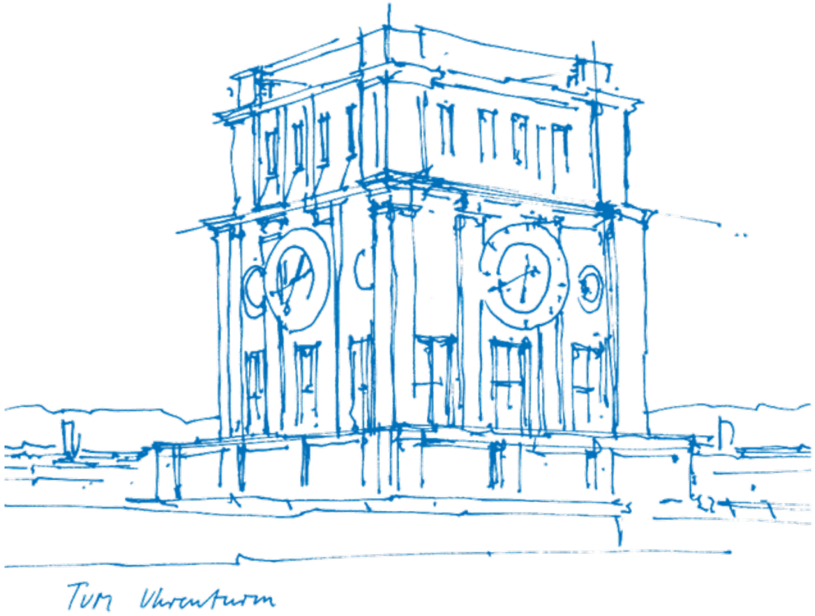
Communication-Avoiding Numerical Schemes

Natalia Saiapova

Technische Universität München

1. Februar 2017



Contents

Problem Statement

- One-dimensional conservation law for $u(t, x)$:

$$\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0, \quad t \in [0, T], \quad x \in \Omega$$

Problem Statement

- One-dimensional conservation law for $u(t,x)$:

$$\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0, \quad t \in [0,T], \quad x \in \Omega$$

- flux function is linear: $f(u(x,t)) = cu(x,t), \quad c \in \mathbb{R}$

Problem Statement

- One-dimensional conservation law for $u(t,x)$:

$$\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0, \quad t \in [0,T], \quad x \in \Omega$$

- flux function is linear: $f(u(x,t)) = cu(x,t), \quad c \in \mathbb{R}$
- initial conditions:

$$u(0,x) = u_0(x)$$

- periodic boundary conditions

Problem Statement

- One-dimensional conservation law for $u(t,x)$:

$$\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0, \quad t \in [0,T], \quad x \in \Omega$$

- flux function is linear: $f(u(x,t)) = cu(x,t), \quad c \in \mathbb{R}$
- initial conditions:

$$u(0,x) = u_0(x)$$

- periodic boundary conditions
- residual:

$$\mathcal{R}(x,t) = \frac{\partial u_h}{\partial t} + \frac{\partial f(u_h)}{\partial x}$$

Problem Statement

- One-dimensional conservation law for $u(t,x)$:

$$\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0, \quad t \in [0,T], \quad x \in \Omega$$

- flux function is linear: $f(u(x,t)) = cu(x,t), \quad c \in \mathbb{R}$
- initial conditions:

$$u(0,x) = u_0(x)$$

- periodic boundary conditions
- residual:

$$\mathcal{R}(x,t) = \frac{\partial u_h}{\partial t} + \frac{\partial f(u_h)}{\partial x}$$

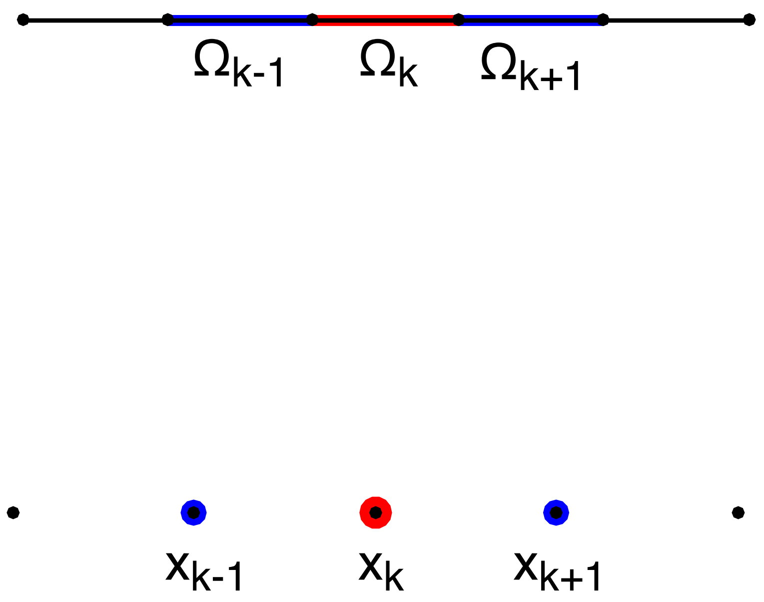
- $\Omega = \bigcup_{k=1}^{K} \Omega_k, \quad \Omega_k = [x_k, x_{k+1}]$

