Advanced Multi-Scale Methods for Hypersonic Propulsion

By

J. M. Powers, D. Wirasaet and S. Paolucci

Aerospace and Mechanical Engineering
University of Notre Dame, Indiana 46556

NASA Hypersonics NRA Review, San Antonio, Texas, 23-25 January 2008

Motivation

- Hypersonic propulsion problems with detailed kinetics admit a startling breadth of length and time scales.
- e.g. for reaction alone, length scales can span five orders of magnitude.
- Smallest scale for DNS often at or below micron scales.
- Some adaptive method needed for DNS.
- We use an adaptive wavelet method building on our earlier NSF- and AFOSR-supported work (late 1990s).

Project Status

- Post-Doc Dr. Damrongsek Wirasaet began September 2007.
- Two Ph.D. students, Mr. Christopher Romick, and Mr. Zack Zilkowski will be supported starting Summer 2008.
- Three-dimensional unsteady adaptive wavelet framework built and verified on model problems.
- Compressible Navier-Stokes model built and tested on one-dimensional viscous Sod shock tube problem.
- Three-dimensional reactive flow solver should be ready in May 2008.
- Code to be exercised carefully on wedges, cones, shear layers, shock tubes, etc. (i.e. problems with a good NASA experimental data base), 2008-2009.
- Parallel version 2009.

"Simple" Example: Ozone Detonation

Model Equations: Reactive Euler (Aslam (DOE/LANL) and Powers, AIAA Reno 2008)

$$
\begin{aligned}
\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x}(\rho u) &= 0, \\
\frac{\partial}{\partial t}(\rho u) + \frac{\partial}{\partial x}\left(\rho u^2 + p\right) &= 0, \\
\frac{\partial}{\partial t}(\rho E) + \frac{\partial}{\partial x}(u(\rho E + p)) &= 0, \\
\frac{\partial}{\partial t}(\rho Y_i) + \frac{\partial}{\partial x}(\rho u Y_i) &= M_i \dot{\omega}_i, \\
p &= \rho \Re T \sum_{i=1}^{N} \frac{Y_i}{M_i}, \\
E &= e(T, Y_i) + \frac{1}{2}u^2, \\
\dot{\omega}_i &= \dot{\omega}_i(T, Y_i).
\end{aligned}
$$

Ozone Reaction Kinetics

Reaction	a_j^f, a_j^r	β_j^f, β_j^r	E_j^f, E_j^r
$O_3 + M \leftrightarrows O_2 + O + M$	6.76×10^6	2.50	1.01×10^{12}
	1.18×10^2	3.50	0.00
$O + O_3 \leftrightarrows 2O_2$	4.58×10^6	2.50	2.51×10^{11}
	1.18×10^6	2.50	4.15×10^{12}
$O_2 + M \leftrightarrows 2O + M$	5.71×10^6	2.50	4.91×10^{12}
	2.47×10^2	3.50	0.00

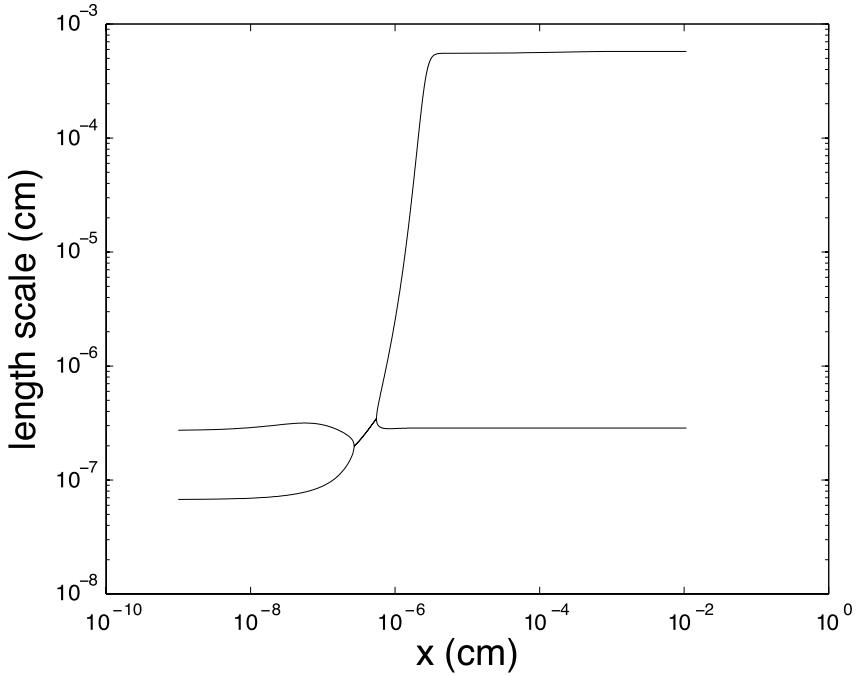
Hirschfelder, *et al.*, *J. Chem. Phys.*, 1953.

Computational Method for "Simple" Problem

- Steady wave structure (Powers and Paolucci, 2005, *AIAA J.*):
  - LSODE solver,
  - Ten second run time on single processor machine,
- Unsteady wave structure (Henrick, Aslam, Powers, 2006, *J. Comp. Phys.*):
  - Uniform grid here.
  - Shock fitting coupled with a high order method for continuous regions,
  - Two hour run time on desktop Macintosh.

Stable Strong Overdriven Case

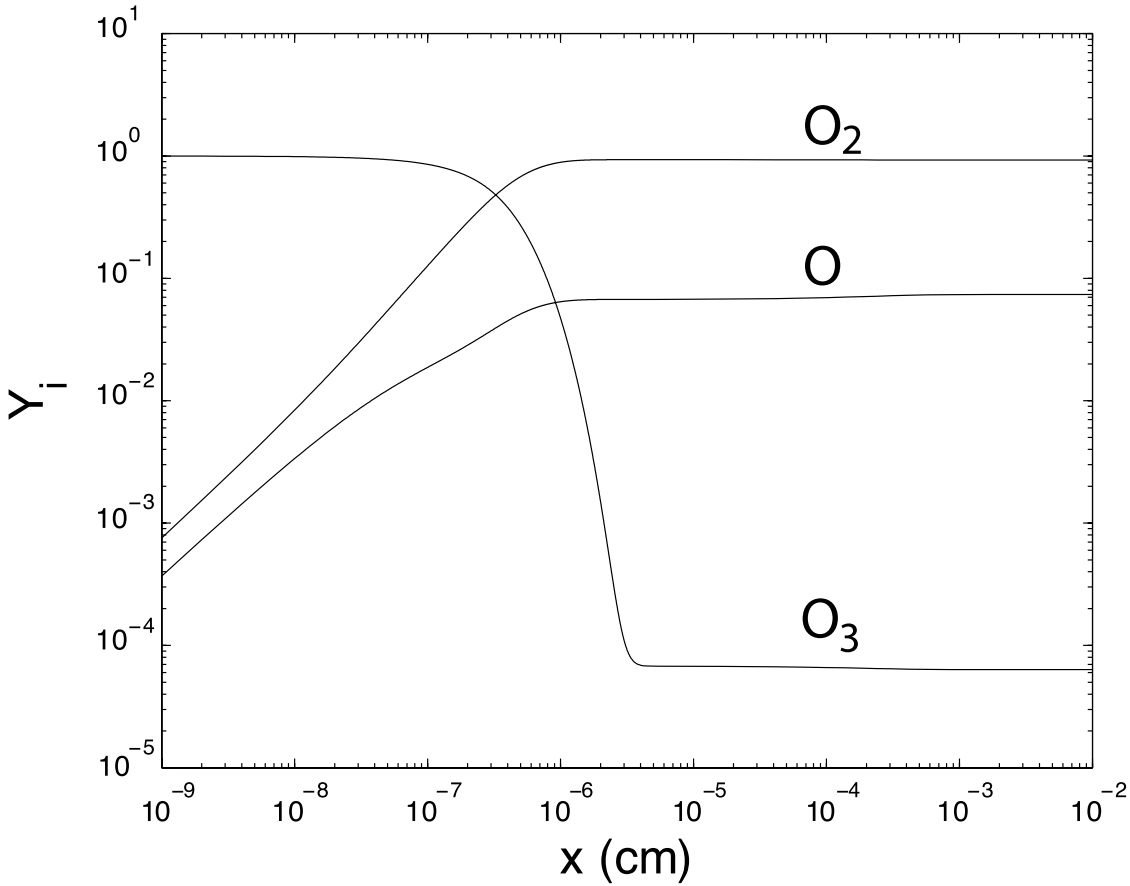
- Length scales from spatial eigenvalue analysis: $D = 2.5 \times 10^5\ cm/s$. Smallest scale $\approx 10^{-7} cm$.



