем от него дивергенцию: $\nabla^2 \phi = \nabla \cdot \mathbf{B}^*$
- Получили уравнение Пуассона для поправки ϕ . Решив его, имеем:

$$\mathbf{B}^{n+1} = \mathbf{B}^* - \nabla \phi.$$

«+» работает почти всегда
«-» уравнение Пуассона
бывает дорого решать

Метод Constrained transport (CT) – МГД-моделирование с $\operatorname{div} \mathbf{B} = 0$

- Закон Фарадея для магнитного поля может быть записан на грани ячеек ($\Omega = E_z$) – «переселим» поле на грань



$$b_{j+1/2,k}^{x,n+1} = b_{j+1/2,k}^{x,n} - \Delta t \frac{\Omega_{j+1/2,k+1/2} - \Omega_{j+1/2,k-1/2}}{\Delta y}$$
$$b_{j,k+1/2}^{y,n+1} = b_{j,k+1/2}^{y,n} + \Delta t \frac{\Omega_{j+1/2,k+1/2} - \Omega_{j-1/2,k+1/2}}{\Delta x}.$$

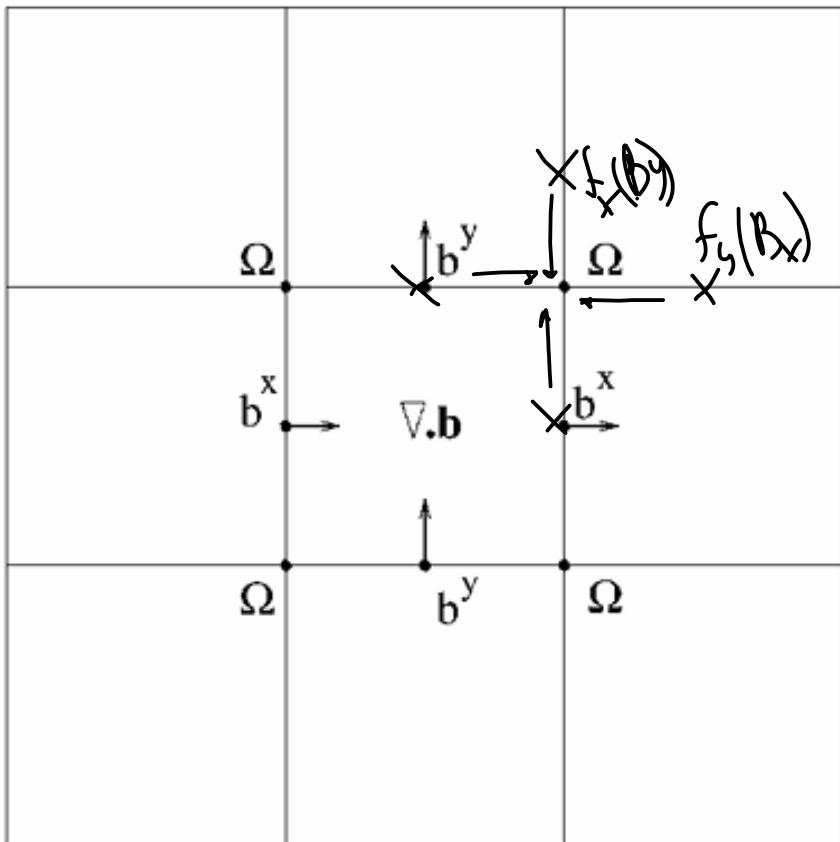
Можно показать, что в этом случае будет сохраняться дивергенция поля в следующем виде:

Но как нам найти электрическое поле в углах?

$$(\nabla \cdot \mathbf{b})_{j,k} = \frac{b_{j+1/2,k}^x - b_{j-1/2,k}^x}{\Delta x} + \frac{b_{j,k+1/2}^y - b_{j,k-1/2}^y}{\Delta y}$$

Метод Constrained transport

- Будем интерполировать электрическое поле в углы наших ячеек, зная, что потоки магнитного из МКО поля равны E_z на гранях



$$F_y(B_x) = v_y B_x - v_x B_y = E_z = \Omega$$
$$F_x(B_y) = -v_y B_x + v_x B_y = -E_z = -\Omega$$

$$\Omega_{j+1/2,k+1/2} = \frac{1}{4} (-f_{j+1/2,k}^{x,*} - f_{j+1/2,k+1}^{x,*} + f_{j,k+1/2}^{y,*} + f_{j+1,k+1/2}^{y,*})$$