Our Goals

- Efficiency
- high-order accuracy
- complex geometries
- parallelization

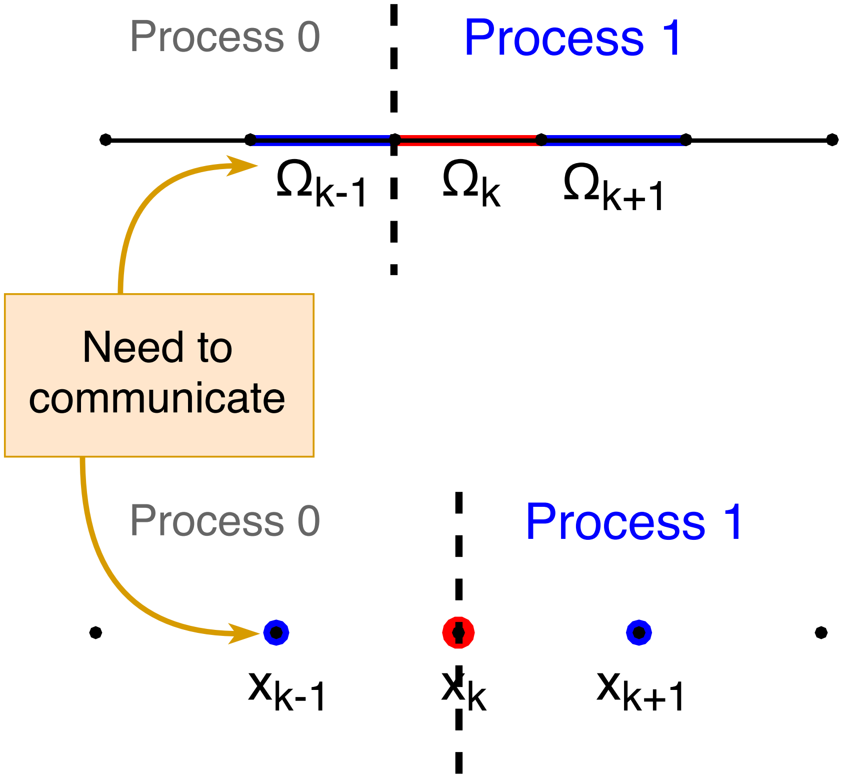
Our Goals

- Efficiency
- high-order accuracy
- complex geometries
- parallelization



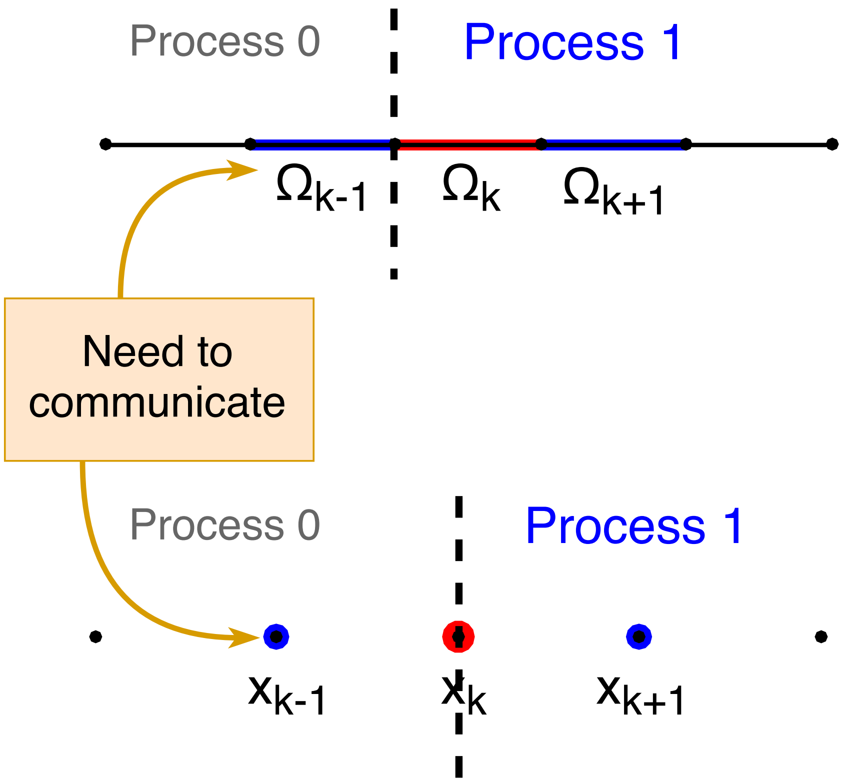
Our Goals

- Efficiency
- high-order accuracy
- complex geometries
- parallelization



Our Goals

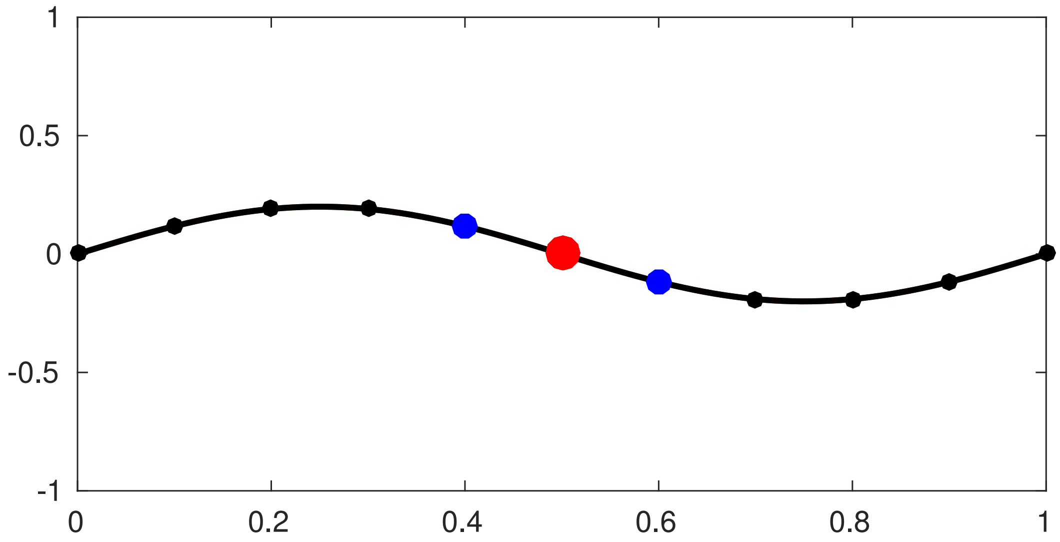
- Efficiency
- high-order accuracy
- complex geometries
- parallelization



To avoid communication a stencil needs to be small!

Finite Difference Method

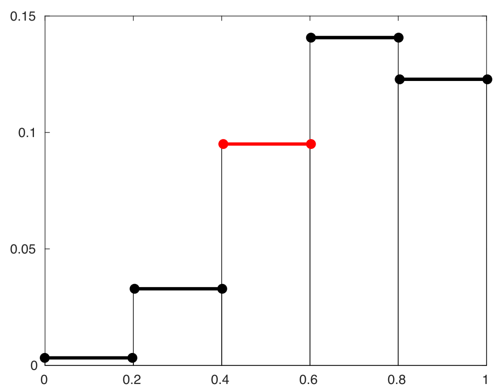
- ☺ Straightforward implementation through Taylor series