- Mean-Free-Path Estimate: a simple estimate for this scale is given by *Vincenti and Kruger, '65*: $\ell_{mfp} = \frac{M}{\sqrt{2}\mathcal{N}\pi d^2 \rho} \sim 10^{-7}\ cm.$

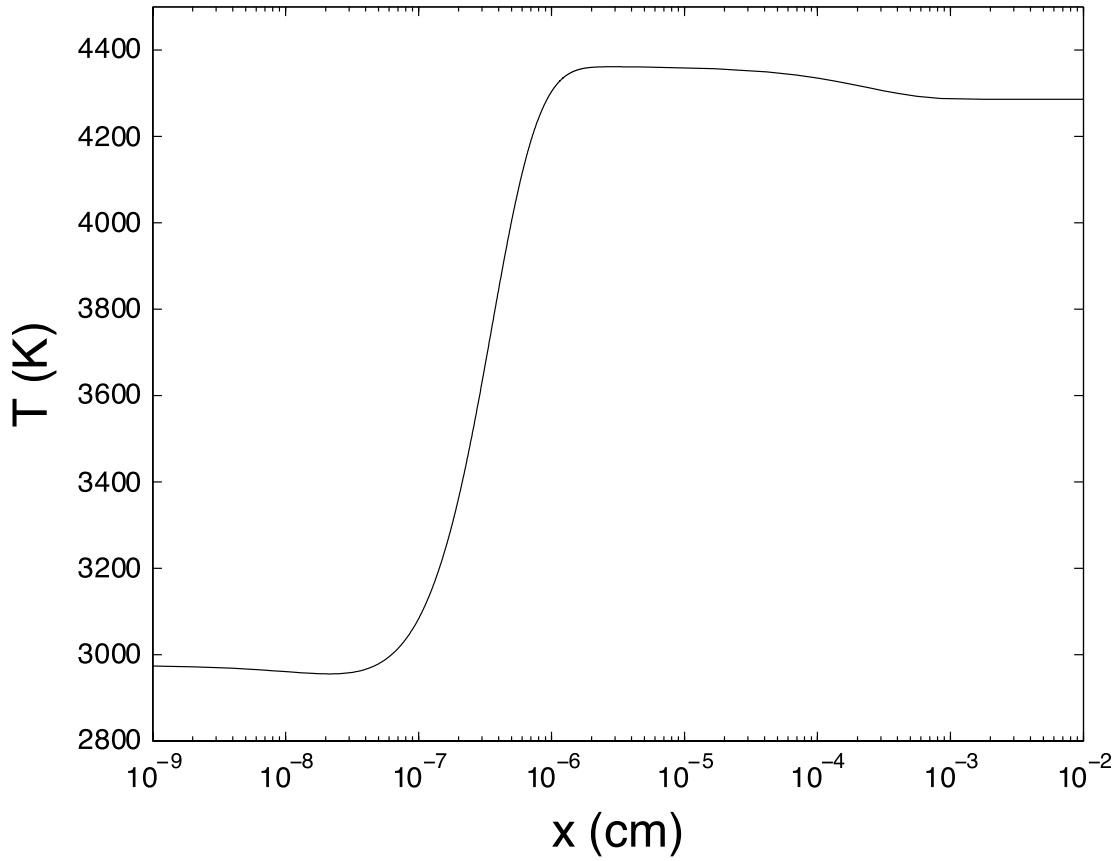
Stable Strongly Overdriven Case: Mass Fractions

$D = 2.5 \times 10^5\ cm/s.$



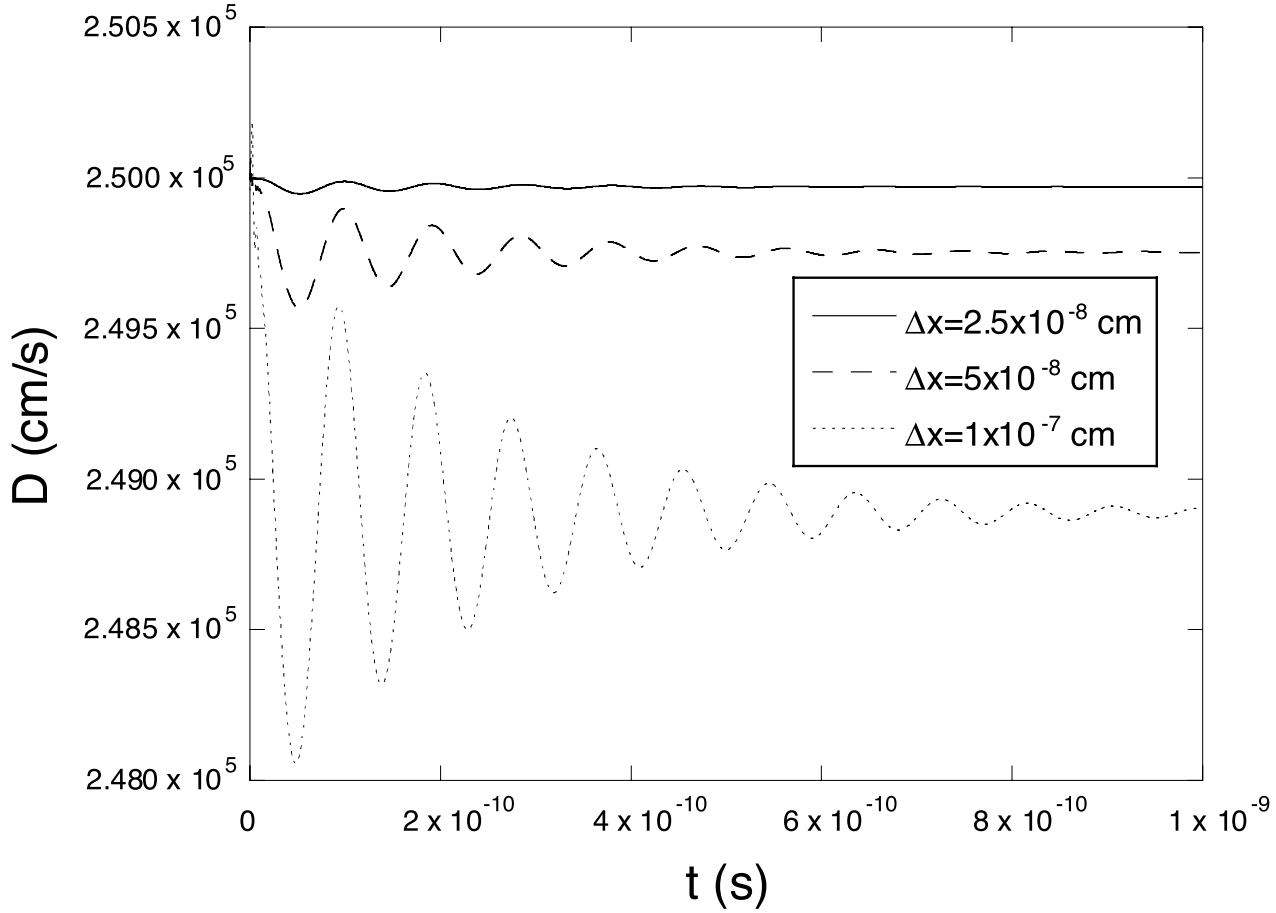
Stable Strongly Overdriven Case: Temperature

$D = 2.5 \times 10^5\ cm/s.$



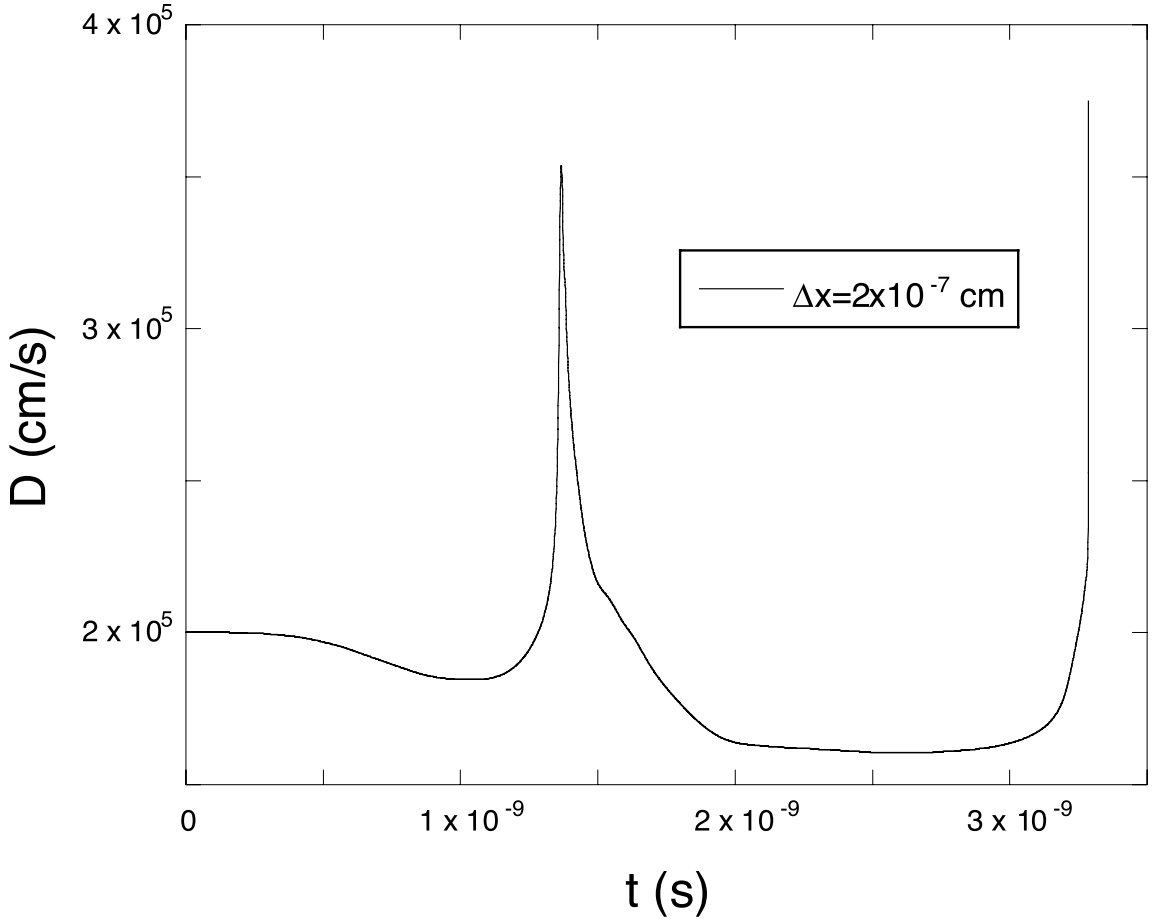
Stable Strongly Overdriven Case: Transient Behavior for various resolutions

