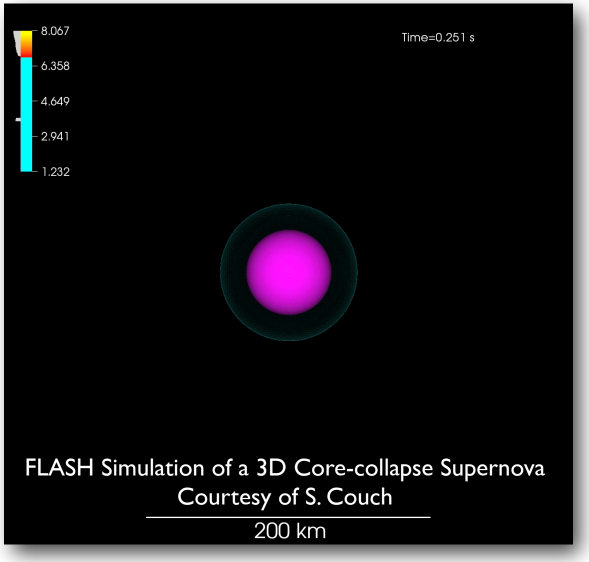
High-order Numerical Methods for Plasma Simulations On Large-Scale High-Performance Computing Architectures

Dongwook Lee
Applied Mathematics & Statistics
University of California, Santa Cruz

Collaborators: C. Graziani & P. Tzeferacos (U of Chicago)

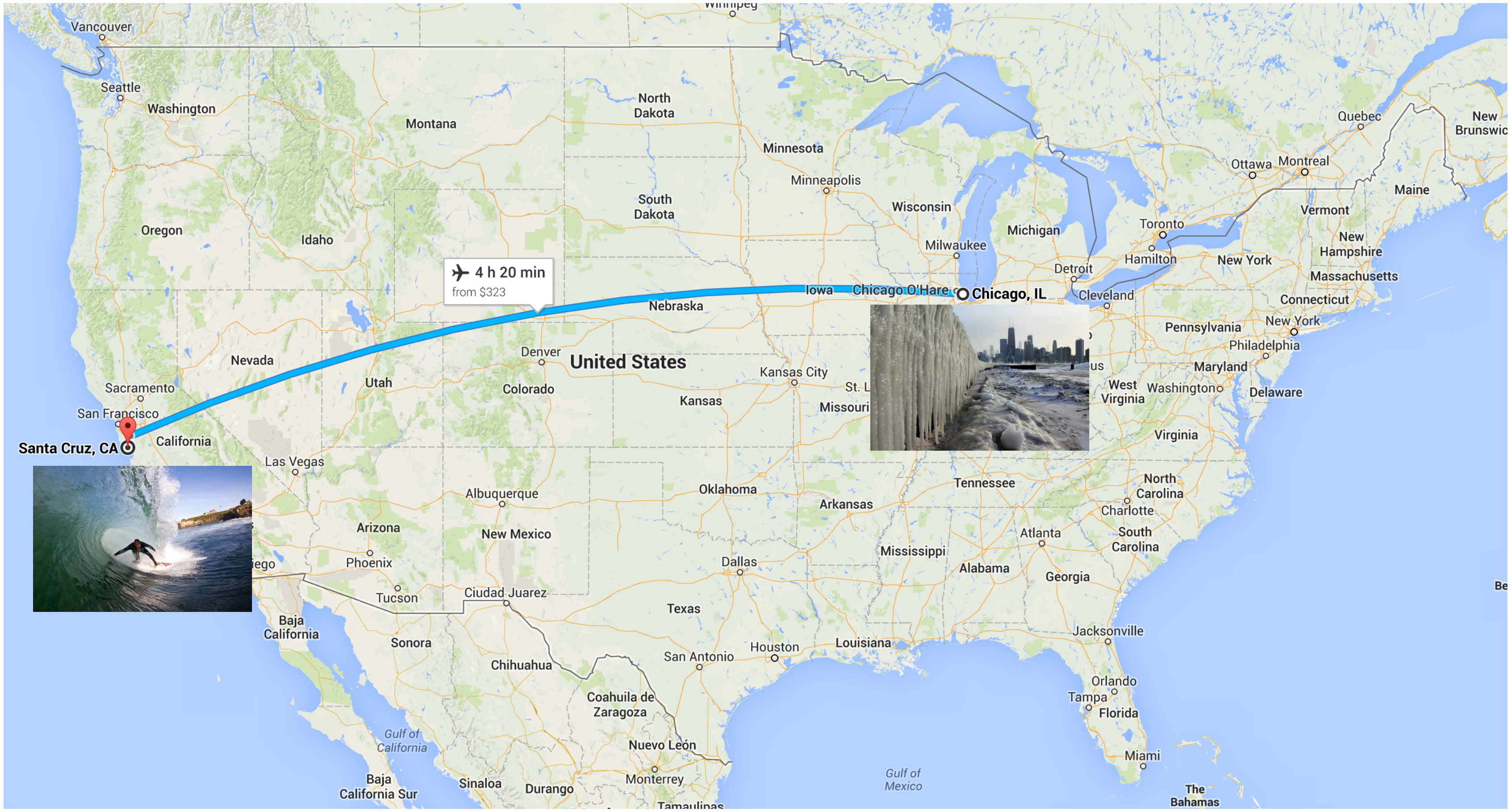
3rd ICM Theory and Computation Workshop
Niels Bohr Institute
Copenhagen
August 11, 2014

MIRA, BG/Q, Argonne National Lab
49,152 nodes, 786,432 cores



FLASH Simulation of a 3D Core-collapse Supernova
Courtesy of S. Couch

Hurtig Annoncering



Applied Mathematics and Statistics in UCSC

dlee79@ucsc.edu

ams.soe.ucsc.edu/people/dongwook

High Performance Computing (HPC)

- To solve large problems in science, engineering, or business
- Modern HPC architectures have
  - increasing number of cores
  - declining memory/core
- This trend will continue for the foreseeable future

High Performance Computing (HPC)

- This tension between computation & memory brings a paradigm shift in numerical algorithms for HPC
- To enable scientific computing on HPC architectures:
  - efficient parallel computing, (e.g., data parallelism, task parallelism, MPI, multi-threading, GPU accelerator, etc.)
  - better numerical algorithms for HPC

Numerical Algorithms for HPC

- Numerical algorithms should conform to the abundance of computing power and the scarcity of memory
- But…
  - without losing solution accuracy
  - with maintaining maximum solution stability
  - with faster convergence to "correct" solution

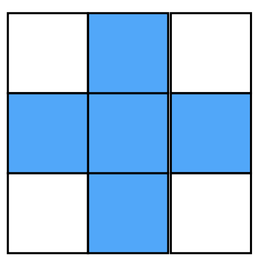
High-Order Numerical Algorithms

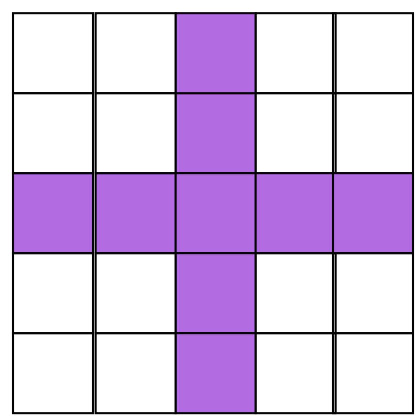
- A good solution to this is to use high-order algorithms
- They provide more accurate numerical solutions using
  - less grid points (=memory save)
  - higher-order mathematical approximations (promoting floating point operations, or computation)
  - faster convergence to solution

Traditional High-Order Schemes

- Traditional approaches to get Nth high-order schemes take (N-1)th degree polynomial for interpolation/reconstruction
  - only for normal direction (e.g., PLM, PPM, ENO, WENO, etc)
  - with monotonicity controls (e.g., slope limiters, artificial viscosity)
- High-order in FV is tricky (when compared to FD)
  - volume-averaged quantities (quadrature rules)
  - preserving conservation w/o losing accuracy
  - higher the order, larger the stencil
  - high-order temporal update (ODE solvers, e.g., RK3, RK4, etc.)