В конце шага усредним поле в ячейку из граней

Получили поле без дивергенции!
«+» – метод максимально физичный
«-» – требует наличия разнесенных сеток

$$B_{j,k}^{x,n+1} = \frac{b_{j+1/2,k}^{x,n+1} + b_{j-1/2,k}^{x,n+1}}{2}$$

$$B_{j,k}^{y,n+1} = \frac{b_{j,k+1/2}^{y,n+1} + b_{j,k-1/2}^{y,n+1}}{2}$$

Модификация МГД-систем – 8-wave

- Два наиболее употребляемых варианта – 8-волной метод (Powell (1999)) и метод обобщенного множителя Лагранжа (Dedner et al (2002))
- Восьмиволной метод: рассмотрим МГД-систему, в которой **есть** дивергенция магнитного поля. Учтем эту силу во всех уравнениях.

Детали в файле
`one_step_mhd.py`

$$\frac{\partial(\rho\mathbf{v})}{\partial t} + \nabla \cdot \left[\rho\mathbf{v}\mathbf{v} + \left(p + \frac{\mathbf{B}^2}{8\pi} \right) \mathbf{I} - \frac{\mathbf{B}\mathbf{B}}{4\pi} \right] = -\mathbf{B}(\nabla \cdot \mathbf{B})$$

$$\frac{\partial E}{\partial t} + \nabla \cdot \left[\left(E + p + \frac{\mathbf{B}^2}{8\pi} \right) \mathbf{v} - \frac{(\mathbf{v} \cdot \mathbf{B})\mathbf{B}}{4\pi} \right] = -\mathbf{B} \cdot \mathbf{v}(\nabla \cdot \mathbf{B})$$

$$\frac{\partial \mathbf{B}}{\partial t} + \nabla \cdot (\mathbf{v}\mathbf{B} - \mathbf{B}\mathbf{v}) = -\mathbf{v}(\nabla \cdot \mathbf{B})$$

Дивергенция поля переносится
вместе с потоком (8-ая волна)

$$\frac{\partial(\nabla \cdot \mathbf{B})}{\partial t} + \nabla \cdot ((\nabla \cdot \mathbf{B})\mathbf{v}) = 0$$