Initialize with steady structure of $D = 2.5 \times 10^5\ cm/s$.



Unstable Moderately Overdriven Case: Transient Behavior

Initialize with steady structure of $D = 2 \times 10^5\ cm/s$.



Effect of Resolution on Unstable Moderately Overdriven Case

Δx	Numerical Result
$1 \times 10^{-7}\ cm$	Unstable Pulsation
$2 \times 10^{-7}\ cm$	Unstable Pulsation
$4 \times 10^{-7}\ cm$	Unstable Pulsation
$8 \times 10^{-7}\ cm$	O_2 mass fraction > 1
$1.6 \times 10^{-6}\ cm$	O_2 mass fraction > 1

- Algorithm failure for insufficient resolution.
- At low resolution, one misses critical dynamics.

Implications for Operator Splitting for Implicit Time Integration of Chemistry

- This popular method, while numerically stable, misses fine scale dynamics entirely.
- This method would capture the dynamics if $\Delta x = 10^{-7} cm$, in which case there would be no need for implicit time integration.

Remarks

- ☞ Unsteady detonation dynamics can be accurately simulated when sub-micron scale structures admitted by detailed kinetics are captured with uniform ultra-fine grids.
- ☞ Shock fitting coupled with high order spatial discretization assures numerical corruption is minimal.
- ☞ At these length scales, diffusion will play a role and should be included.
- ☞ Since an ultra-fine grid is required to capture fine scale structures, accurate simulations with uniform grids may not be practical (since DOFs of the uniform grid increases exponentially w.r.t. the spatial dimension d, *i.e.* DOFs $\sim N^d$)

Alternative: An Adaptive Wavelet Method

- An adaptive method is designed based on interpolating multiscale wavelet bases.
- An adaptive grid is constructed from an ε-truncated wavelet approximation of the solution and connected to a dyadic grid.
- The grid refinement strategy is simple and has no requirement on mesh connectivity. It amounts to retaining and adding grid points associated with important wavelet bases and removing grid points associated with negligible wavelet bases.
- The method utilizes consistent finite differences on the adaptive grid for derivative approximation to avoid costly derivative approximation with direct differentiation of multiscale bases.
- See detail in Y. Rastigejev and S. Paolucci, *Int. J. Num. Fluids*, 52, 2005, and D. Wirasaet, PhD Thesis, U. of Notre Dame, 2007.

Algorithm for Adaptively Solving PDEs

① Perform time-discretization of the PDEs:

$$\frac{\partial u}{\partial t} = F(t, u, u_x, u_{xx}, \ldots)$$

$$\Downarrow \text{ time discretization}$$

$$\mathcal{A}^m u^m = \mathcal{F}^m(t^{m-q}, u^{m-q}, u_x^{m-q}, u_{xx}^{m-q}, \ldots), q = 0, \ldots, r \quad (1)$$

② Determine an adaptive grid $\boldsymbol{\mathcal{V}}^0$ by thresholding the wavelet representation of initial condition u^0.

③ `For m = 1, nstep`

❶ Solve (1) on the grid $\boldsymbol{\mathcal{V}}^{m-1}$ to obtain approximate solution u^m.

❷ Determine the new adaptive grid, $\boldsymbol{\mathcal{V}}^m$ by thresholding the wavelet representation of the obtained solution u^m.

❸ Perform wavelet interpolation to obtain the function value of u^{m-q}, $q = 0, \ldots, r$ on the new adaptive grid $\boldsymbol{\mathcal{V}}^m$.

Some Remarks

- In each integration step m, the operations required in the algorithm are of $O(N)$, $N = \dim(\boldsymbol{\mathcal{V}}^m)$.
- Assuming temporal resolution, an accurate approximate solution is expected to behave in overall like

$$\|u_{num} - u_{exact}\|_{L_{\boldsymbol{\mathcal{V}}},\infty} = O(N^{\min(p-2,n)/d}) = O(\varepsilon^{\min(p-2,n)/p})$$

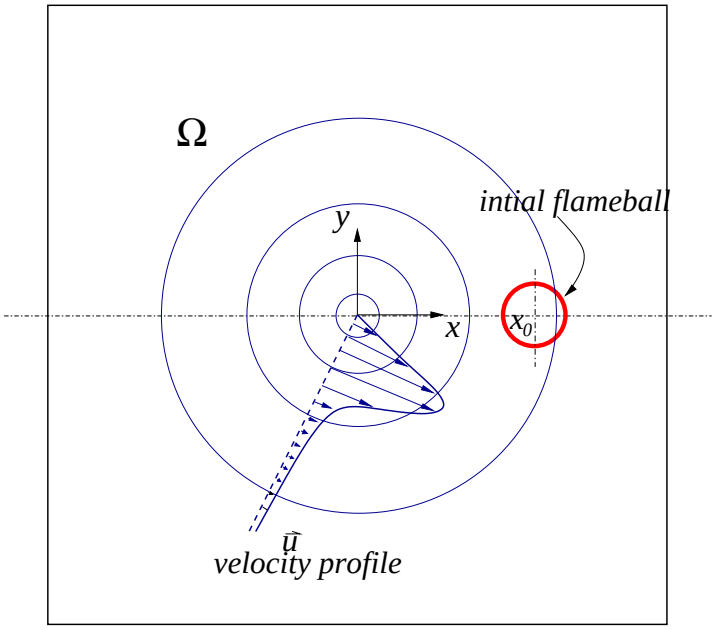
  where

  ε - the value of threshold parameter
  p - the order of interpolating wavelet
  n - the order of FD used in derivative approximation
  d - the spatial dimension

- The behavior of errors in the solution from numerical experiments conforms reasonably well with the estimate given above. Often, the achieved order is better than the estimate!

Flameball-Vortex interaction

Governing equations [O. Roussel and K. Schneider, 2005]:



$$\frac{\partial T}{\partial t} + (\mathbf{u}\cdot\nabla)T = \nabla^2 T + \omega - s,$$

$$\frac{\partial Y}{\partial t} + (\mathbf{u}\cdot\nabla)Y = \frac{1}{Le}\nabla^2 Y - \omega,$$

$$\omega(T,Y) = \frac{Ze^2}{2Le} Y \exp\left[-\frac{Ze(1-T)}{1-\alpha(1-T)}\right],$$

$$s(T) = \gamma\left[(T+\alpha^{-1}-1) - (\alpha^{-1}-1)\right]^4.$$

Note: where T denotes temperature, Y partial mass of the fresh premixed gas, ω the reaction rate, s heat loss due to radiation, α burnt-unburnt temperature ratio, γ radiation coefficients, Le the Lewis number, Ze the Zeldovich number.