- ☹ Stencil is not invariant to increasing the accuracy order



Finite Volume Method

Requirement for the cellwise average of the residual:

$$\frac{1}{h_k} \int_{\Omega_k} \mathcal{R}_k(x,t)\,dx = 0$$



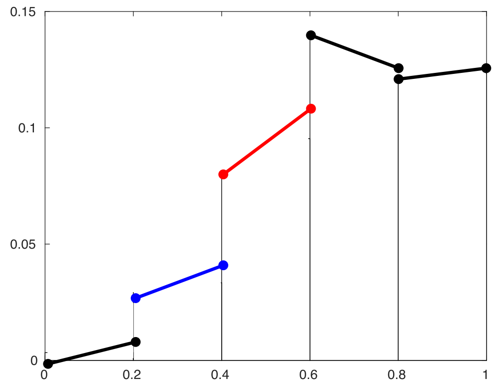
first order: u_h is piecewise constant

$u_h^k = a^k$

Finite Volume Method

Requirement for the cellwise average of the residual:

$$\frac{1}{h_k} \int_{\Omega_k} \mathcal{R}_k(x,t)dx = 0$$



first order: u_h is piecewise constant

$$u_h^k = a^k$$

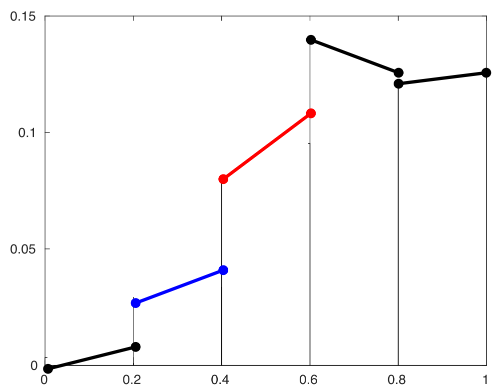
second order: u_h is piecewise linear

$$u_h^k = a^k x + b^k$$

Finite Volume Method

Requirement for the cellwise average of the residual:

$$\frac{1}{h_k} \int_{\Omega_k} \mathcal{R}_k(x,t)dx = 0$$



first order: u_h is piecewise constant

$u_h^k = a^k$

second order: u_h is piecewise linear

$u_h^k = a^k x + b^k$

☹ stencil is not invariant to increasing the order

Finite Element Method

Solution $u_h \in V_h$ (linear case):

$$u_h(t,x) = \sum_{j=1}^{K} a_j(t)\psi_j(x), \; \psi_j \text{ is piecewise linear}$$

residual has to be orthogonal to $\forall \psi_h \in V_h$:

$$\int_\Omega \left(\frac{\partial u_h}{\partial t} + \frac{\partial f_h}{\partial x} \right) \psi_h(x) dx = 0$$

$$M\boldsymbol{a}_t - cS\boldsymbol{a} = 0,$$

matrices M and S have dimensions $K \times K$

Finite Element Method

Solution $u_h \in V_h$ (linear case):

$$u_h(t,x) = \sum_{j=1}^{K} a_j(t)\psi_j(x),\ \psi_j \text{ is piecewise linear}$$

residual has to be orthogonal to $\forall \psi_h \in V_h$:

$$\int_\Omega \left(\frac{\partial u_h}{\partial t} + \frac{\partial f_h}{\partial x} \right) \psi_h(x) dx = 0$$

$$M\boldsymbol{a}_t - cS\boldsymbol{a} = 0,$$

matrices M and S have dimensions $K \times K$

☺ higher order through additional degrees of freedom

☹ global matrix needs to be solved at every timestep

What do we want?

- $$u_h \in V_h, \quad V_h = \bigoplus_{k=1}^{K} V_h^k, \quad V_h^k = \text{span}\left\{\psi_n^k\right\}_{n=0}^{n=N}$$

- for each element solution has a form:

$$u_h^k(t,x) = \sum_{j=0}^{N} a_j^k(t)\psi_j^k(x)$$

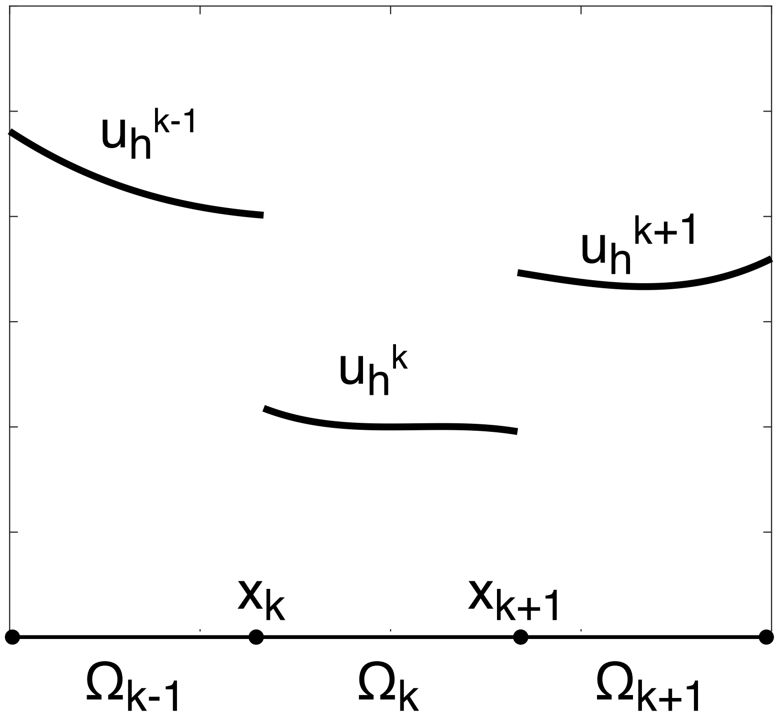
- residual has to be orthogonal to every function in V_h^k:

$$\int_{\Omega_k} \left(\frac{\partial u_h^k}{\partial t} + \frac{\partial f(u_h^k)}{\partial x}\right)\psi_i^k(x)dx = 0, \; 0 \leq i \leq N$$

Discontinuous Galerkin FEM

Divergence theorem:

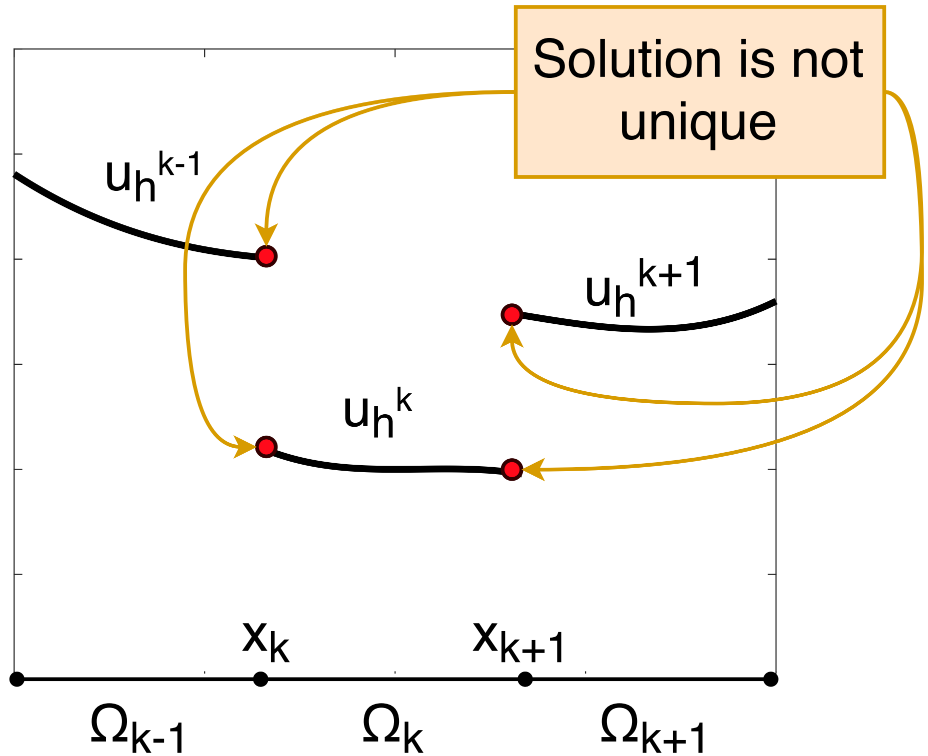
$$\int_{\Omega_k} \frac{\partial u_h}{\partial t}\psi_i^k dx - \int_{\Omega_k} f(u_h)\frac{\partial \psi_i^k}{\partial x} dx + \int_{\partial\Omega_k} f(u_h)(\psi_i^k \cdot n) ds = 0$$



Discontinuous Galerkin FEM

Divergence theorem:

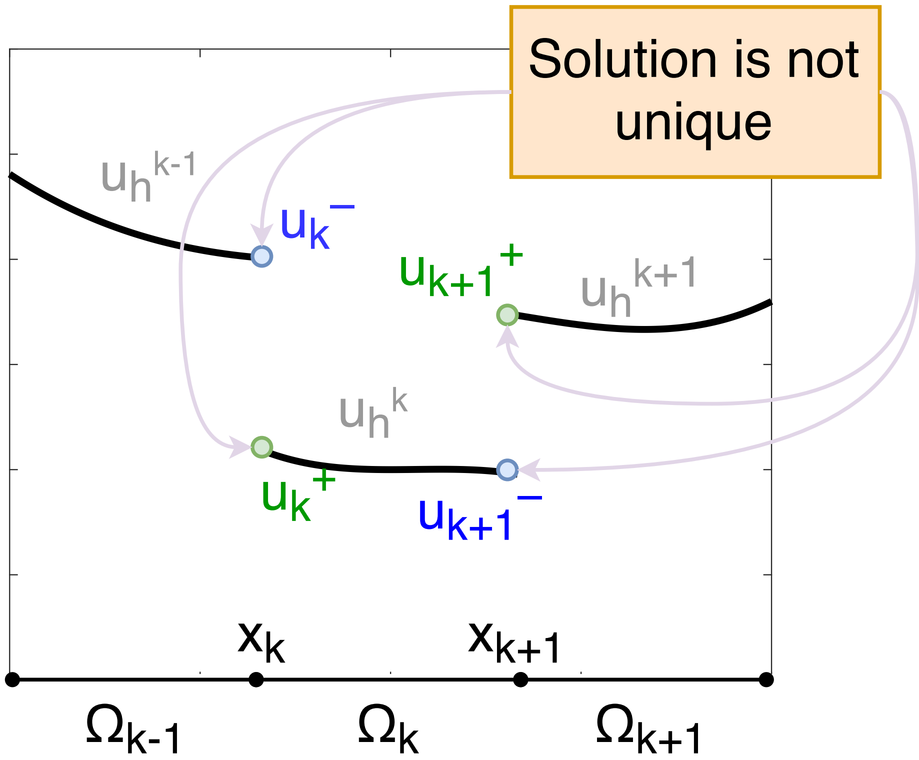
$$\int_{\Omega_k} \frac{\partial u_h}{\partial t} \psi_i^k dx - \int_{\Omega_k} f(u_h) \frac{\partial \psi_i^k}{\partial x} dx + \int_{\partial \Omega_k} f(u_h)(\psi_i^k \cdot n) ds = 0$$