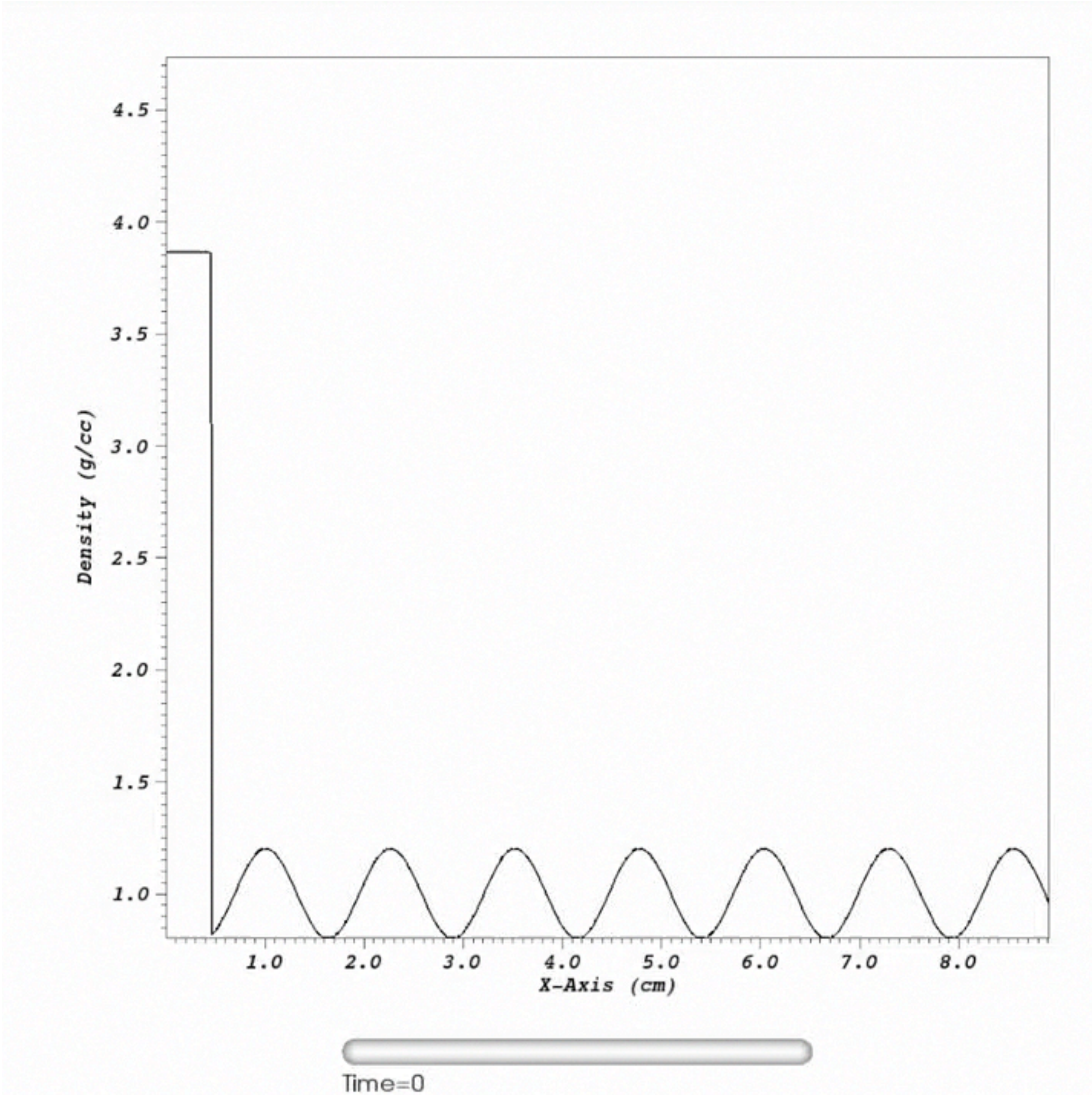
2D stencil for 2nd order PLM



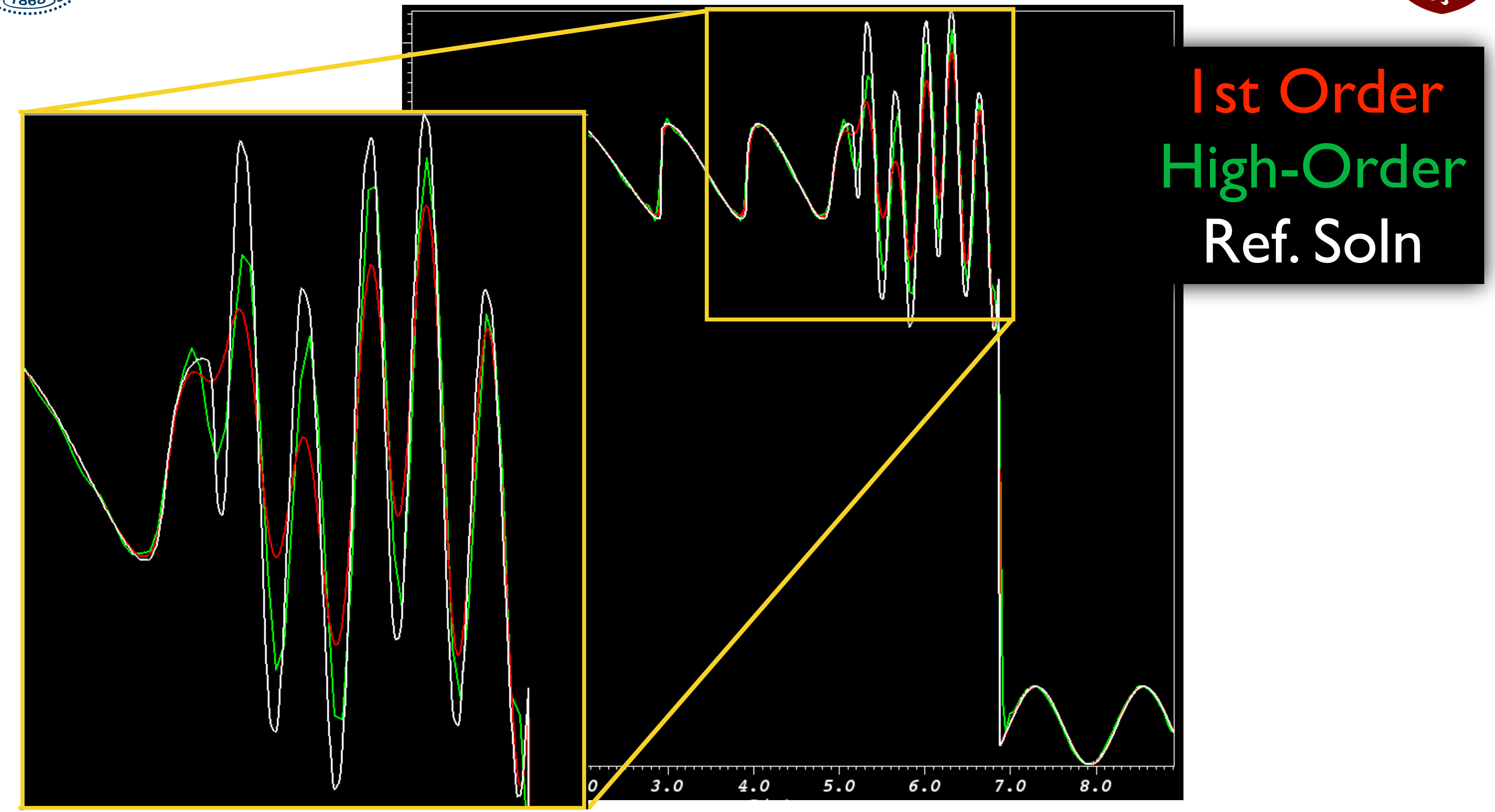


2D stencil for 3rd order PPM

Shu-Osher Problem: 1D Mach 3 Shock



Low-Order vs. High-Order

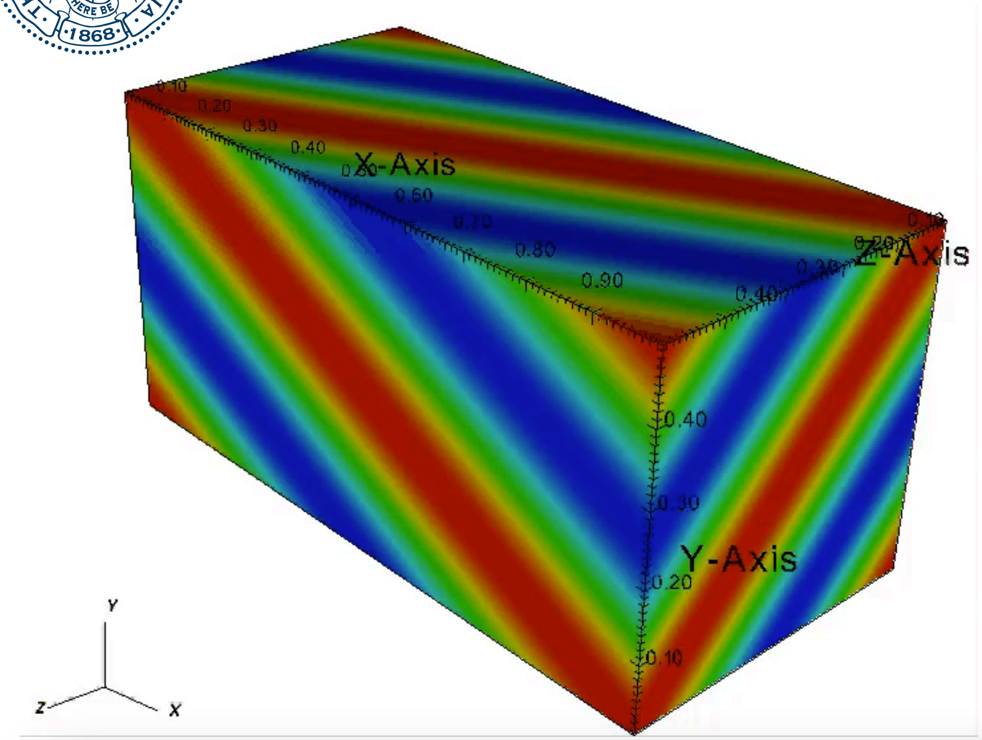


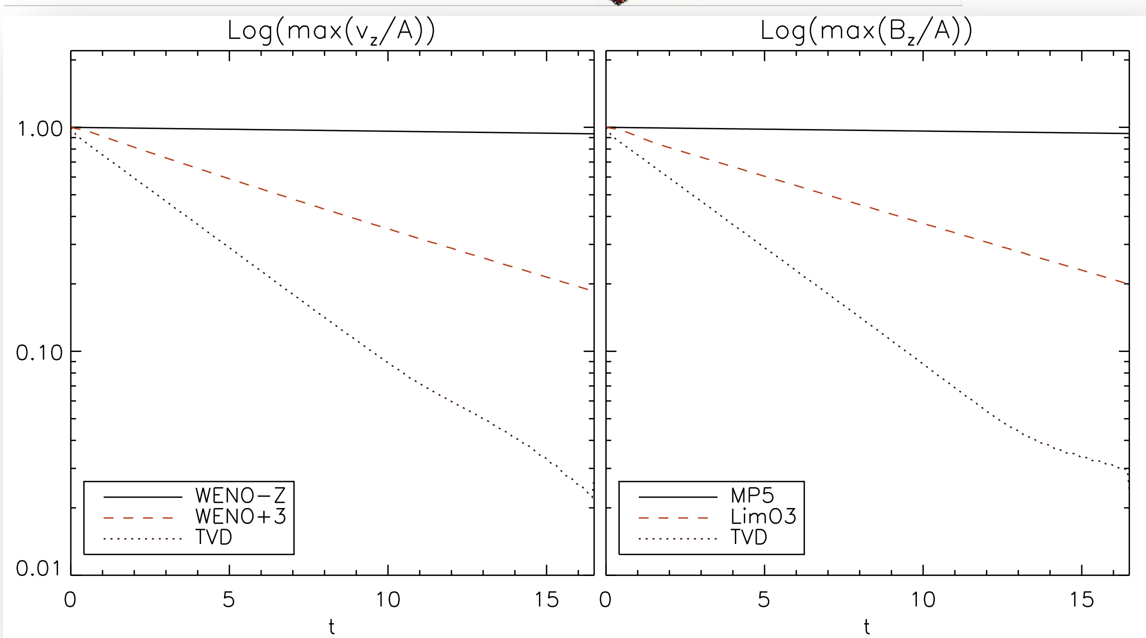
1st order: 3200 cells (50 MB), 160 sec, 3828 steps

vs.

High-order: 200 cells (10 MB), 9 sec, 266 steps

Circularly Polarized Alfven Wave (CPAW)

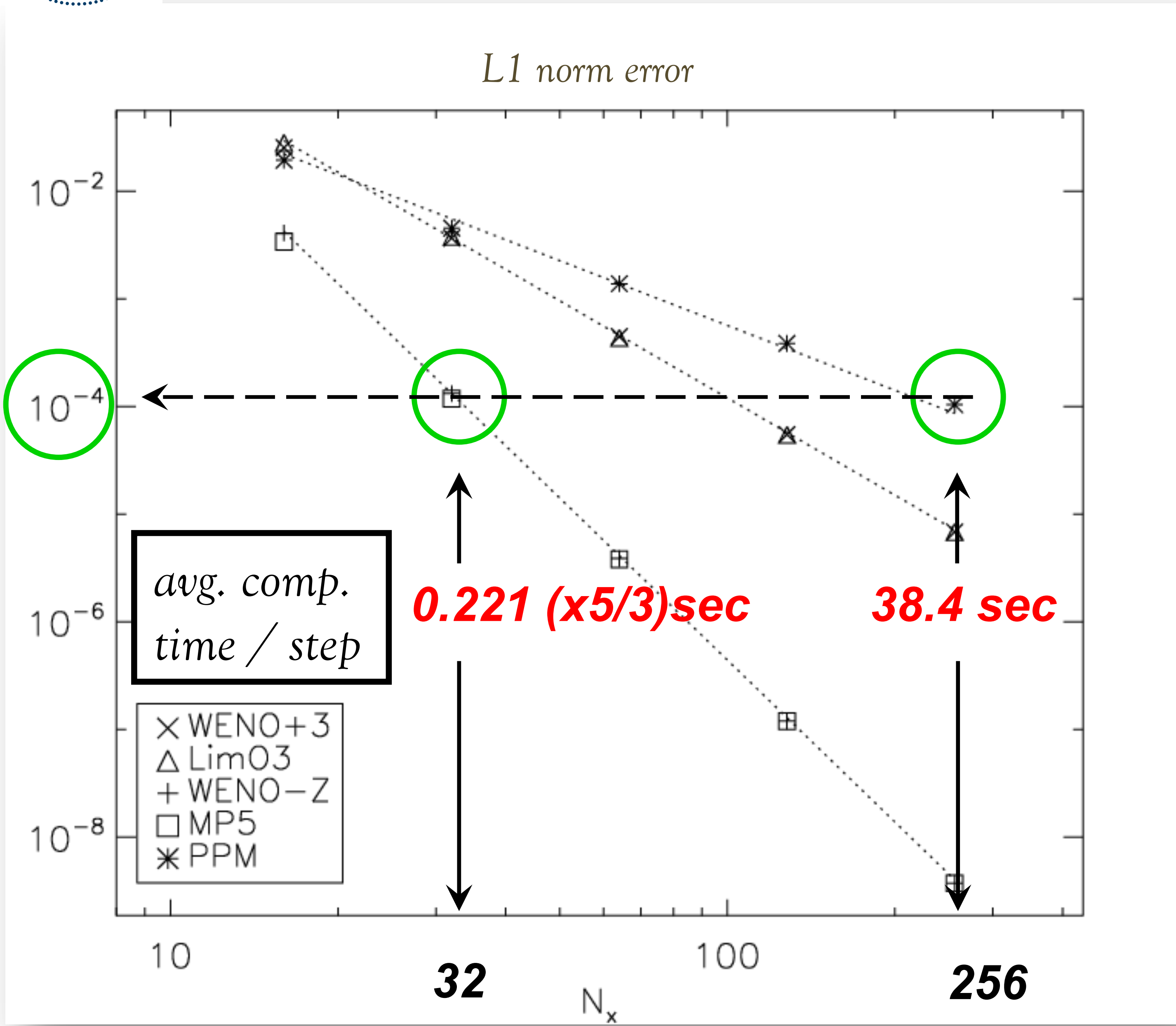




- A CPAW problem propagates smoothly varying oscillations of the transverse components of velocity and magnetic field
- The initial condition is the exact nonlinear solutions of the MHD equations