Numerical Simulations

Velocity:

$$\mathbf{u} = \frac{\Gamma}{2\pi r}\left(1 - \exp\left[-\frac{r^2}{4Pr(t+\tau)}\right]\right), \quad r = \sqrt{x^2+y^2}$$

Initial conditions:

$$T(\hat{r},0) = \begin{cases} 1 & \text{if } \hat{r} \le \hat{r}_0, \\ \exp(1-\hat{r}/\hat{r}_0) & \text{if } \hat{r} > \hat{r}_0, \end{cases}$$

$$Y(\hat{r},0) = \begin{cases} 0 & \text{if } \hat{r} \le \hat{r}_0, \\ 1-\exp(Le(1-\hat{r}/\hat{r}_0)) & \text{if } \hat{r} > \hat{r}_0, \end{cases}$$

$$\alpha = 0.64$$
$$\gamma = 0$$
$$Le = 0.3$$
$$Ze = 10$$
$$Pr = 0.01$$
$$\tau = 0.01$$
$$\Gamma = 0 \text{ to } 1350$$

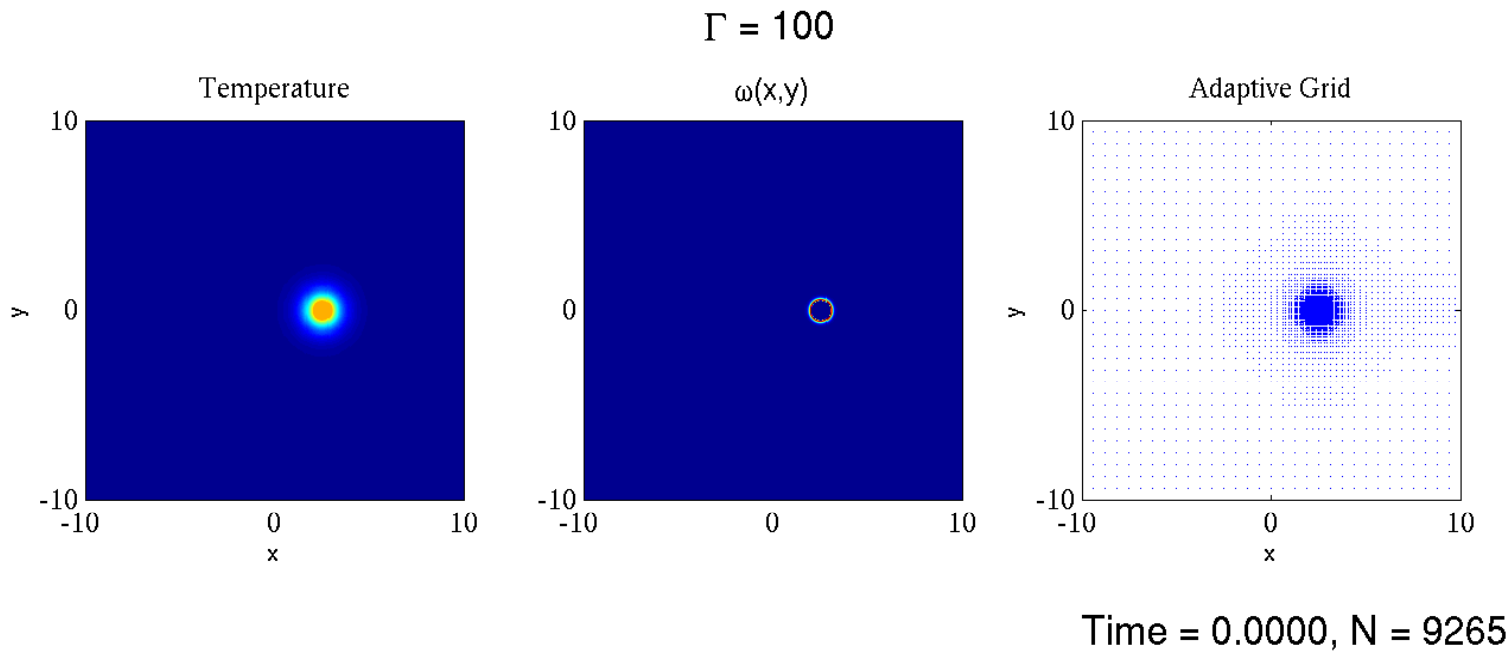
Boundary conditions:

$$\left.\frac{\partial T}{\partial n}\right|_{\partial\Omega} = \left.\frac{\partial Y}{\partial n}\right|_{\partial\Omega} = 0.$$

Numerical Simulation

Computational domain	:	$\Omega = [-10, 10]^2$
Time-Discretization	:	2^{nd} trapezoidal scheme for linear term 2^{nd} Adam-Bashforth for nonlinear term $t_{\text{final}} = 1.5$
Interpolating wavelet	:	$p = 6$
Derivative approximation	:	$n = 4$ (9 point-stencil)
Coarsest level	:	$j_0 = 4$
Max. levels of refinement	:	$J - j_0 = 6$
Threshold parameters	:	$\boldsymbol{\varepsilon} = (5 \times 10^{-3}, 5 \times 10^{-3})$, $(1 \times 10^{-3}, 1 \times 10^{-3})$, and $(5 \times 10^{-4}, 5 \times 10^{-4})$

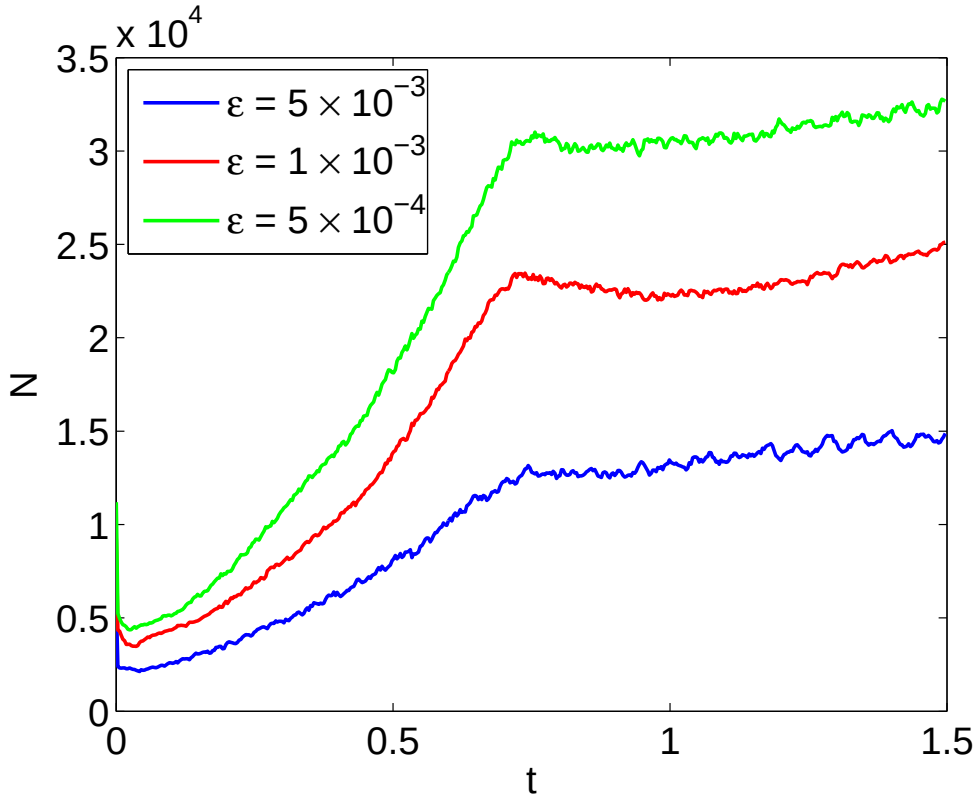
Result of simulation for $\Gamma = 100$



Threshold	: $\boldsymbol{\varepsilon} = \{10^{-3}, 10^{-3}\}$
Adaptive scheme	: $p = 6$ and $n = 4$
Resolution	: $j_0 = 4$ and $J - j_0 = 6$

