# Implicit kinetic schemes for fluid models

J. Badwaik, David Coulette, E. Franck, P. Helluy, C. Hillairet, H. Oberlin, M. Mehrenberger, L. Mendoza L. Navoret

IRMA Strasbourg & Inria TONUS, France

IPL FRATRES Meeting
Nov-18-2016

# Outline

# Section 1

## Context and overall approach

# Target problem

## Typical problem

Unknowns $w_k(t, \mathbf{x}), k = 1, N$ macroscopic fields.
System of (mostly) hyperbolic conservation laws

$$\partial_t w_k + \nabla \cdot \mathbf{\Phi}_k(w) = \mathcal{D}_k(w) + \mathcal{S}_k$$

$\mathbf{\Phi}_k$ : fluxes (nonlinear) - $\mathcal{D}_k$ : parabolic terms (diffusion).
Systems of interest : Euler/Navier Stokes - MHD

## Numerical challenges

- explicit schemes
  - CFL conditions :time scale constrained by space grid
  - forces to resolve possibly unwanted fast times scales
- implicit schemes
  - large nonlinear system
  - costly matrix assembly/storage/inversion.

# Kinetic schemes

Distribution function $f(t, \mathbf{x}, \mathbf{v})$.
Boltzmann-BGK equation

$$\partial_t f + \nabla \cdot (f \mathbf{v}) = \frac{1}{\tau}(F^{eq}(Mf), \mathbf{v}) - f)$$

with

- macroscopic data $m(t, \mathbf{x}) = Mf = \int K(\mathbf{v}) f(t, \mathbf{x}, \mathbf{v}) d^3 v$ obtained by **linear** map from $f$
- collision vector $K(\mathbf{v}) \in \mathbb{R}^N$.
- $F^{eq}(m)$ equilibrium state
  - $\int K(\mathbf{v}) F^{eq}(m, \mathbf{v}) d^3 v = m, \forall m$ (macro conservation)
  - $\displaystyle\int s(F^{eq}(m)) d^3 v = \max_{\mathcal{M}f=m} \int s(f) d^3 v$, $s$ entropy

# Kinetic schemes

In the limit of short relaxation times $\tau \to 0$

$$\partial_t m + \nabla \cdot \mathbf{\Phi}(m) = 0, \quad \text{with } \mathbf{\Phi}(m) = \int \mathbf{v} F^{eq}(m, v) d^3 v$$

## Basic idea

Solving the **split** transport/relaxation kinetic system for small $\tau$ provides a natural scheme to approximate the relaxed system.

## Interesting Features

- transport stage (T) is **linear**

$$\partial_t f + \nabla \cdot (f \mathbf{v}) = 0$$

- nonlinearities in the relaxation stage (R) are **local** :

$$\partial_t f = (1/\tau)(F^{eq}(Mf) - f)$$

- finite $\Delta t$ (splitting) or/and $\tau$ generate generate additional diffusive terms.

# Lattice Boltzmann schemes

## A particular discretization of $f(t, \mathbf{x}, \mathbf{v})$

- $v$ : small set of discrete velocities $\mathbf{v}_i, i = 1, q$
- Finite set of Boltzmann-BGK equations coupled only through relaxation

$$\partial_t f_i + \nabla \cdot \mathbf{v}_i f_i = \frac{1}{\tau}(f_i^{eq} - f_i), \forall i$$

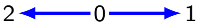
- $x$ : structured cartesian mesh $\mathbf{x}_k$ generated by the velocity set for a given time scale $\Delta t$

$$\forall (k, k'), \exists (i, j) \in [1, q] \times \mathbb{Z}, \mathbf{x}_k - \mathbf{x}'_\mathbf{k} = j \Delta t \mathbf{v}_i$$

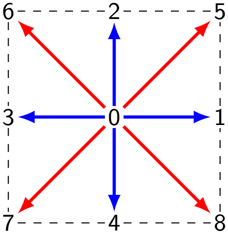
- splitting scheme
  1. exact transport : $f_i^\star(\mathbf{x}) = f_i(t^n, \mathbf{x} - \Delta t \mathbf{v}_i)$
  2. local relaxation : $f_i(t^n + \Delta t) = (1 - s) f_i^* + s f^{eq}(m(f^\star))$
     relaxation parameter $s = \frac{2\Delta t}{2\tau + \Delta t}$ (Crank-Nicolson)
- $\tau \ll \Delta t$ (over-relaxation) fast oscillations around equilibrium manifold

# Standard Lattice Boltzmann models

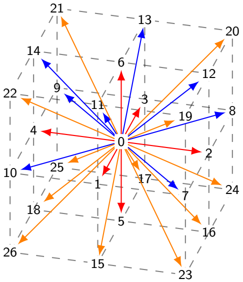
Notation : DdQq with $d = 1, 2, 3$ space dimension $q$ number of velocities.