- The decay of the max of Vz and Bz is solely due to numerical dissipation: direct measurement of numerical diffusion (Ryu, Jones & Frank, ApJ, 1995; Toth 2000, Del Zanna et al. 2001; Gardiner & Stone 2005, 2008)

Source: Mignone & Tzeferacos, 2010, JCP

Performance of High-Order on CPAW

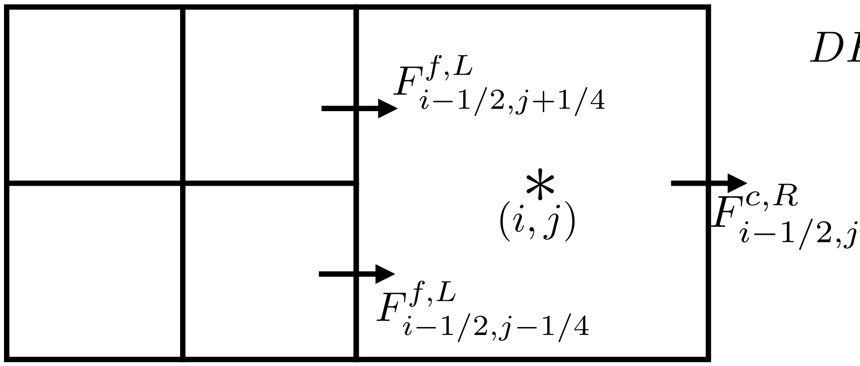


Source: Mignone & Tzeferacos, 2010, JCP

- PPM (overall 2nd order): 2h42m50s
- MP5 (5th order): 15s(x5/3)=25s

- More computational work & less memory
- Better suited for HPC
- Easier in FD; harder in FV
- High-orders schemes are better in preserving solution accuracy on AMR