Discontinuous Galerkin FEM

Divergence theorem:

$$\int_{\Omega_k} \frac{\partial u_h}{\partial t}\psi_i^k dx - \int_{\Omega_k} f(u_h)\frac{\partial \psi_i^k}{\partial x}dx + \int_{\partial\Omega_k} f(u_h)(\psi_i^k \cdot n)ds = 0$$



Numerical flux:

$$\hat{f}(u^-, u^+) = \begin{cases} cu^-, \text{ if } c > 0, \\ cu^+, \text{ if } c < 0 \end{cases}$$

Semidiscrete Formulation

On every element:

$$M^k (\boldsymbol{a}^k)_t - cS^k \boldsymbol{a}^k + \boldsymbol{b}^k = 0,$$

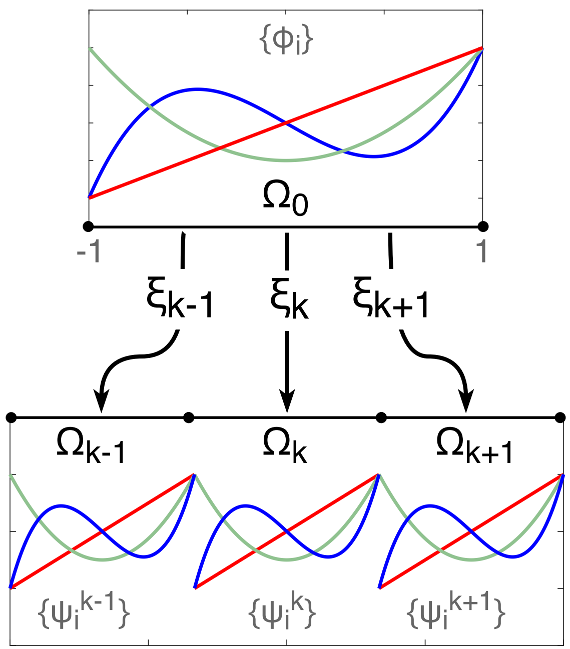
where

$$M_{ij}^k = \int\limits_{\Omega_k} \psi_i^k \psi_j^k dx, \quad S_{ij}^k = \int\limits_{\Omega_k} \psi_j^k \frac{\partial \psi_i^k}{\partial x} dx$$

- M^k – mass matrix, $(N+1) \times (N+1)$
- S^k – stiffness matrix, $(N+1) \times (N+1)$
- $\boldsymbol{a}^k$ – vector of the unknown coefficients
- $\boldsymbol{b}^k$ – flux term evaluated on the interface

Canonical Element

Parameterize with a canonical element $\Omega_0 = [-1, 1]$:



$$u_h^k(t,x) = \sum_{j=0}^{N} a_j^k(t)\phi_j(x)$$

$$J^k M \boldsymbol{a}_t^k - cS\boldsymbol{a}^k + \boldsymbol{b}^k = 0$$

$$M_{ij} = \int_{\Omega_0} \phi_i \phi_j dx, \quad S_{ij} = \int_{\Omega_0} \phi_j \frac{\partial \phi_i}{\partial x} dx,$$

$$J^k = \frac{\partial \xi_k(r)}{\partial r}$$

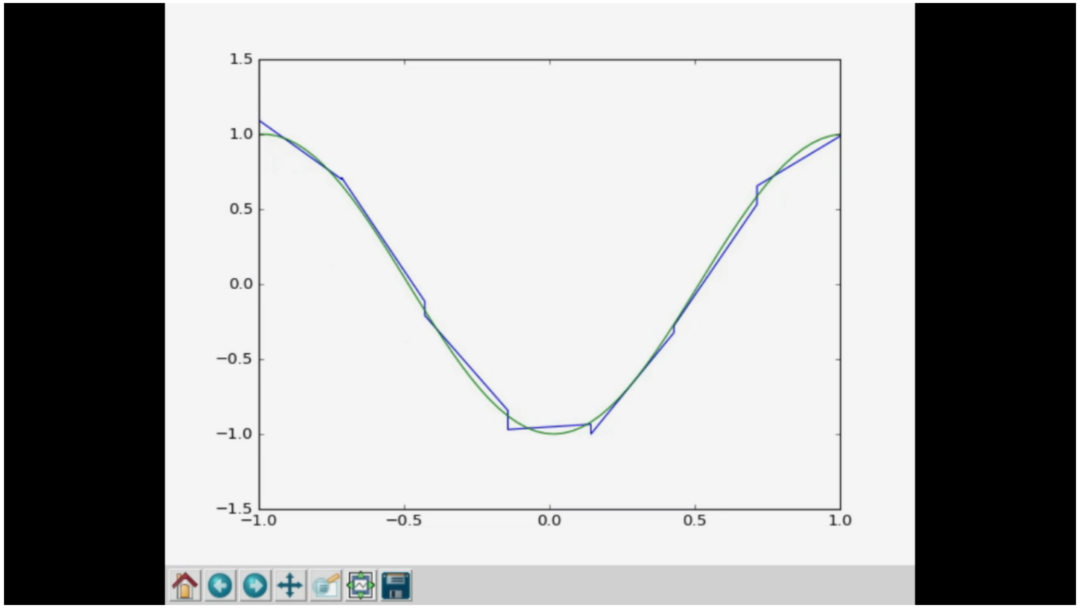
Classical choice of the basis:
Legendre polynomials