D1Q3




D2Q9




D3Q27

- $K$ built from low-order polynomials : macro quantities are moments.
- splitting error generates diffusive terms : can mimic physical diffusion
- Can be applied to any hyperbolic system of conservations laws (fluid mechanics [CD98], Maxwell [Gra14], MHD [Del02], etc.)
- transport is easy but $\Delta t / \Delta x$ linked : integer CFL-like condition

Section 2

# A tour in DG Lattice Boltzmann schemes family

# Simple example D1Q3 (1)

- $v_i = i\lambda, i \in \{-1, 0, 1\}$.
- $K(v) = (1, v)^T \rightarrow w = (\rho, q)$. Density $\rho$, momentum $q = \rho u$.
- limit system is the Euler isothermal equation

$$\left\{ \begin{array}{ll} \sum_i [\partial_t f_i + \partial_x (v_i f_i) - \tau^{-1}(f_i^{eq} - f_i)] & = 0 \\ \sum_i [\partial_t (f_i v_i) + \partial_x (v_i^2 f_i) - \tau^{-1}(v_i f_i^{eq} - v_i f_i)] & = 0 \end{array} \right. \rightarrow \left\{ \begin{array}{ll} \partial_t \rho + \partial_x q & = 0 \\ \partial_t q + \partial_x (q^2/\rho + c^2 \rho) & = 0 \end{array} \right.$$

- $f^{eq}$ is an equilibrium (Maxwellian) state if:

$$\rho = \sum_i f_i^{eq} \quad q = \sum_i f_i^{eq} v_i \quad \frac{q^2}{\rho} + \rho c^2 = \sum_i f_i^{eq} v_i^2$$

- solving the linear system for $f^{eq}$ we obtain

$$f_0^{eq} = \rho(\lambda^2 - u^2 - c^2)/\lambda^2 \quad f_{\pm 1}^{eq} = \frac{\rho}{2}\left(\pm\lambda u + u^2 + c^2\right)/\lambda^2$$

# Simple example D1Q3 (2)

Let's consider the extended moment set $(\rho, q, z)$, with $z = \sum_i f_i v_i^2$.
In moment space, the D1Q3 model reads

$$\begin{aligned} \partial_t \rho + \partial_x q &= 0, \\ \partial_t q + \partial_x z &= 0, \\ \partial_t z + \lambda^2 \partial_x q &= \tau^{-1}(z^{eq}(\rho, q) - z) = \tau^{-1}(q^2/\rho + c^2\rho - z). \end{aligned}$$

Chapman-Enskog method shows that for small $\tau$ we have

$$\begin{aligned} \partial_t \rho + \partial_x q &= 0, \\ \partial_t q + \partial_x \left( \frac{q^2}{\rho} + c^2 \rho \right) &= \tau \partial_x \left( \underbrace{\left(\lambda^2 - c^2 - 3u^2\right)}_{sign!} \partial_x q + 2u \left(u^2 - c^2\right) \partial_x \rho \right). \end{aligned}$$

The viscosity terms are not entropy dissipative → small Mach flows.
Different from the Jin-Xin relaxation [JX95, Dub13].

# Dellar's approach : D2Q9 + 2 x D2Q5 for 2D MHD

Bibliography [Del02]

Basic resistive MHD
$$\begin{cases} \partial_t \rho + \nabla \cdot (\rho \mathbf{u}) = 0 \\ \partial_t (\rho \mathbf{u}) + \nabla \cdot \left[ (p + B^2/2)\mathbb{I} + \rho \mathbf{u}\mathbf{u} - \mathbf{B}\mathbf{B} \right] = \nabla \cdot (\mu S) \\ \partial_t \mathbf{B} + \nabla \times [\eta \nabla \times \mathbf{B} - \mathbf{u} \times \mathbf{B}] = 0 \end{cases}$$

## LBM modelisation

- fluid part : standard Euler/Navier Stokes ; Lorentz force → included in the equilibrium flux
- induction equation : **B** cannot be directly cast as a first order moment (due to antisymmetry).
- Dellar's approach
  - associate each component of **B** to a separate *DsQq* model
  - $\alpha = 1, d : B^{\alpha} = \sum_i g_i^{\alpha}$.
  - coupling is done though the equilibrium.
  - extension to more complex MHD models