Truncation Errors at Fine-Coarse Boundary



$$
\begin{aligned}
DF &= \frac{1}{\Delta x}\left[F^{c,R}_{i+1/2,j} - \frac{1}{2}\left(F^{f,L}_{i-1/2,j+1/4} + F^{f,L}_{i-1/2,j-1/4}\right)\right] \\
&= \frac{1}{\Delta x}\left[F\big((i+1/2)\Delta x, j\Delta y\big) - F\big((i-1/2)\Delta x, j\Delta y\big) + O(\Delta y^2)\right] \\
&= \frac{\partial F}{\partial x} + O(\Delta h), \text{ assuming } \Delta x \approx \Delta y (= \Delta h)
\end{aligned}
$$

$$
\boxed{\frac{\delta U}{\delta t} + \nabla \cdot F = TE = \begin{cases} O(\Delta h) & \text{at F/C boundary} \\ O(\Delta h^2) & \text{otherwise} \end{cases}}
$$

✓Any 2nd order Scheme becomes 1st order at fine-coarse boundaries

✓The deeper AMR level, the worse truncation errors accumulated and solutions will become 1st order almost everywhere if grid pattern changes frequently

✓High-order scheme is NOT just an option! (see papers by Colella et al.)

High-Order using Gaussian Processes (GP)

- Gaussian Processes (GP) are a class of a stochastic processes that yield sampling data from a function that is probabilistically constrained, but not exactly known. GP modeling is a technique from probability and statistics that is popular for nonparametric fitting of spatial data

- A GP can be thought of as a way of expressing multivariate Gaussians on spaces of functions. It is, in effect, a theory of random functions

- GP function interpolation is, effectively, Bayesian updating of a prior distribution by training the data given a new observation. The interpolation at such a new point is a "prediction" made by the updated probability distribution

- It's a great way to do function interpolation, with many advantages over polynomial/multinomial interpolation

High-Order using Gaussian Processes (GP)

- One very nice thing about GP function interpolation is that the training data and new "predictions" are NOT based directly on the function. Instead, they are linear combinations of the functions. Therefore GP interpolation is easily adjusted because linear transformations of Gaussian random variables are again Gaussian!

- A new high-order GP interpolation scheme is based on:
  - samples (i.e., volume-averaged data points) of the function
  - train the GP model on the samples by means of Bayes' theorem
  - the posterior mean function is our high-order interpolant of the unknown function

- The result is to pass from an "agnostic" prior model (a mean function and a covariance kernel) to a data-informed posterior model (an *updated* mean function and covariance kernel)

- C. Graziani, P. Tzeferacos (U of C) & D. Lee (UCSC)

Agnostic Prior Model

GP is completely defined by
(1) a mean function, and
(2) a symmetric positive-definite integral kernel K(x,y):

- Mean function

$$\bar{f}(\mathbf{x})$$

- Kernel (covariance function)

$$\left\langle \left(f(\mathbf{x}) - \bar{f}(\mathbf{x})\right)\left(f(\mathbf{y}) - \bar{f}(\mathbf{y})\right)\right\rangle = K(\mathbf{x}, \mathbf{y})$$

- Write

$$f \sim \mathcal{GP}(\bar{f}, K)$$

- The likelyhood function (the probability of f given the GP model)

$$\begin{aligned} \mathcal{L} &\equiv P(\mathbf{f}) \\ &= (2\pi)^{-N/2} \det|\mathbf{K}|^{-1/2} \exp\left[-\frac{1}{2}\left(\mathbf{f} - \bar{\mathbf{f}}\right)^T \mathbf{K}^{-1} \left(\mathbf{f} - \bar{\mathbf{f}}\right)\right] \end{aligned}$$

Data-Informed Posterior Model

The result is to pass from an agnostic prior model (a mean function and a covariance kernel) to a data-informed posterior model (an *updated* mean function and covariance kernel)

- Want to predict an unknown function f probabilistically at a new point x*

$$f^* \equiv f(\mathbf{x}^*)$$

- Then the augmented likelyhood function is

$$\begin{aligned}\mathcal{L}^* &\equiv P(\mathbf{f}, f^*) \\ &= (2\pi)^{-(N+1)/2} \det|\mathbf{M}|^{-1/2} \exp\left[-\frac{1}{2}(\mathbf{g}-\bar{\mathbf{g}})^T \mathbf{M}^{-1} (\mathbf{g}-\bar{\mathbf{g}})\right]\end{aligned}$$

where

$$\mathbf{g}^T \equiv [f^*, \mathbf{f}] \ ; \ \bar{\mathbf{g}}^T \equiv [\bar{f}(\mathbf{x}^*), \bar{\mathbf{f}}]$$

$$\mathbf{M} = \begin{pmatrix} k^{**} & \mathbf{k}^{*T} \\ \mathbf{k}^* & \mathbf{K} \end{pmatrix} \qquad \begin{aligned} k^{**} &\equiv K(\mathbf{x}^*, \mathbf{x}^*) \\ [\mathbf{k}^*]_\mu &\equiv K(\mathbf{x}^*, \mathbf{x}^\mu) \end{aligned}$$

Updated Mean Function

The result is to pass from an agnostic prior model (a mean function and a covariance kernel) to a data-informed posterior model (an *updated mean function* and covariance kernel)

- Bayes' Theorem gives

$$P(f^*|\mathbf{f}) = \left(2\pi U^2\right)^{-1/2} \exp\left[-\frac{1}{2}\frac{\left(f^* - \bar{f}^*\right)^2}{U^2}\right]$$

$$\boxed{\bar{f}^* \equiv \bar{f}(\mathbf{x}^*) + \mathbf{k}^{*T} \cdot \mathbf{K}^{-1}\left(\mathbf{f} - \bar{\mathbf{f}}\right),}$$

Our high-order interpolated value at each interface:
a Gaussian probability distribution on the unknown function value f*

$$U^2 \equiv k^{**} - \mathbf{k}^{*T} \cdot \mathbf{K}^{-1} \cdot \mathbf{k}^*$$

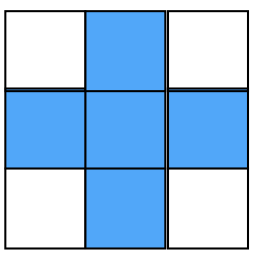
$$K(\mathbf{x}, \mathbf{y}) \equiv \Sigma^2 \exp\left[-\frac{(\mathbf{x} - \mathbf{y})^2}{\sigma^2}\right] \qquad \bar{f}(\mathbf{x}) = f_0 \quad \Sigma^2 = 1 \quad \sigma \gtrapprox R$$

Truly Multidimensional Use of Stencil

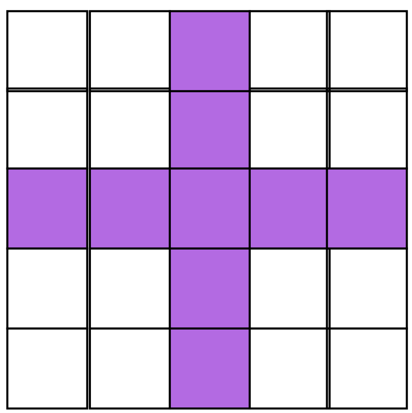
The current GP interpolation method in FLASH for smooth flow tests. For this, we use square exponential (SE) covariance kernel function & interpolation on *"blocky sphere"* of radius R

$$K(\mathbf{x},\mathbf{y}) \equiv \Sigma^2 \exp\left[-\frac{(\mathbf{x}-\mathbf{y})^2}{\sigma^2}\right] \qquad \bar{f}(\mathbf{x}) = f_0 \qquad \Sigma^2 = 1 \qquad \sigma \gtrapprox R$$

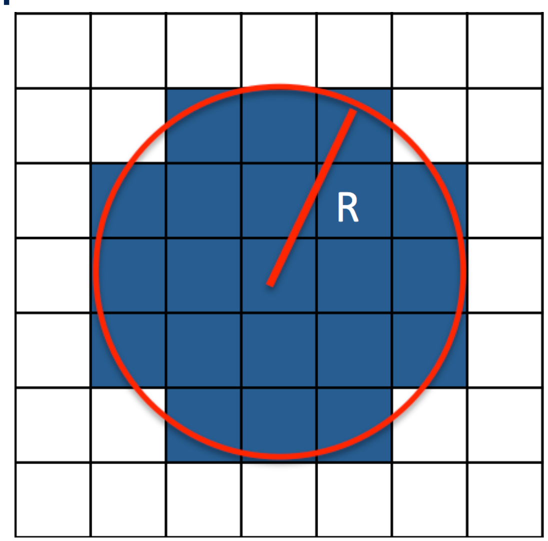
- SE has the property of having a native C^∞ functions, thus can provide with spectral convergence rates when the underlying approximated function is itself C^∞