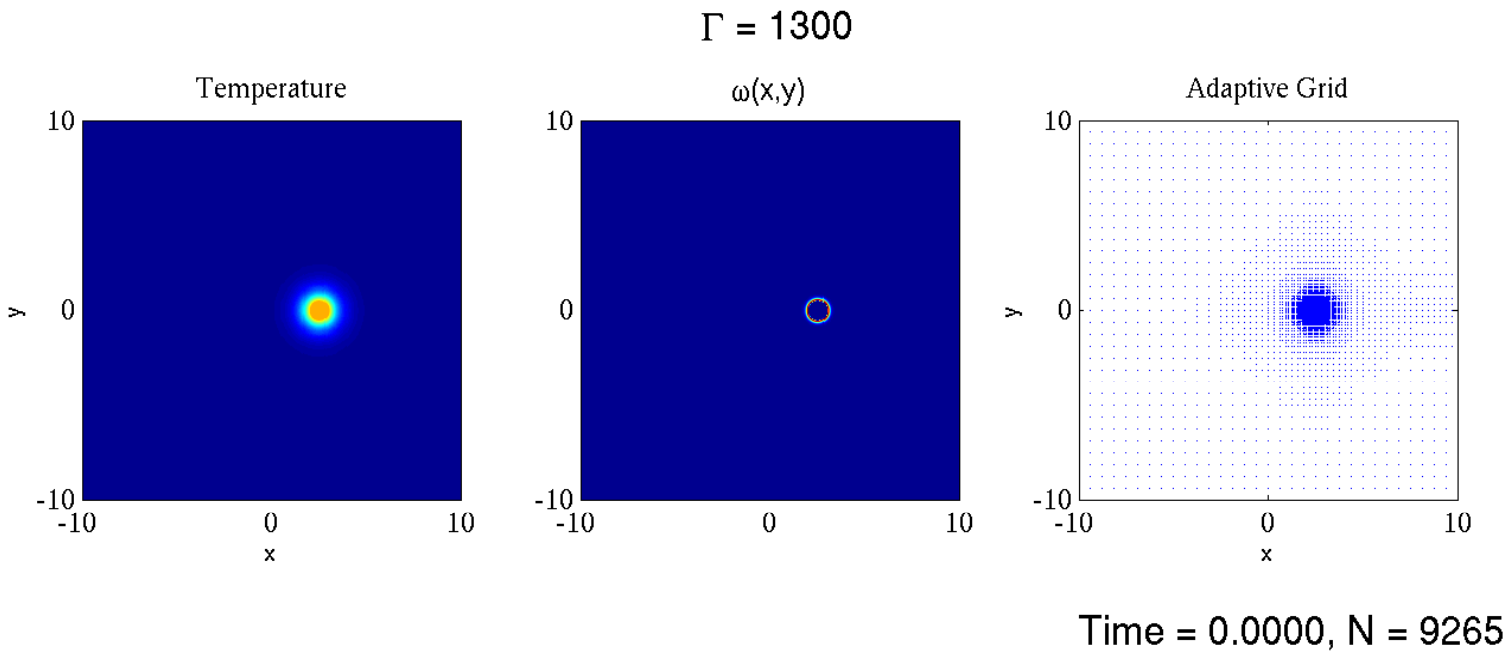
Results for $\Gamma = 100$

Evolution of DOFs required



☞ As a threshold value ε is decreased, the number of DOFs, N, generated by the algorithm increases automatically.

Result of simulation for $\Gamma = 1300$



Threshold	: $\varepsilon = \{10^{-3}, 10^{-3}\}$
Adaptive scheme	: $p = 6$ and $n = 4$
Resolution	: $j_0 = 4$ and $J - j_0 = 6$

Compressible Flows

A code is being developed for numerical solution of 1, 2, and 3D reactive compressible Navier-Stokes equations written as

$$
\begin{aligned}
\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) &= 0 \\
\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u}^T \mathbf{u}) &= -\nabla \cdot (p\mathbf{I} + \boldsymbol{\tau}), \\
\frac{\partial \rho E}{\partial t} + \nabla \cdot (\rho E \mathbf{u}) &= -\nabla \cdot (\mathbf{u} \cdot (p\mathbf{I} + \boldsymbol{\tau})) - \nabla \cdot (\mathbf{q}_c + \mathbf{j}_r), \\
\frac{\partial \rho Y_i}{\partial t} + \nabla \cdot (\rho Y_i \mathbf{u}) &= -\nabla \cdot (\rho Y_i \mathbf{U}_{Di}) + M_i \dot{\omega}_i, \quad i = 1, \ldots, N
\end{aligned}
$$

• ρ-density, $\mathbf{u}$-velocity vector, E-specific total energy, Y_i-mass fraction of species i, $\mathbf{I}$-unit tensor, $\boldsymbol{\tau}$-viscous stress tensor, $\mathbf{q}_c$ and $\mathbf{j}_r$-heat flux, $\mathbf{U}_{Di}$-diffusion velocity of species i, and $\dot{\omega}_i$-reaction rate of species i.

Compressible flows (cond't)

☞ Where

$$\boldsymbol{\tau} = 2\mu\left(\frac{\nabla\mathbf{u} + (\nabla\mathbf{u})^T}{2} - \frac{1}{3}(\nabla\cdot\mathbf{u})\mathbf{I}\right),$$

$$\boldsymbol{q}_c = -\kappa\nabla T,$$

$$E = e + \frac{1}{2}|\mathbf{u}|^2,$$

$$P = \frac{\rho RT}{M}.$$

☞ Additional equations for a chemical system of N species and M reactions,

$$\sum_{k=1}^{N}\nu'_{kj}\mathcal{M}_k \rightleftharpoons \sum_{k=1}^{N}\nu''_{kj}\mathcal{M}_k, \quad j = 1,\ldots,M$$

Compressible flow (cond't)

$$\sum_i Y_i = 1,$$

$$\mathbf{j}_r = \rho \sum_{k=1}^{N} h_k Y_k \mathbf{U}_{Dk}$$

$$e = \sum_{i=1}^{N} Y_i h_i - \frac{RT_0}{M}, \quad h_i = h_{i0} + \int_{T_0}^{T} c_{pi} dT,$$

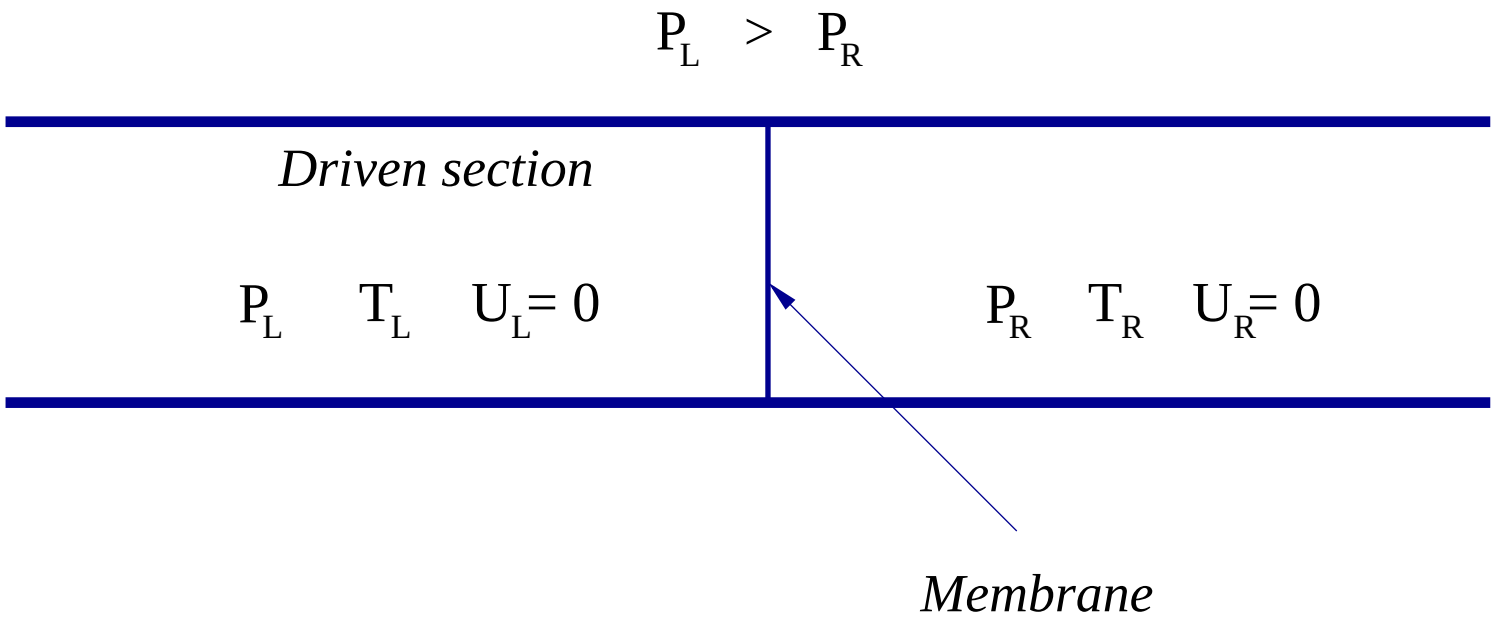
$$\dot{\omega}_i = M_k \sum_{j=1}^{M} (\nu''_{kj} - \nu'_{kj}) \left[K_{fj} \prod_{k=1}^{N} \left(\frac{\rho Y_k}{M_k} \right)^{\nu'_{kj}} - K_{rj} \prod_{k=1}^{N} \left(\frac{\rho Y_k}{M_k} \right)^{\nu''_{kj}} \right],$$

$$\sum_{k=1}^{N} (\nu''_{kj} - \nu'_{kj}) M_k = 0, \; j = 1, \ldots, M,$$

$$\nabla \left(\frac{Y_j M}{M_k} \right) = \sum_{k=1}^{N} \frac{Y_j Y_k M^2}{M_j M_k D_{jk}} (\mathbf{U}_{Dk} - \mathbf{U}_{Dj}) + \left(Y_j - \frac{Y_j M}{M_j} \right) \frac{\nabla P}{P}, \; j = 1, \ldots, N.$$

1-D viscous Shock Tube

Schematic diagram of Sod's problem:



Numerical Simulation

Test case:

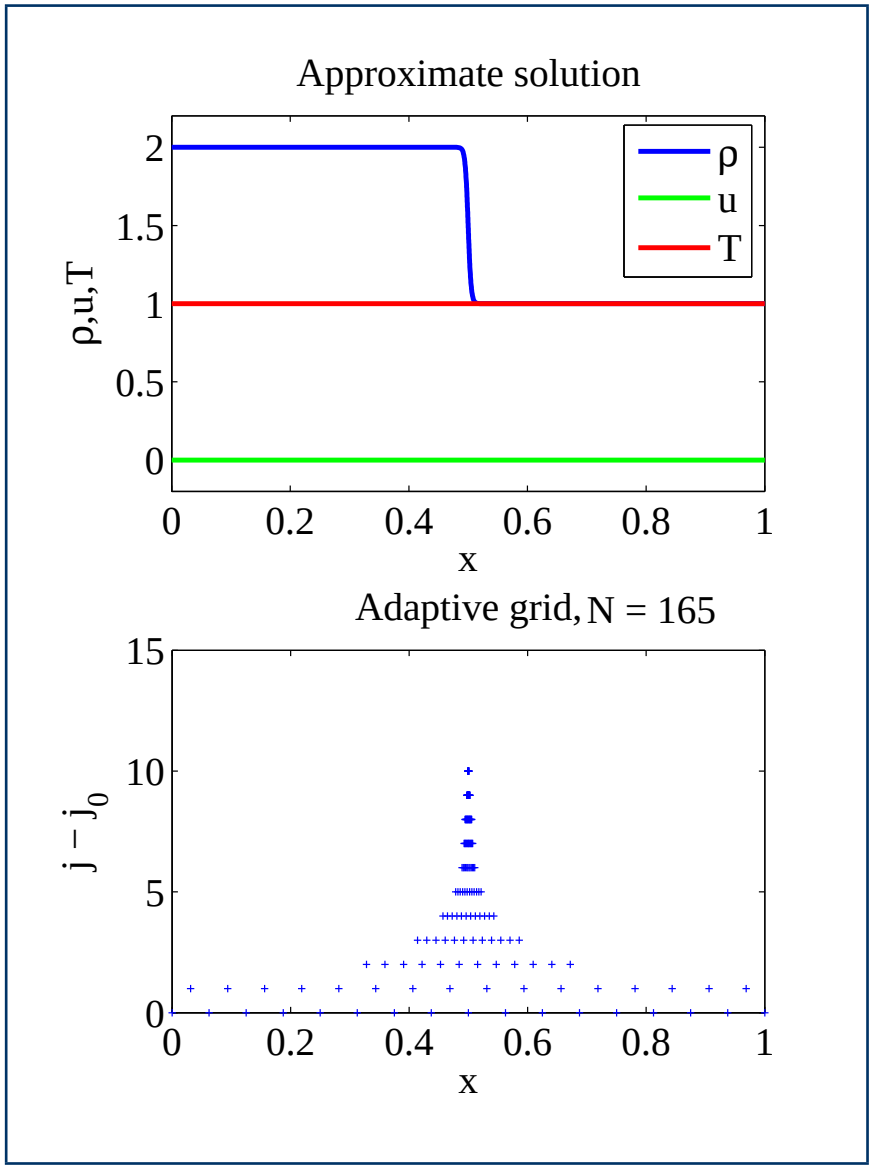
Fluid	:	Air
		$P_R = 2 \times 10^5 \ \ [N/m^2]$, $T_R = 300 \ \ [K]$
At $t = 0$	:	$P_L = 1 \times 10^5 \ \ [N/m^2]$, $T_L = 300 \ \ [K]$
		$U_L = 0 \ \ [m/s]$, $U_R = 0 \ \ [m/s]$
Reynolds number	:	$Re = \dfrac{\rho_L L}{\mu_0 \sqrt{P_L/\rho_L}} = 2 \times 10^3, 1 \times 10^4, 1 \times 10^5$

Numerical aspect:

Wavelet	: $p = 6$
Derivative Approx	: $n = 4$ (5 point-stencil)
Coarsest level	: $j_0 = 4$
Threshold parameters	: $\varepsilon = 1 \times 10^{-4}, 5 \times 10^{-5}, 1 \times 10^{-5}$
Time discretization	: 4^{th} order Runge-Kutta

$$\Delta t = 0.8 \cdot \left(\frac{|u|}{\Delta x} + \sqrt{\frac{\gamma P}{\rho}} \frac{1}{\Delta x} + \max \left[\frac{4}{3}\mu, \kappa \frac{\gamma}{(\gamma - 1) Pr} \right] \frac{1}{\rho Re \Delta x^2} \right)^{-1}$$

Result of simulation for $Re = 1 \times 10^5$

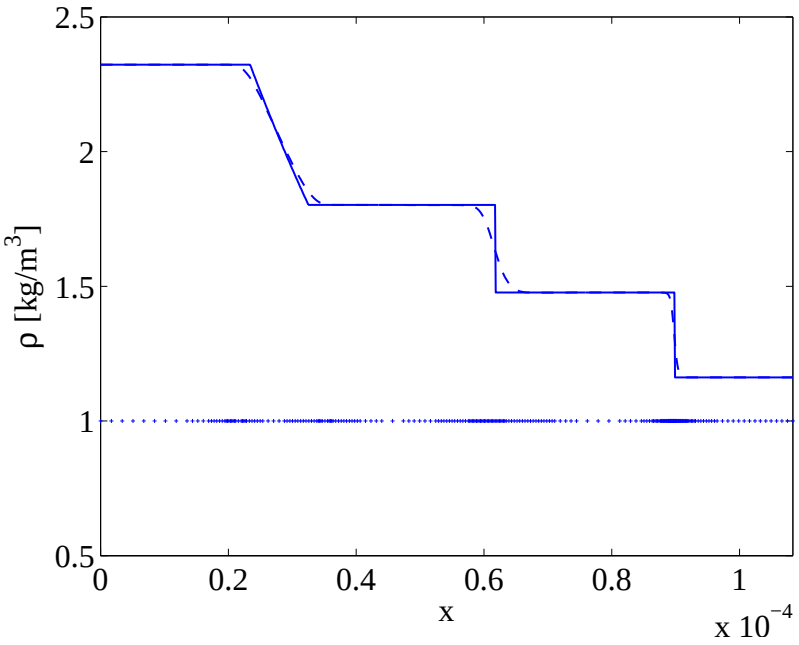


Threshold : $\varepsilon = 1 \times 10^{-5}$
Adaptive scheme : $p = 6\ n = 4$
Resolution : $j_0 = 4,$
: $J - j_0 = 15$

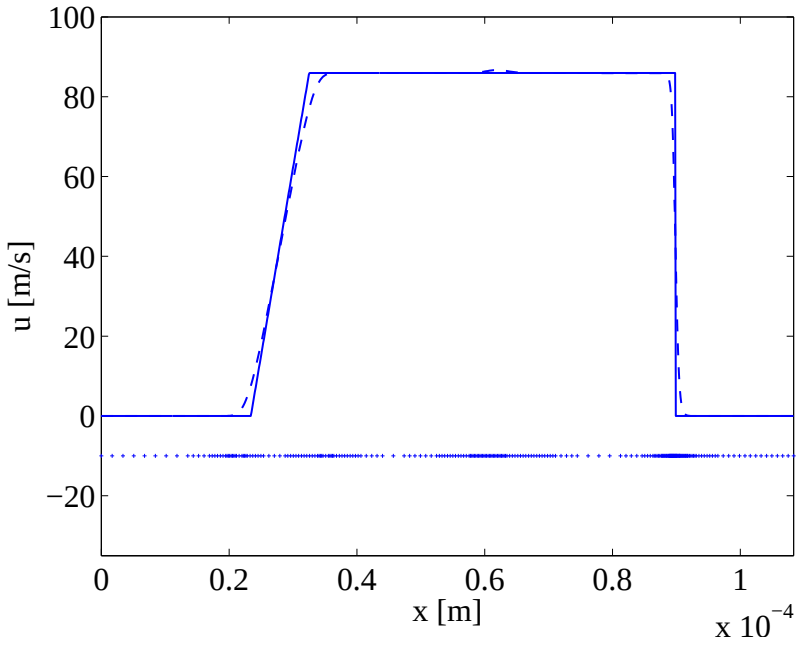
Comparison with the Inviscid Solution: $Re = 2 \times 10^3$