# A (too) general approach : $\Pi_k(DdQq_k)$ schemes

## Generic Multi-LBM Scheme building

For each $k = 1, N$ consider a $DsQq_k$ model given by

- $q_k$ velocities $v_{k,i}$
- a $q \times q$ invertible square matrix $P_k$ mapping micro/macro variables $m = Pf$
- a subset of $1 \leq c_k \leq q_k - 1$ conserved variables.
- $q$ equilibrium functions $f_i^{eq}$ ( or equivalently $m_i^{eq} = [Pf_{eq}]_i$)

In moment space we have

$$\partial_t m_k + \nabla \cdot \left[ P_k diag[v_k] P_k^{-1} m_k \right] = \frac{1}{\epsilon} \sum_{j=1}^{N} P_k \Omega_{kj} P_j^{-1} (m_j^{eq} - m_j)$$

With $\Omega$ the linear relaxation matrix made of $q_k \times q_j$ blocks. $\Omega$ may depend on conserved variables.

- limit system yields $C = \sum_k c_k$ equations on the conserved variables.
- diffusive terms are shaped by equilibrium and the structure of $\Omega$.

# $(DdQ(d+1))^n$ schemes

Bibliography [Gra14]

## Target problem

$$\partial_t w_k + \nabla \cdot \mathbf{\Phi}_k(w) = \mathcal{D}_k(w), k = 1, n$$

For each $k$ one conserved quantity and $d$ flux components : $d+1$ scalar fields

## $(DdQ(d+1)^n$ scheme building

For each $k = 1, n$ we consider the "same" $DdQ(d+1)$ model given by

- $d+1$ velocities $v_i$ forming a simplex, $\sum_i \mathbf{v}_i = 0$
- 1 conserved quantity $m_{k,0} = \sum_i f_{k,i}$. $(w = [m_{1,0}, m_{2,0} .... m_{n,0}])$
- $d$ non-conserved quantities are the $m_{k,\alpha} = \sum_i f_{k,i} v_i^{\alpha}, \alpha = 1, d$
- $q = d+1$ equilibrium functions $f_i^{eq}$ obtained by solving $w_k = \sum_i f_{i,k}^{eq}(w)$ and $F_k^{\alpha}(w) = \sum_i f_{i,k}^{eq}(w) v_i^{\alpha}$.
- dissipative provided $|\mathbf{v}_i| > \sup\{|\lambda|, \lambda \in spec(d_w \mathbf{\Phi})\}$

Section 3

# Implementation of an implicit DG-LBM solver

# Back to our problem

## Guiding principle

- Replace strongly coupled non-linear hyperbolic system by a (larger) set of more loosely coupled ones.
- in essence : split spatial coupling / inter-variable nonlinear coupling.
- compensate for larger problem size by efficient parallelization.

## Requisites

- unstructured meshes to handle complex geometry
- no CFL
- high order in space and time.

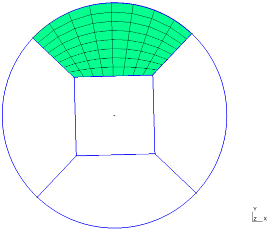
## Project guidelines

- unstructured mesh : DG space discretization ( h,p refinement, locality)
- no CFL? $\rightarrow$ implicit schemes
- control diffusive terms : high space and time order + relaxation tweaking.

# DG - Implicit upwind transport scheme 1

We consider a coarse mesh made of hexahedral curved macrocells

- Each macrocell is itself split into smaller subcells of size $h$.
- In each subcell $L$ we consider polynomial basis functions $\psi_k^L$ of degree $p$.
- Expansion on the polynomial basis: discontinuous approximation of $f$.



$$f(x, v, p\Delta t) \simeq f_L^p(x, v) = \sum_k f_{L,k}^p(v)\psi_k^L(x), \quad x \in L.$$

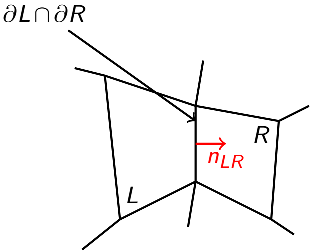
# DG - Implicit upwind transport scheme 2

## Implicit DG approximation scheme

$\forall L, \forall k$

$$\int_L \frac{f_L^p - f_L^{p-1}}{\Delta t}\psi_k^L - \int_L v\cdot\nabla\psi_k^L f_L^p + \int_{\partial L}\left(v\cdot n^+ f_L^p + v\cdot n^- f_R^p\right)\psi_k^L = 0.$$

- time step index: $p$
- $R$ denotes the neighbor cells along $\partial L$.
- $v\cdot n^+ = \max(v\cdot n, 0)$,
  $v\cdot n^- = \min(v\cdot n, 0)$.
- $n_{LR}$ is the unit normal vector on $\partial L$ oriented from $L$ to $R$.