Time Integration

At the end:

$$\boldsymbol{a}_t^k = M^{-1}(J^k)^{-1}(cS\boldsymbol{a}^k - \boldsymbol{b}_k)$$

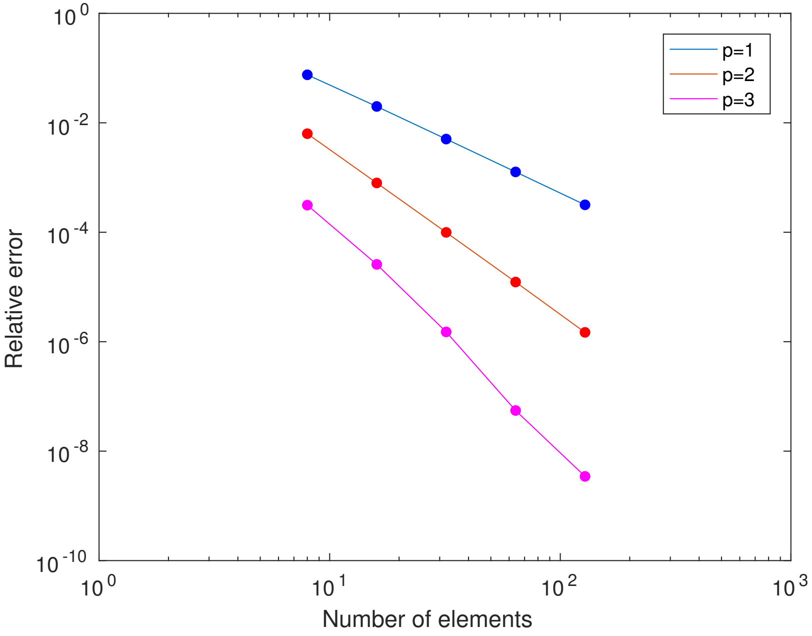
We can solve it with RK4:



7 elements, polynomials of degree N=1

Accuracy

- With polynomials of degree p accuracy order is at least $p+1/2$ [1986]
- In practice, the order $p+1$ is usually observed



Sequential Algorithm Outline

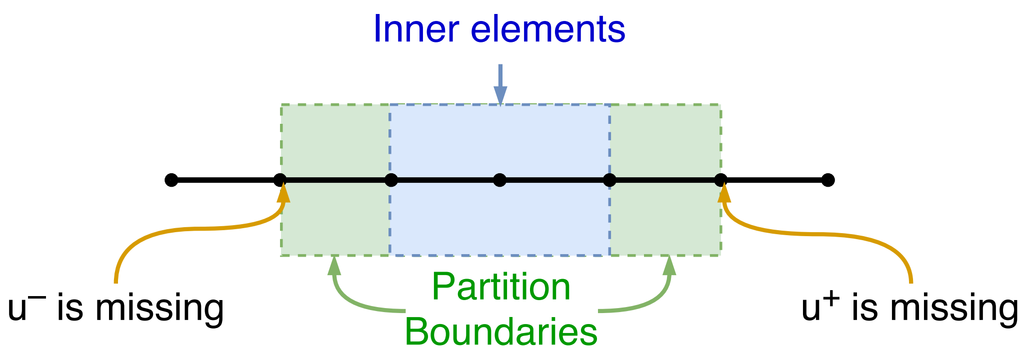
$$\boldsymbol{a}_t^k = M^{-1}(J^k)^{-1}(cS\boldsymbol{a}^k - \boldsymbol{b}_k)$$

At every timestep, for every element do:

1. Compute u^- and u^+ from coefficients $\boldsymbol{a_k}$ for every node x_k
2. Compute $\boldsymbol{b}^k$ for every element
3. Compute the new $\boldsymbol{a}$ with time integration scheme