$t = 8.86 \times 10^{-8}$ $[s]$, $N = 249$

DENSITY



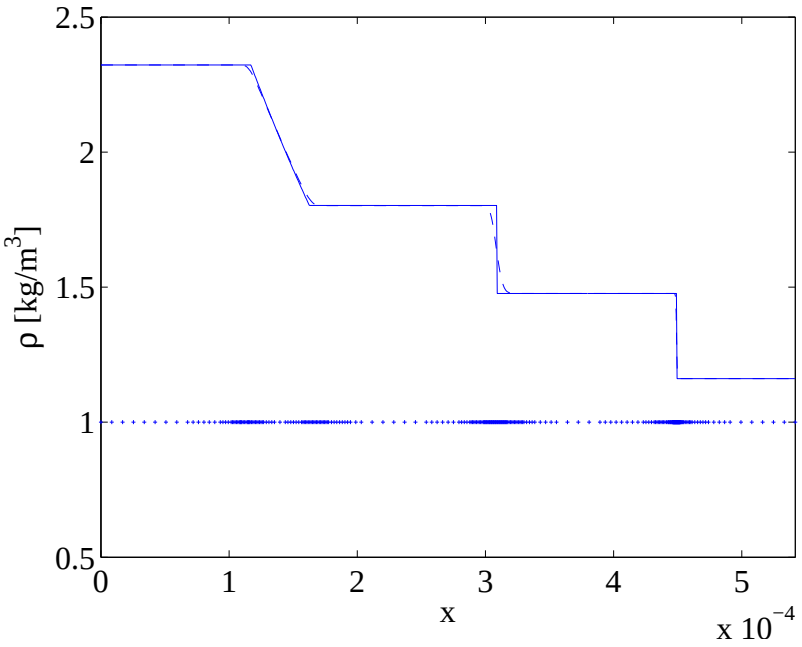
VELOCITY



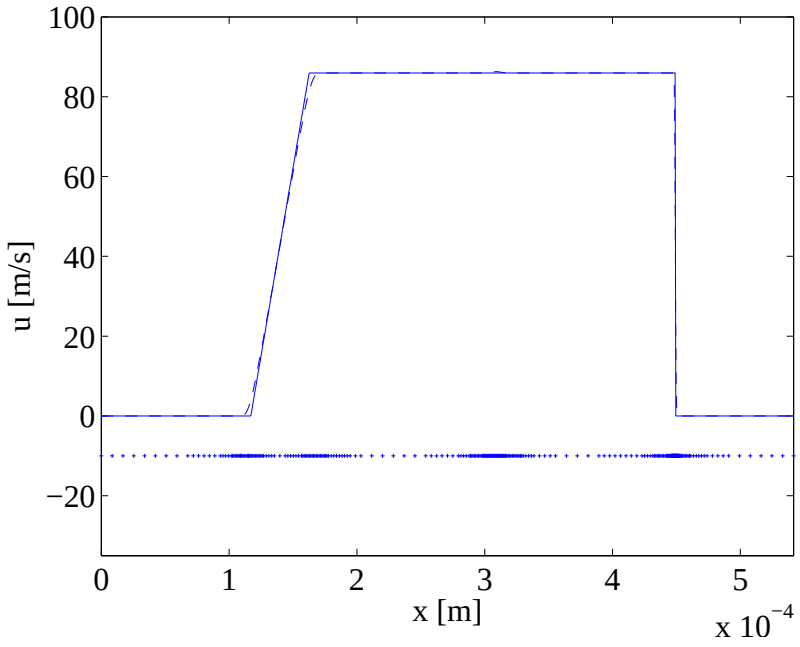
Comparison with the Inviscid Solution: $Re = 1 \times 10^4$

$t = 4.43 \times 10^{-7}$ $[s]$, $N = 291$

DENSITY



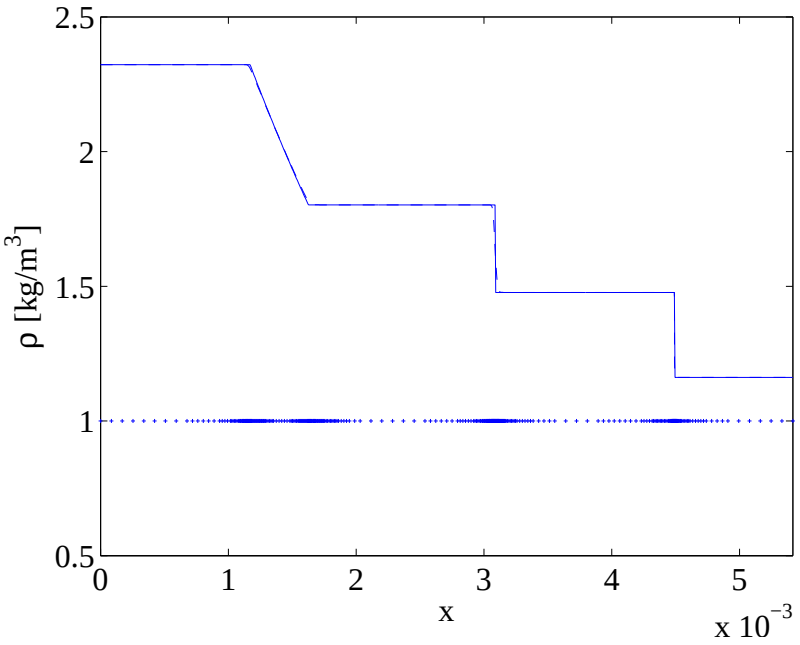
VELOCITY



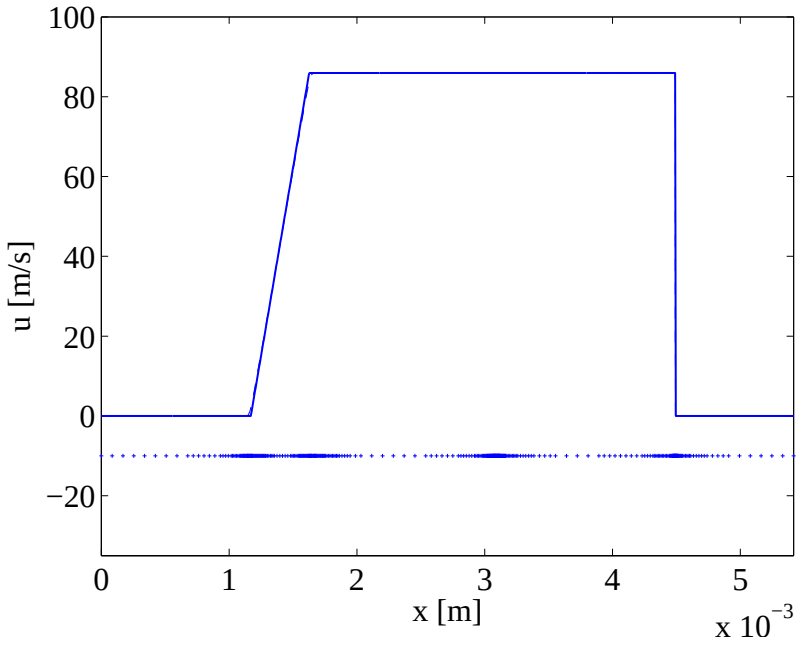
Comparison with the Inviscid Solution: $Re = 1 \times 10^5$

$$t = 4.43 \times 10^{-6}\ [s],\ N = 435$$

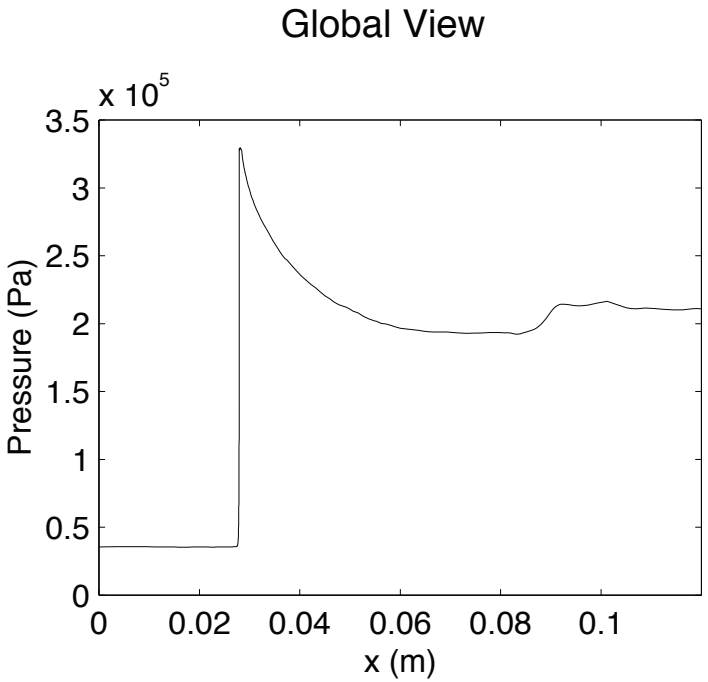
DENSITY

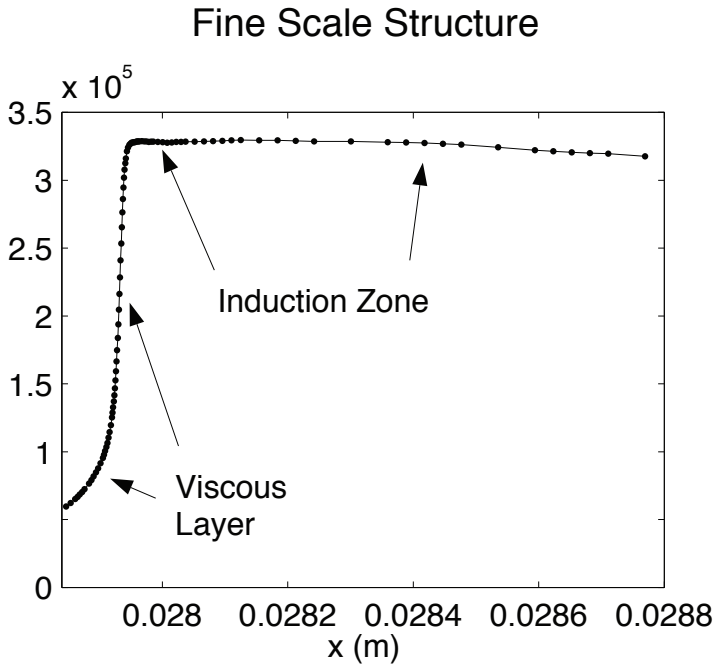


VELOCITY



Viscous $H_2/O_2/Ar$ Detonation with Wavelets





Rastigejeev, Singh, Powers, Paolucci (*Combust. Theory Model.*, 2001)