## Features

- implicit scheme , unconditionally stable , $(h,p)$ refinement
- requires a priori the resolution of a large linear system for each $v$.

# Getting high order in time : symmetric splitting

Bibliography: [MQ02]

**Example**

- D2Q9 model ,Euler stationary state in a constant gravity field $\mathbf{g} = g\mathbf{e}_y$.
- Analytical solution $\rho = \rho_0 e^{-gy/T}$

**Splitting schemes made of symmetric building blocks**

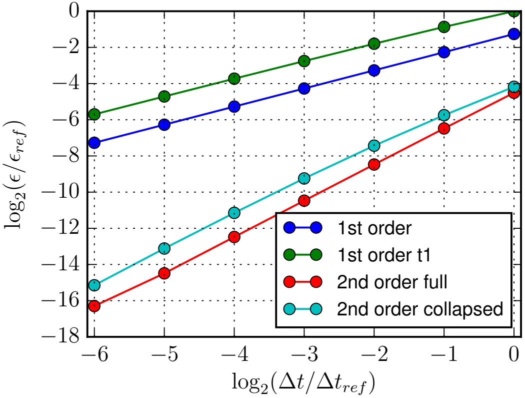
All steps implemented as $\theta$ weighted schemes.
$\theta = 0$ explicit , $\theta = 1$ implicit ; $\theta = 1/2$ (Crank-Nicolson) $\rightarrow$ symmetric

- Transport (T)
- Macroscopic source (S) (gravity)
- BGK Relaxation (R).

| | | | | | |
|---|---|---|---|---|---|
| $1^{st}$ order | $T(\Delta t)$ | $S(\Delta t)$ | $R(\Delta t)$ | | |
| $2^{nd}$ order | $T(\Delta t/2)$ | $R(\Delta t/2)$ | $S(\Delta t)$ | $R(\Delta t/2)$ | $T(\Delta t/2)$ |
| $2^{nd}$ order collapsed | $R(\Delta t/2)$ | $S(\Delta t)$ | $R(\Delta t/2)$ | $T(\Delta t)$ | |

# Euler gravity stationary

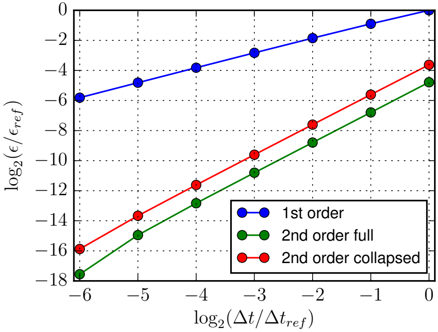
$\Delta t$ Convergence of $L2$ error on macroscopic data wrt analytical solution.




First order t1 : first order splitting with fully implicit $\theta = 1$ blocks.
First order : first order splitting with $\theta = 1/2$ second order symmetric blocks.

# Euler Gaussian pulse

- D2Q9 model on a square : 8x8 elements, 10x10 subcells, $3^{rd}$ order
- Initial condition : narrow gaussian density bump
  $\rho = 1 + 0.1\exp(-40*(x^2+y^2)$.
- Convergence evaluated from highly time-resolved solution.



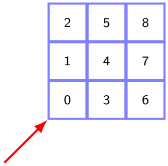
# The benefits of upwinding

## upwind flux → data dependencies follow the (constant) velocity

- transport operator can be cast into Block Triangular Form (BTF) by appropriate data renumbering.
- inversion : BTF + inversion of diagonal blocks.
- data blocks at the subcell scale : too small for efficient parallelism.

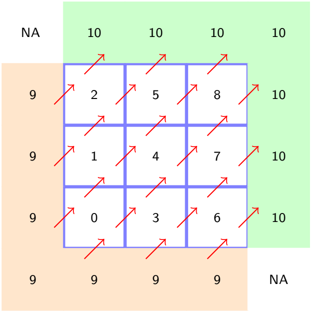
## Coarse grain block structure at macrocell level

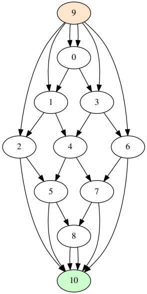
- $L$ is *upwind* with respect to $R$ if $v \cdot n_{LR} > 0$ on $\partial L \cap \partial R$.
- In a cell $L$, the solution depends only on the values of $f$ in the upwind macrocells.