Parallelization Strategy

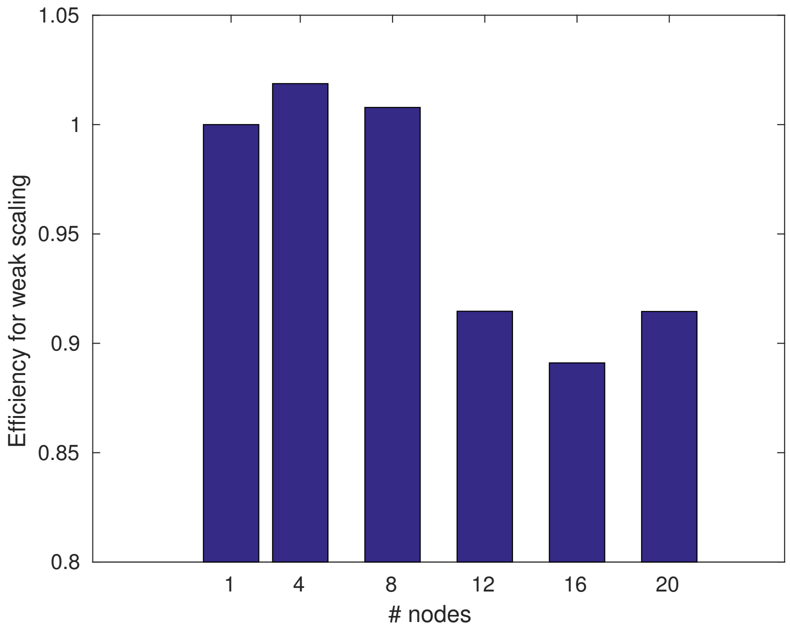
$$\boldsymbol{a}_t^k = M^{-1}(J^k)^{-1}(cS\boldsymbol{a}^k - \boldsymbol{b}_k)$$



1. Compute the solution u_h on the partition boundaries
2. Non-blocking send of these values to the neighboring partitions
3. Compute $\boldsymbol{b}_k$ for inner elements
4. Compute the new $\boldsymbol{a}_k$ for inner elements
5. Receive the boundary values from the neighboring partitions
6. Compute $\boldsymbol{b}_k$ fluxes and coefficients $\boldsymbol{a}_k$ on boundaries

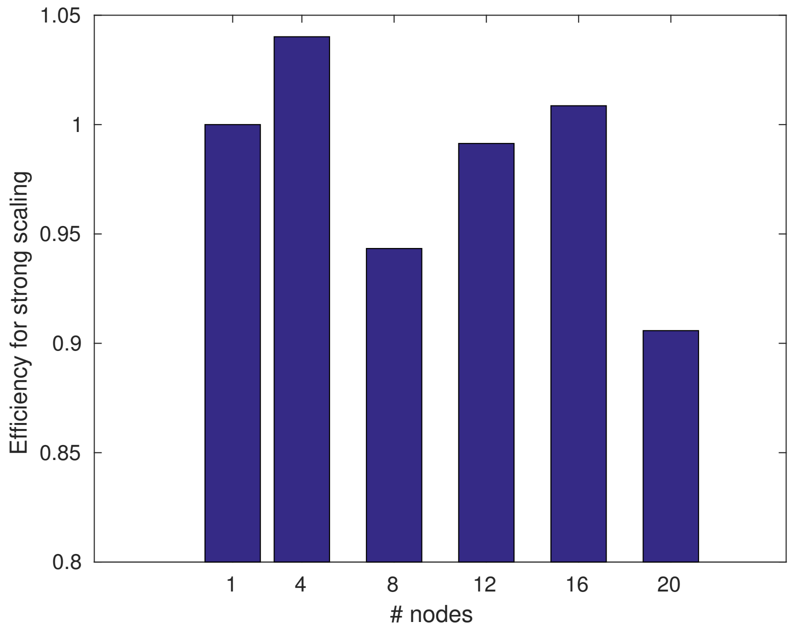
Parallel Efficiency

- Weak scaling



4000 elements per task

- Strong scaling



$2.26 \cdot 10^6$ elements in total

SuperMUC, Haswell nodes (2 phase).
28 tasks per node.

Further Motivation

- The SeisSol software for earthquake simulations, shows an efficiency of more than 90% on SuperMUC cluster with 9000 nodes [1]
- the framework for unsteady turbulent flow simulations has an efficiency more than 85% with only 1 element per process for 4096 processors [2]

Thank you!

Questions?

A. Heinecke, A. Breuer, S. Rettenberger, M. Bader, A.-A. Gabriel, C. Pelties, A. Bode, W. Barth, X.-K. Liao, K. Vaidyanathan, M. Smelyanskiy, and P. Dubey, “Petascale high order dynamic rupture earthquake simulations on heterogeneous supercomputers,” in *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, ser. SC ’14. Piscataway, NJ, USA: IEEE Press, 2014, pp. 3–14. [Online]. Available: https://doi.org/10.1109/SC.2014.6

F. Hindenlang, G. J. Gassner, C. Altmann, A. Beck, M. Staudenmaier, and C.-D. Munz, “Explicit discontinuous galerkin methods for unsteady problems,” *Computers & Fluids*, vol. 61, pp. 86 – 93, 2012. [Online]. Available: //www.sciencedirect.com/science/article/pii/S004579301200093X