2D stencil for 2nd order PLM



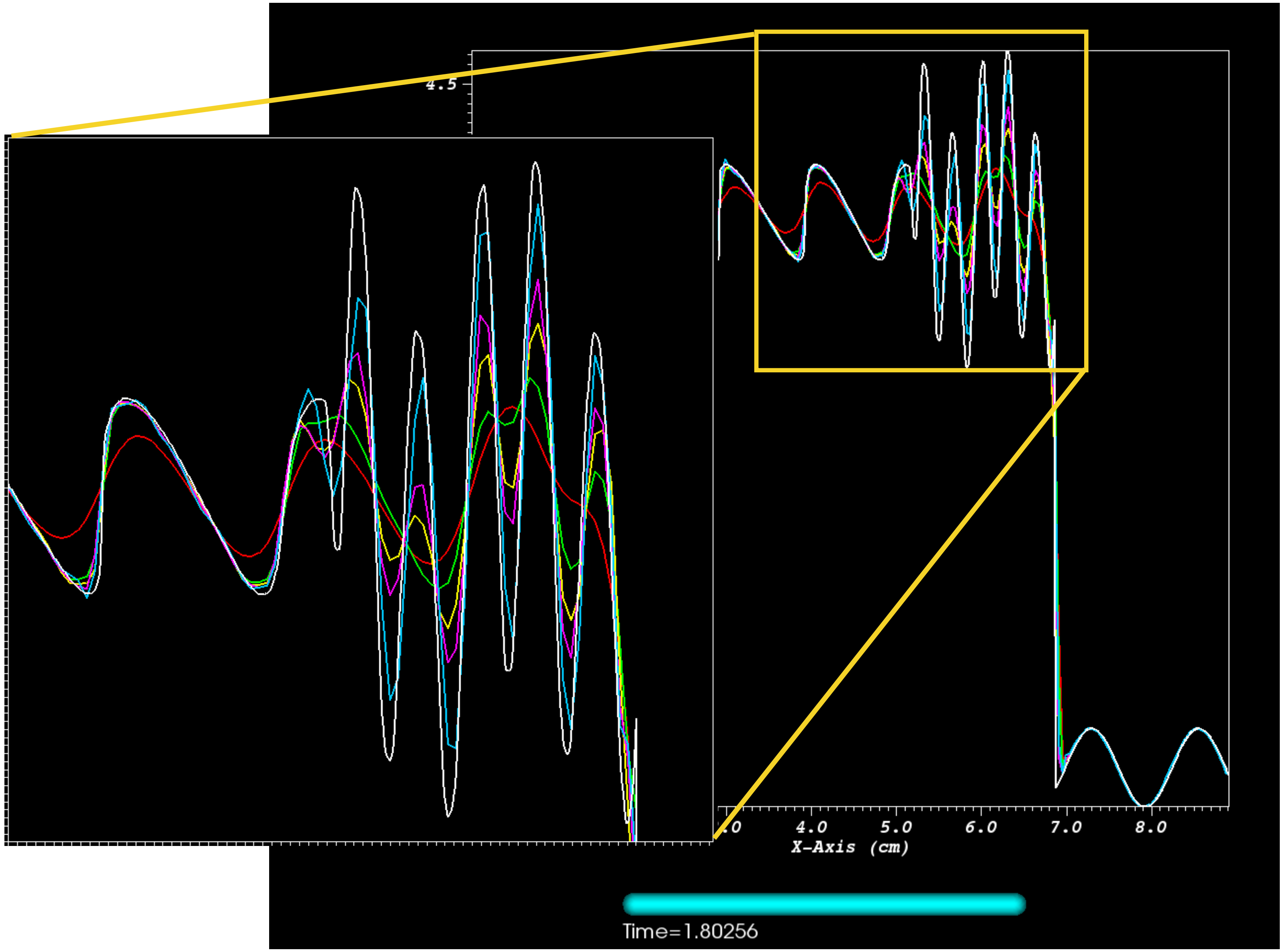
2D stencil for 3rd order PPM



2D stencil for GP

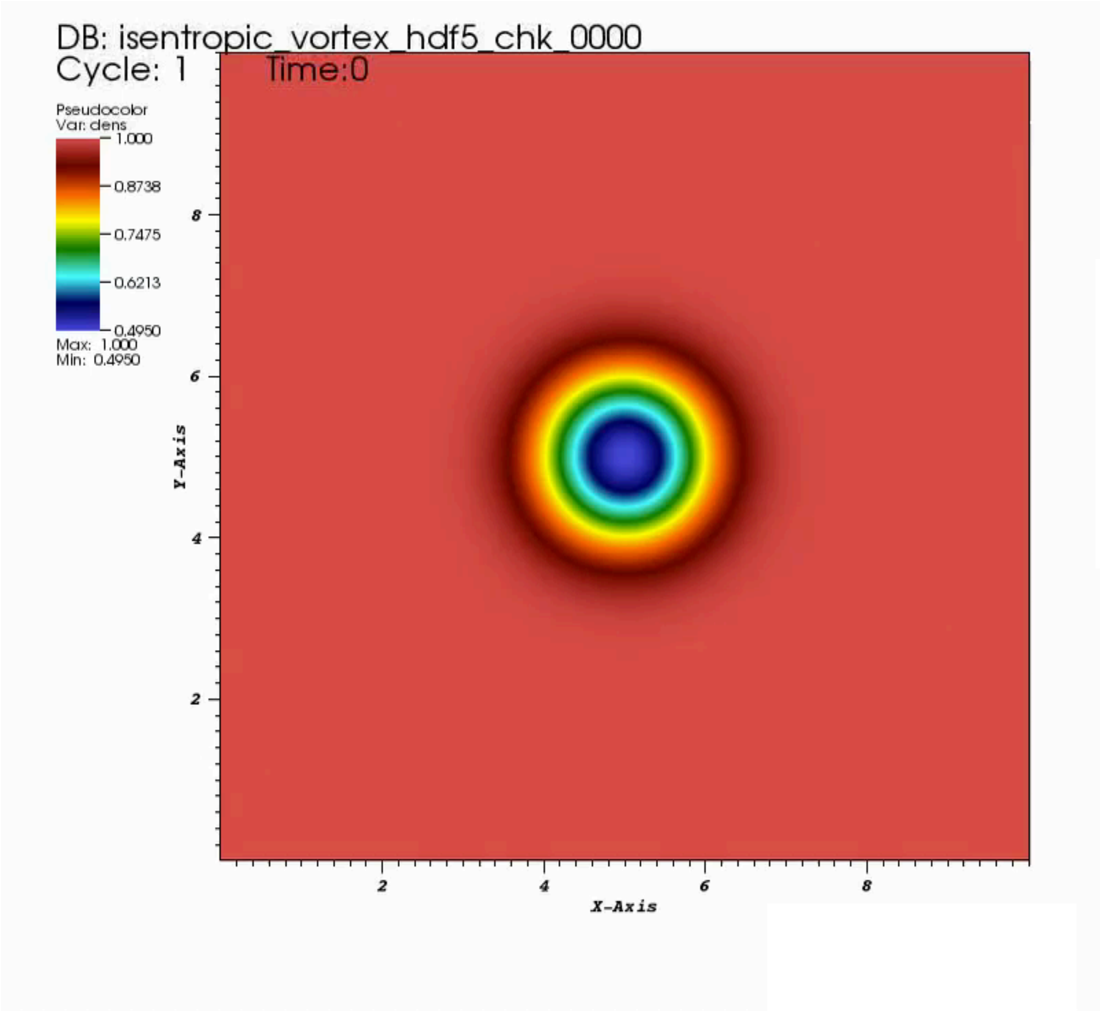
Revisited: 1D Mach 3 Shock

PLM on 1600
GP (spectral)
WENO-Z (5)
PPM (3)
PLM (2)
FOG (1)



Results on Smooth Flows 1

- 2D advection of an isentropic vortex along the domain diagonal on a periodic box $(R = 2\Delta, \sigma = 6\Delta)$



Scheme	ρ	u	v	P
PLM	2.37E-2	8.63E-2	8.62E-2	3.08E-2
PPM	5.44E-4	1.88E-3	1.99E-3	6.93E-4
GP	1.33E-4	4.56E-4	3.75E-4	1.63E-4

Table 1: L1 error norm for the vortex problem.