# Dependency graph

For a given velocity $v$ we can build a dependency graph. Vertices are associated to macrocells and edges to macrocells interfaces or boundaries. We consider two fictitious additional vertices: the "upwind" vertex and the "downwind" vertex.



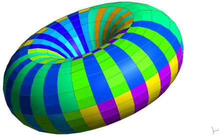




- The dependency graph yields a coarse block triangular ordering
- the local system in each macrocell is solved "on the fly" using the KLU library.
- no need to assemble, store, and factorize the global system !

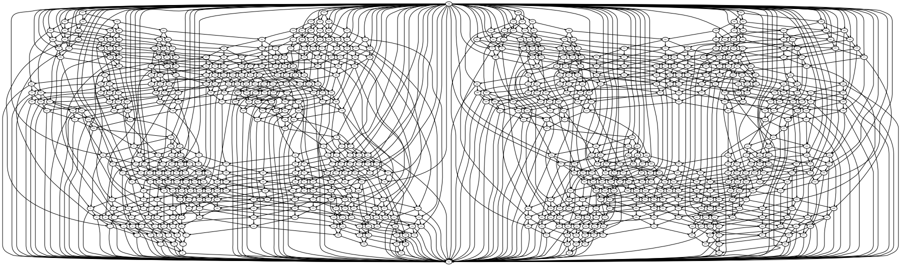
# Transport solver parallelism

- ideal across velocities (uncoupled)
- across macrocells : can be high but load imbalance
- realistic mesh : complex to manage...

Toroidal mesh - 720 macrocells



Toroidal mesh - transport graph for $(1, 0, 0)$ velocity.



We need smart task scheduling

# Kirsch : Task-Based parallel DG-LBM solver

## Here comes StaPU

- StarPU is a task-based scheduling library developed at Inria Bordeaux [AAF+12]: `http://starpu.gforge.inria.fr`
- Task description : codelets, inputs (R), outputs (W or RW).
- The user submits tasks in a correct sequential order.
- StarPU schedules the tasks in parallel if possible.
- MPI extension easy : dispatch data and declare owner process : communications handled transparently.

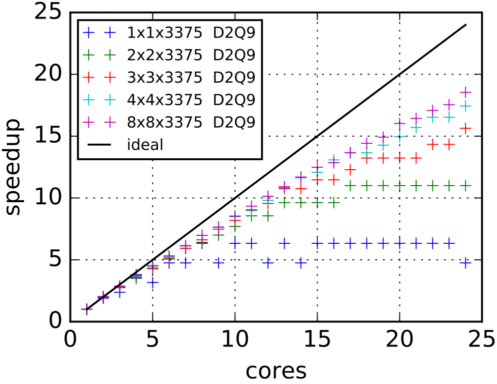
## SCHNAPS + StaRPU + LBM = KIRSCH

- starting point SCHNAPS : general DG explicit solver.
- StarPU + Optimization for Kinetic LBM-like schemes
- KIRSCH : KInetic Representation for SCHnaps

# D2Q9 multithread performance

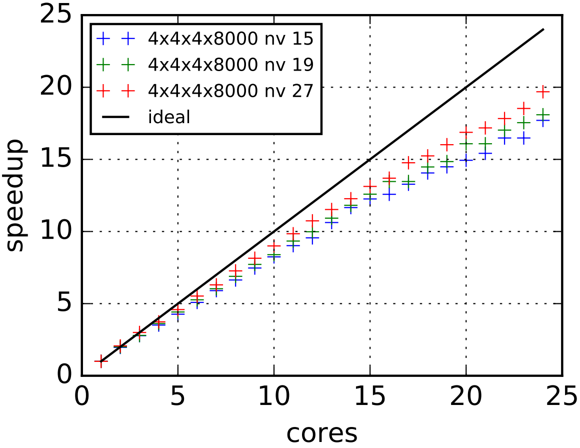
Full D2Q9 scheme on square grids. Constant dof number per macrocell. Number $N$ of macrocells $N$ from 1 to $64 = 8x8$.

- for 1 macrocell : saturation at $n_{core} = n_v$. This is expected.
- efficiency grows with $N$ due to topological parallelism.