«+» – очень
простой
«-» – портится
консервативность

Модификация МГД-систем - GLM

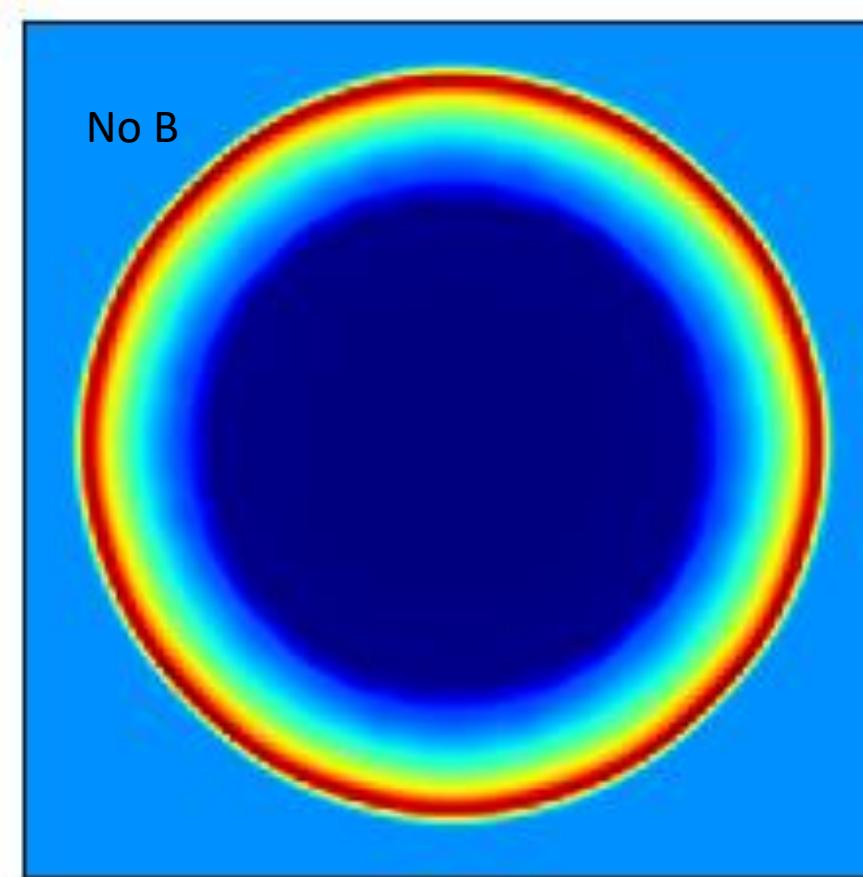
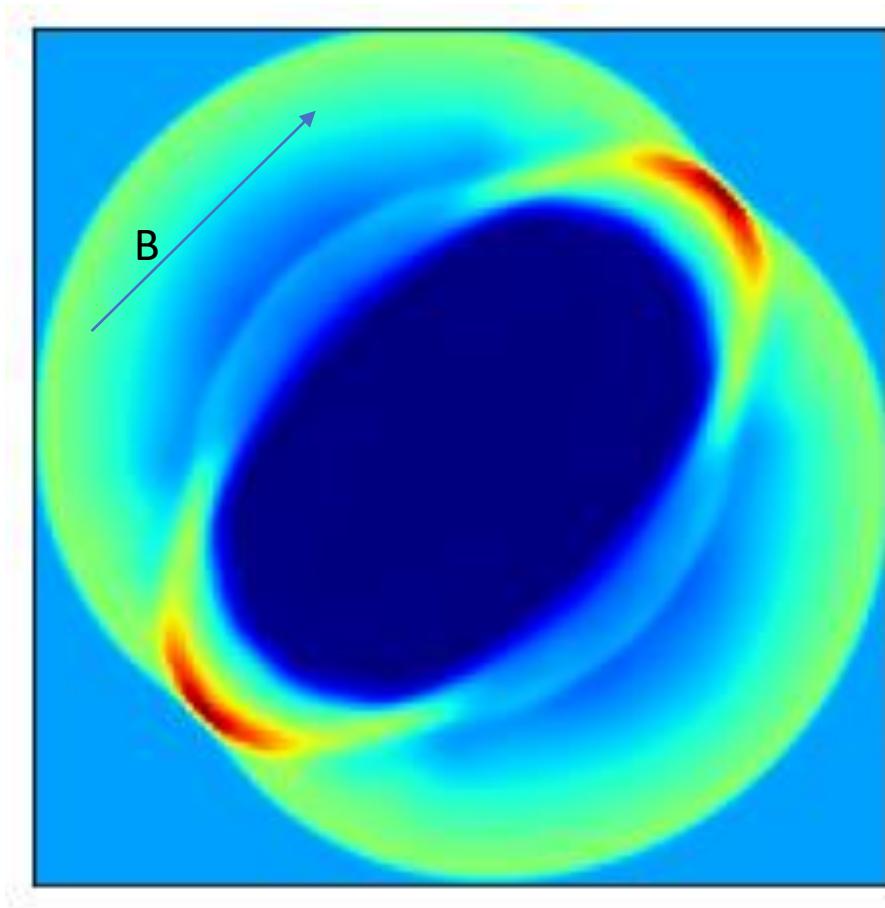
- В рамках подхода **GLM** решается дополнительное уравнение на множитель Лагранжа, который связан с переносом нормальной компоненты поля
- дивергенция магнитного поля подчиняется телеграфному уравнению (перенос со скоростью c_h + затухание на времени $\sim \tau$):

$$\frac{\partial \mathbf{B}}{\partial t} + \nabla \cdot (\mathbf{v}\mathbf{B} - \mathbf{B}\mathbf{v}) + \nabla\psi = 0,$$

c_h - выбирается из условия
устойчивости; $\frac{1}{\tau} \sim 0.1 \frac{c_h}{\Delta x}$

$$\frac{\partial \psi}{\partial t} + c_h^2 \nabla \cdot \mathbf{B} = \frac{\psi}{\tau}.$$