Results on Smooth Flows II

- 1D advection of Gaussian profile ($R = 2\Delta, \sigma = 12\Delta$)

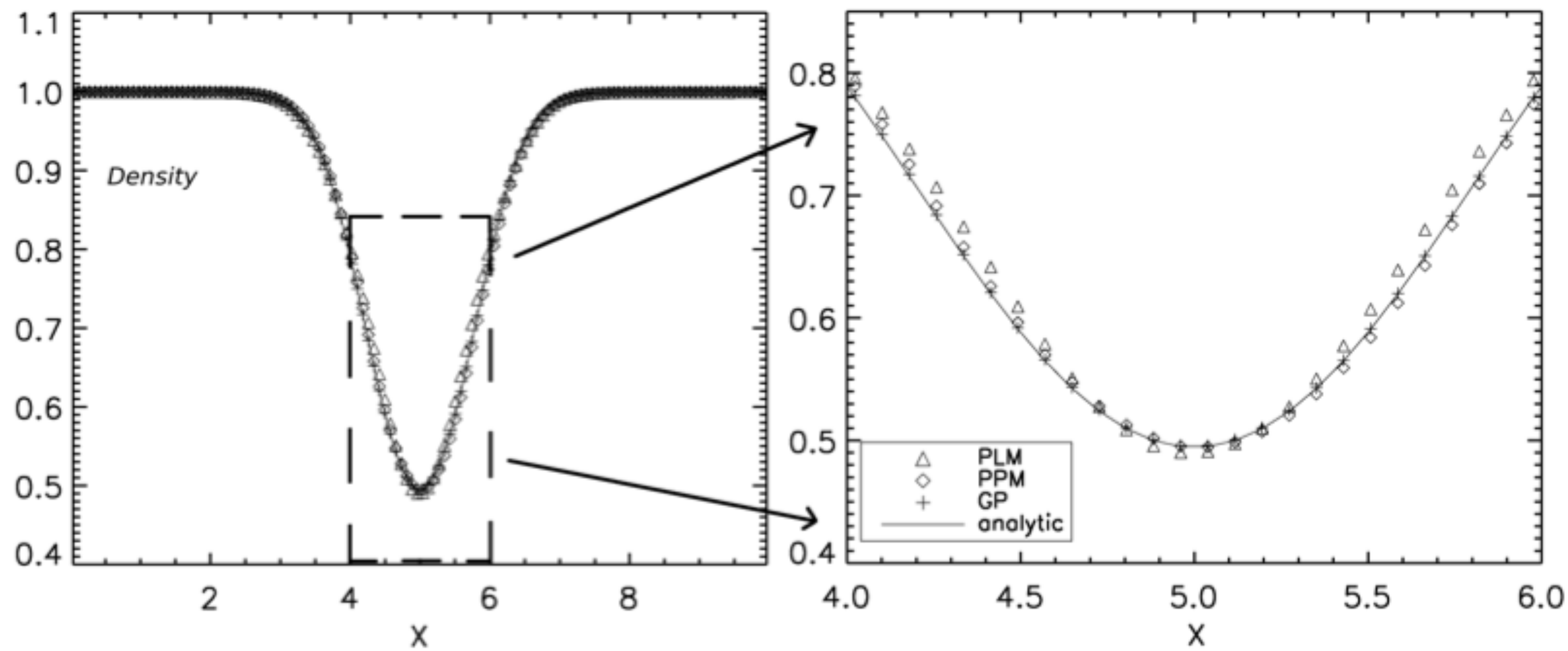
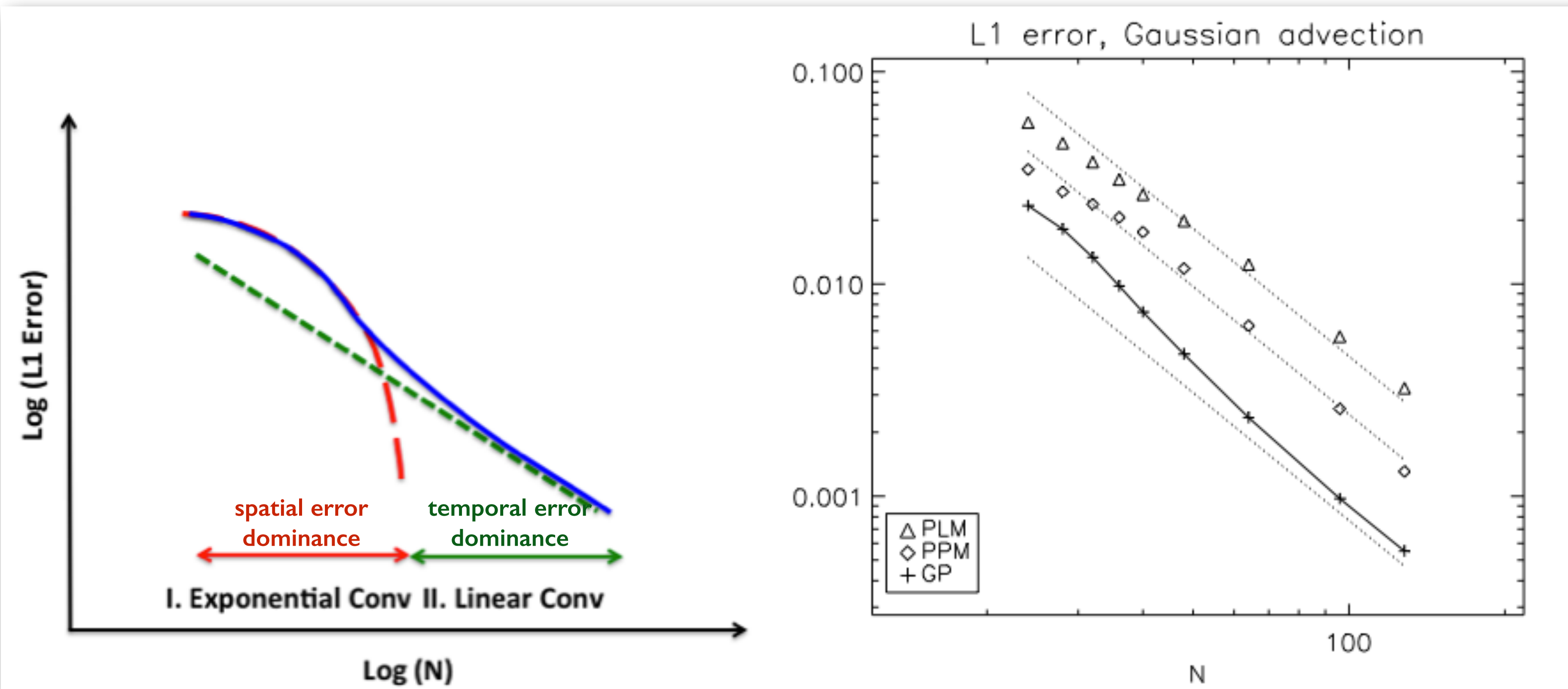


Figure 2: Left panel: One dimensional profile of density after one transition superposed on the initial condition (solid line). The symbols represent different interpolation schemes, namely cross for GP, diamond for PPM and triangle for PLM. Right panel: Close-up in the central region. The solution recovered with GP matches perfectly the reference solution, while the errors are smaller with respect to the other schemes.

Exponential Convergence Rate

- 1D advection of Gaussian profile ($R = 2\Delta, \sigma = 12\Delta$)