# D3Q* multithread performance

D3Q15,D3Q19,D3Q27 models on a cube with 4x4x4 elements and 8000 dof per elements with eager scheduler.

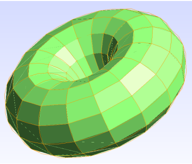
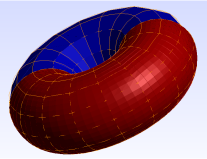
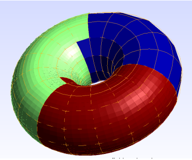
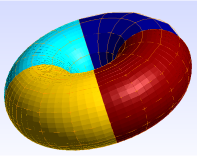


# MPI Scaling : D3Q15 in a torus

Toroidal mesh : 720 macroelements x 3335 dof
2064 interfaces - 192 boundary faces

Wall time in sec for 100 interations.

| Nthreads/Nmpi | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| 14 | 6862 | 2772 | 1491 | 1014 |

# Section 4

## Conclusion and prospects

# Conclusions and prospects

## Current state

- DG-LBM parallel solver
- $2nd$ order in time
- validation tests on standard $2D$ LBM-BGK models (Fluids, 2D MHD)
- good MPI/Multithreaded scaling in both $2D$ and $3D$.

## Next steps

- $(DqQd+1)^n$ approach is appealing : stable and generic.
- optimization : Transport Tasks Optimization / GPU codelets
- higher order in time (composition [CCDV09] , complex time steps) :mitigate diffusion
- validate $2D-3D$ MHD models.
- benchmark wrt JOREK.

# Bibliography I


[AAF+12] Cédric Augonnet, Olivier Aumage, Nathalie Furmento, Raymond Namyst, and Samuel Thibault.
StarPU-MPI: Task Programming over Clusters of Machines Enhanced with Accelerators.
In Siegfried Benkner Jesper Larsson Träff and Jack Dongarra, editors, *EuroMPI 2012*, volume 7490 of *LNCS*. Springer, September 2012. Poster Session.

[CCDV09] François Castella, Philippe Chartier, Stéphane Descombes, and Gilles Vilmart.
Splitting methods with complex times for parabolic equations.
*BIT Numerical Mathematics*, 49(3):487–508, 2009.

[CD98] Shiyi Chen and Gary D Doolen.
Lattice boltzmann method for fluid flows.
*Annual review of fluid mechanics*, 30(1):329–364, 1998.

[Del02] Paul J Dellar.
Lattice kinetic schemes for magnetohydrodynamics.
*Journal of Computational Physics*, 179(1):95–126, 2002.

[Dub13] François Dubois.
Stable lattice boltzmann schemes with a dual entropy approach for monodimensional nonlinear waves.
*Computers & Mathematics with Applications*, 65(2):142–159, 2013.

[Gra14] Benjamin Graille.
Approximation of mono-dimensional hyperbolic systems: A lattice boltzmann scheme as a relaxation method.
*Journal of Computational Physics*, 266:74–88, 2014.

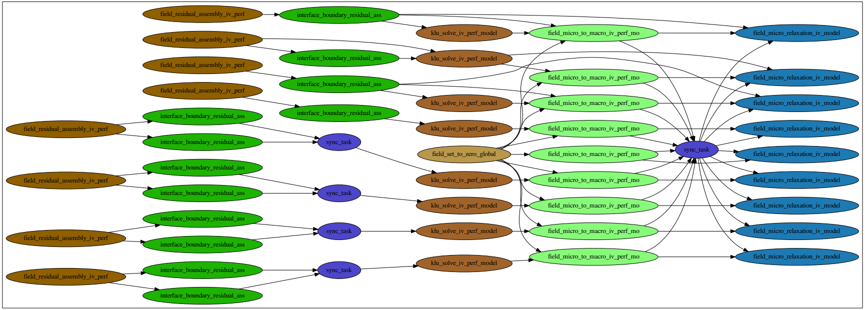
[JX95] Shi Jin and Zhouping Xin.
The relaxation schemes for systems of conservation laws in arbitrary space dimensions.
*Communications on pure and applied mathematics*, 48(3):235–276, 1995.

[MQ02] Robert I McLachlan and G Reinout W Quispel.
Splitting methods.
*Acta Numerica*, 11:341–434, 2002.

# What StarPU does for us

Task graph for 2DQ9 model

- a single 2D macrocell
- a single time step of the first order scheme ( T + R)




Pretty simple...

# What StarPU does for us

- 4 macrocells in a 2D square
- a single time step of the first order scheme ( T + R)