2D Взрыв в замагниченной среде vs в не замагнитенном газе



Пример expl2d

Доп.: Учет диссипативных процессов

- Вязкость, диффузия, тепло- и электропроводность имеют схожие математические модели:

$$\frac{\partial f}{\partial t} = \nabla \cdot (\chi \nabla f)$$

Пример для теплопроводности

$$\frac{\partial E}{\partial t} + \nabla \cdot (F^{\text{adv}} - \kappa \nabla T) = 0,$$

$$\Delta E = \Delta E_{ideal} + \Delta E_{heat}$$
$$\Delta E_{heat} = \kappa \left(\frac{T_{i-1j} - 2T_{ij} + T_{i+1j}}{\Delta x^2} + \frac{T_{ij-1} - 2T_{ij} + T_{ij+1}}{\Delta y^2} \right)$$

Доп.: Учет диссипативных процессов

В форме потока: $F_{i+1/2,j}^{heat} = -\kappa \left(\frac{T_{i+1,j} - T_{i,j}}{\Delta x} \right)$ – дискретный аналог закона Фурье

$$F_{i+1/2,j}^{total} = F_{i+1/2,j}^{ideal} + F_{i+1/2,j}^{heat}$$

Условие CFL

$$\Delta t = \min \left(\Delta t^{ideal}, 0.5 \min_{i,j} \left(\frac{\Delta x^2}{\kappa}, \frac{\Delta y^2}{\kappa} \right) \right)$$

Может быть очень жестким ввиду квадратичной зависимости от шага сетки

Задачи, которые вы можете сделать с кодом

Для адвекции

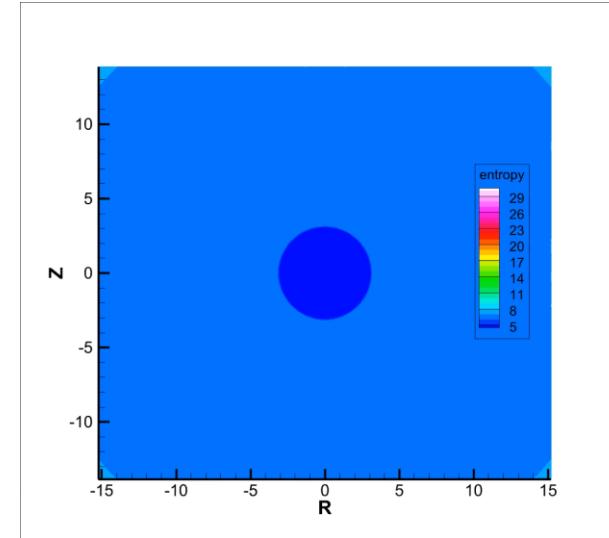
- Реализовать уравнение Бюргерса вместо линейной модели [см. ссылки 1,2,4]
- Добавить диссиацию (аналог вязкости) [см. ссылки 1 и 2]

Для газовой динамики/МГД

- Реализовать цилиндрическую или сферическую геометрию [см. ссылку 6]
- Реализовать уравнения релятивистской газовой динамики [см. ссылки 5,8,9]
- Реализовать бездивергентный метод для эволюции магнитного поля [см., напр., ссылки 3,7]
- Добавить вязкость в газовую динамику [см., напр., ссылку 9]
- Добавить электропроводность в МГД [см., напр., ссылку 9]

Заключение/take home

- Большое количество моделей плазмы/газа/жидкости в астрофизических (и не только) условиях можно исследовать численно, применяя относительно простые и универсальные подходы, которые используют понятия конечного объема и обмена потоками сохраняющихся величин между ячейками
- Написать код для моделирования разных плазменных течений можно, используя «кирпичики» в виде функций потока, уточнения величин и т.д. (Это проще, чем кажется! 😊)
- Также можно использовать готовые пакеты, если это возможно, но необходимо понимать, как они работают!



Evolution (0-1.6s) of
Specific entropy for
asymmetric MR
supernova
(low-to-intermediate-
B case)

Спасибо за внимание!