Summary

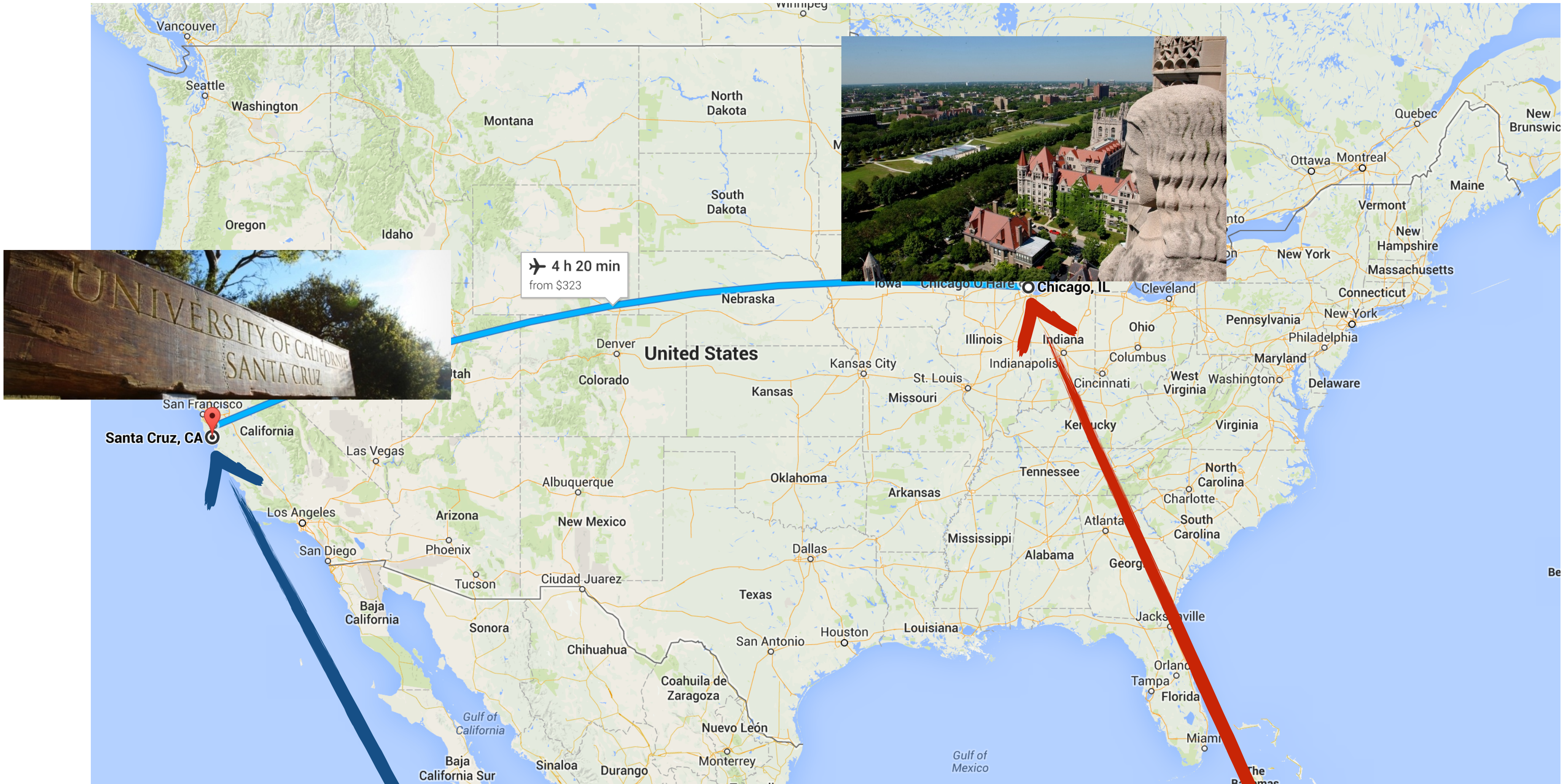
- High-order method is a good approach to embody the desired tradeoff between memory and computation in future HPC

- *A new high-order method based on GP* that utilizes "nonparametric" fitting of spatial data exhibits an evidence of spectral convergence

- GP function interpolation is Bayesian updating of a prior distribution by training the data which is to be interpolated

- The GP interpolation at a new point - in our case, at each interface - is a "prediction" made by the updated probability distribution

Tak! Spørgsmål?

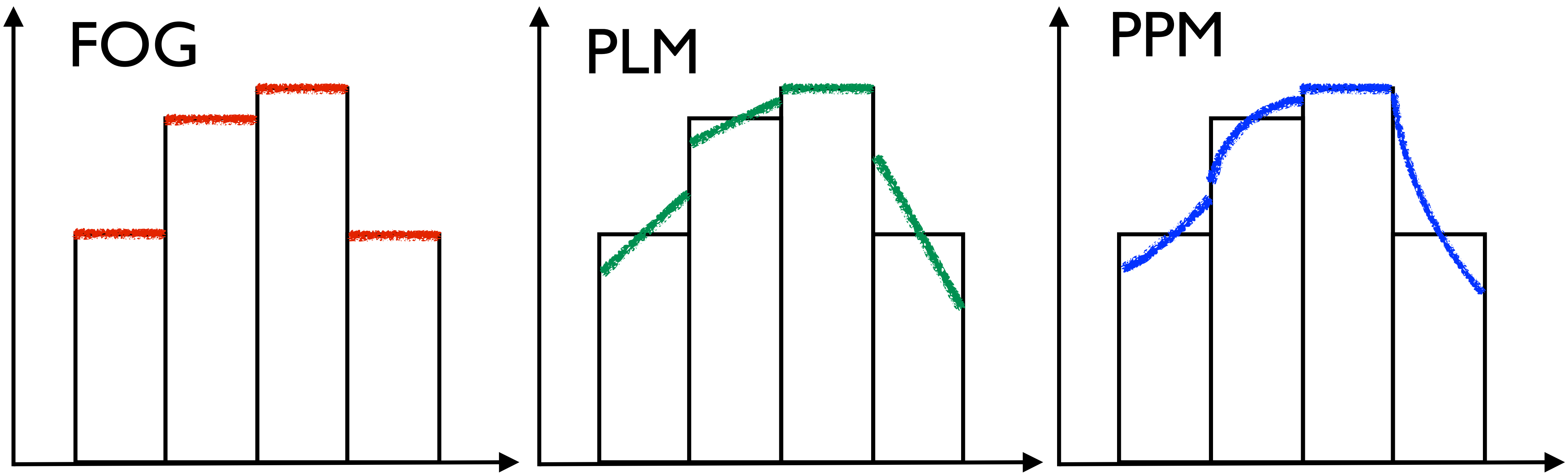


Me:
dlee79@ucsc.edu
ams.soe.ucsc.edu/people/dongwook

FLASH:
www.flash.uchicago.edu

Supplementary Slides

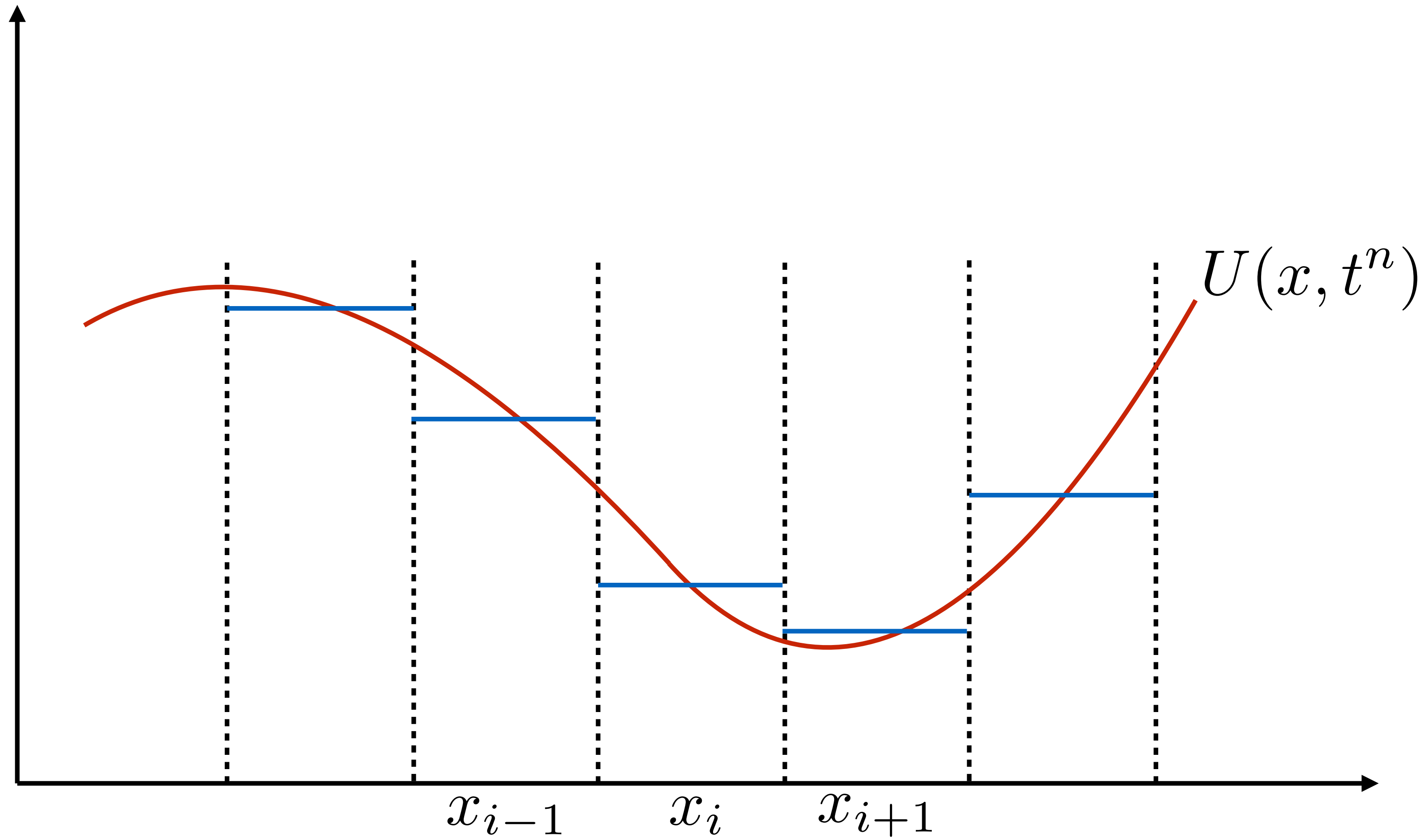
High-Order Polynomial Reconstruction



- **Godunov's order-barrier theorem (1959)**
  - Monotonicity-preserving advection schemes are at most first-order! (Oh no…)
  - Only true for linear PDE theory (YES!)

- **High-order "*polynomial*" schemes became available using non-linear slope limiters**
  (70's and 80's: Boris, van Leer, Zalesak, Colella, Harten, Shu, Engquist, etc)
  - Can't avoid oscillations completely (non-TVD)
  - Instability grows (numerical INSTABILITY!)

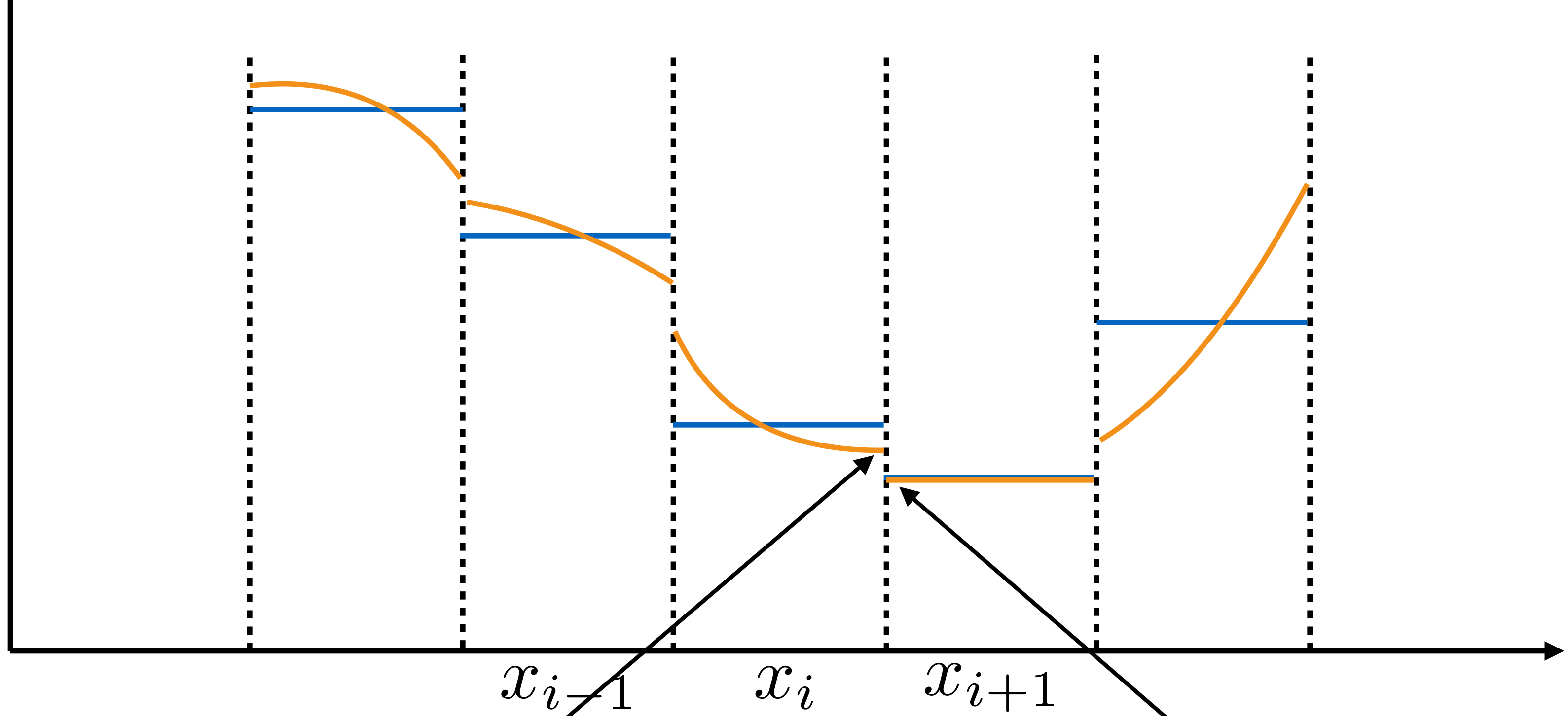
A Discrete World of FV



A Discrete World of FV

piecewise polynomial reconstruction
on each cell

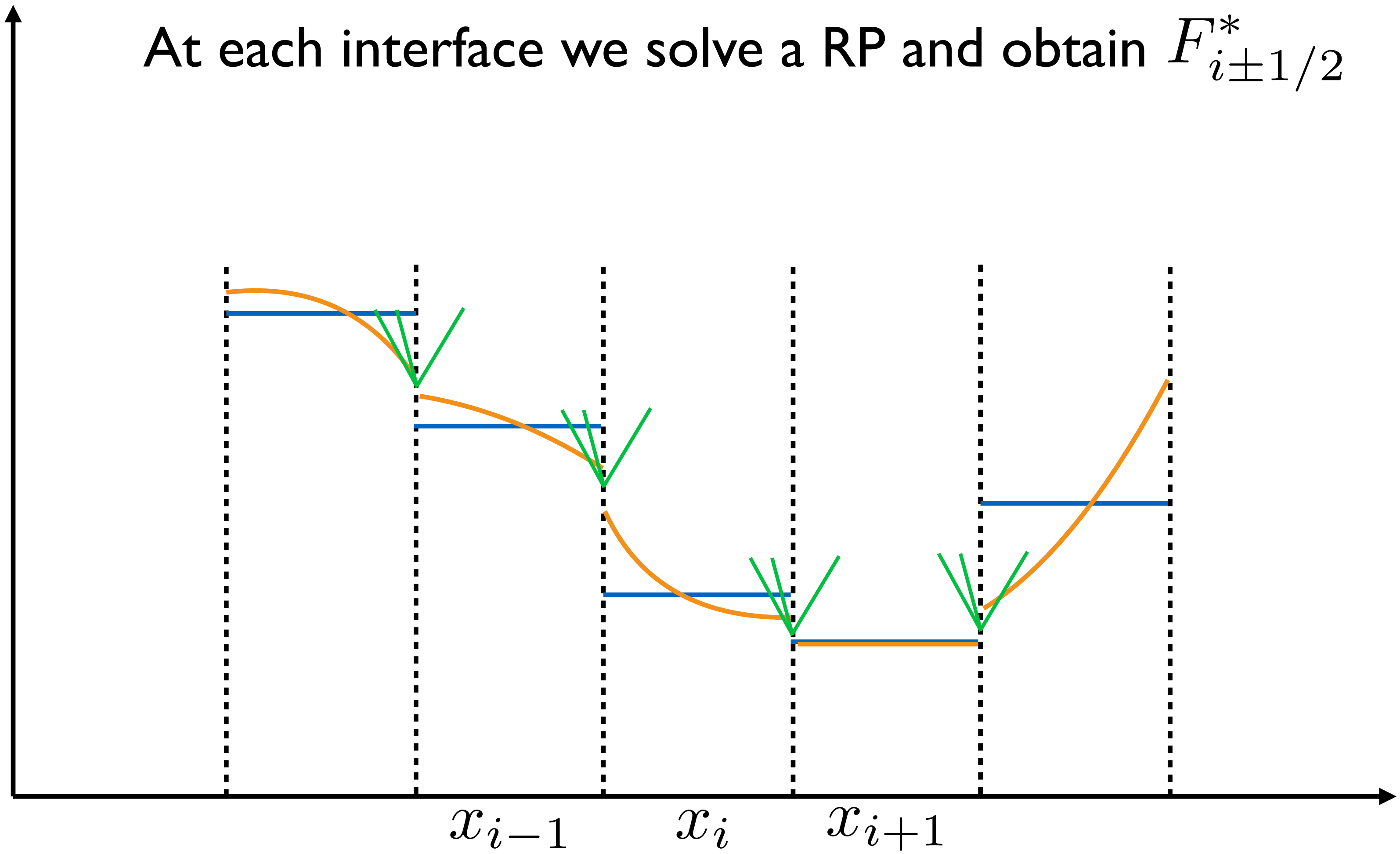
$$u(x_i, t^n) = P_i(x), x \in (x_{i-1/2}, x_{i-1/2})$$



$$u_R = P_i(x_{i+1/2}) \qquad u_L = P_{i+1}(x_{i+1/2})$$

A Discrete World of FV

At each interface we solve a RP and obtain $F^*_{i\pm 1/2}$



A Discrete World of FV

We are ready to advance our solution in time and get new volume-averaged states

$$U_i^{n+1} = U_i^n - \frac{\Delta t}{\Delta x}\left(F^*_{i+1/2} - F^*_{i-1/2}\right)$$