Conclusions

- Hypersonic reacting flows with detailed kinetics admit a challenging variety of length scales, spanning over five orders of magnitude.
- An adaptive wavelet algorithm for solving PDEs in d-dimension has been described. The algorithm is based on d-dimensional interpolating wavelets.
- The method has been applied to solve a low Mach number flameball-vortex interaction (a diffusion-advection-reaction problem) which constitutes a challenging problem due to the evolving thin layers.
- The method has been applied to a one-dimensional viscous Sod shock tube and viscous $H_2/O_2/Ar$ detonation.
- Numerical results clearly indicate that the adaptive method is able to follow the structures of the solutions dynamically.

Wavelet Approximation in Domain $[0,1]^d$

Approximation of $u(\mathbf{x})$ by the interpolating wavelet, a multiscale basis, on $[0,1]^d$ is given by

$$u(\mathbf{x}) \approx u^J(\mathbf{x}) = \sum_{\mathbf{k}} u_{j_0,\mathbf{k}} \Phi_{J_0,\mathbf{k}}(\mathbf{x}) + \sum_{j=J_0}^{J-1} \sum_{\lambda} d_{j,\lambda} \Psi_{j,\lambda}(\mathbf{x}),$$

where $\mathbf{x} \in \mathbf{R}^d$, $\lambda = (\mathbf{e}, \mathbf{k})$ and $\Psi_{j,\lambda}(\mathbf{x}) \equiv \Psi^{\mathbf{e}}_{j,\mathbf{k}}(\mathbf{x})$.

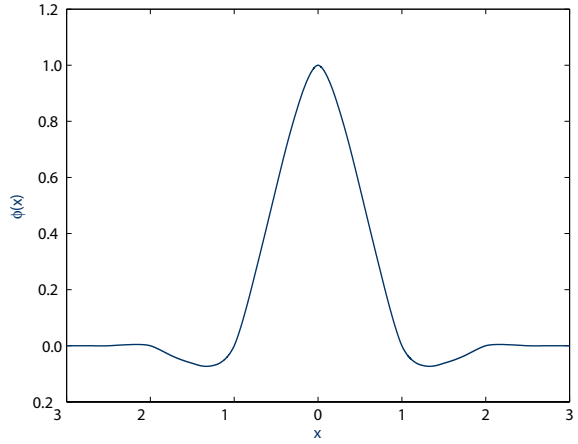
- Scaling function:
$$\Phi_{j,\mathbf{k}}(\mathbf{x}) = \prod_{i=1}^{d} \phi_{j,\mathbf{k}}(x_i),\ k_i \in \kappa_j^0$$

- Wavelet function:
$$\Psi^{\mathbf{e}}_{j,\mathbf{k}}(\mathbf{x}) = \prod_{i=1}^{d} \psi^{e_i}_{j,\mathbf{k}}(x_i),\ k_i \in \kappa_j^{e_i}$$

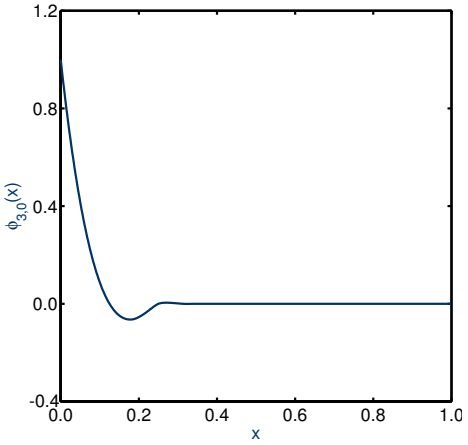
where $\mathbf{e} \in \{0,1\}^d \setminus \mathbf{0}$, $\psi^0_{j,k}(x) \equiv \phi_{j,k}(x)$ and $\psi^1_{j,k}(x) \equiv \psi_{j,k}(x)$, and $\kappa_j^0 = \{0, \cdots, 2^j\}$ and $\kappa_j^1 = \{0, \cdots, 2^j - 1\}$.

Example: 1-D interpolating scaling functions $\phi_{3,k}$, $k = 0, 1, 2,$ and 4 for $p = 4$

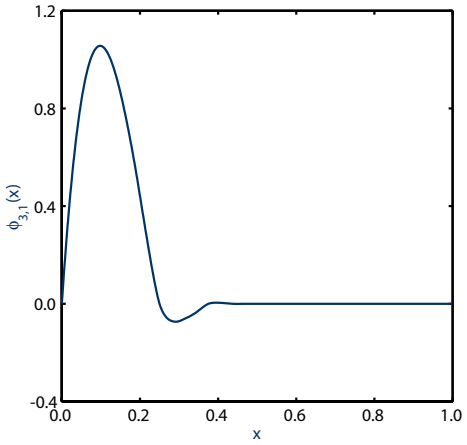
$\phi(x)$ with $p = 4$



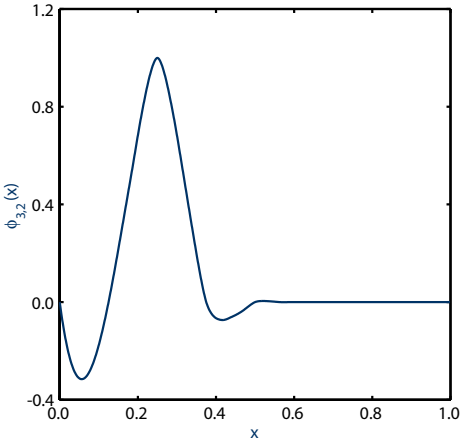
$\phi_{3,0}(x)$



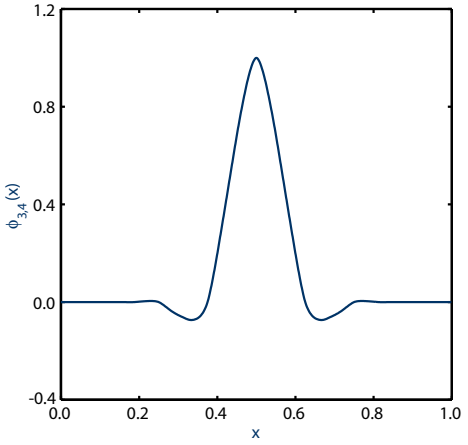
$\phi_{3,1}(x)$



$\phi_{3,2}(x)$



$\phi_{3,4}(x)$



1-D Interpolating Scaling Function and Wavelet

Some properties of $\phi_{j,k}$ and $\psi_{j,k}$ of order p ($p \in \mathbf{N}$, even):

- $\phi_{j,k}$ is defined through $\phi(2^j x - k)$ where $\phi(x) = \int \varphi_p(y)\varphi_p(y-x)dy$, the auto-correlation of the Daubechies wavelet $\varphi_p(x)$.
- The support of $\phi_{j,k}$ is compact, *i.e.* $\text{supp}\{\phi_{j,k}\} \sim |O(2^{-j})|$.
- $\phi_{j,k}(x_{j,n} = n2^{-j}) = \delta_{k,n}$, *i.e.* satisfies the *interpolation property.*
- $\psi_{j,k} = \phi_{j+1,2k+1}$.
- $\text{span}\{\phi_{j,k}\} = \text{span}\{\{\phi_{j-1,k}\}, \{\psi_{j-1,k}\}\}$.
- $\{1, x, \cdots, x^{p-1}\}$, for $x \in [0,1]$, can be written as a linear combination of $\{\phi_{j,k},\ k = 0, \cdots, 2^j\}$.
- $\{\{\phi_{J_0,k}\}, \{\psi_{j,k}\}_{j=J_0}^{\infty}\}$ forms a basis of a continuous 1-D function on the unit interval $[0,1]$.

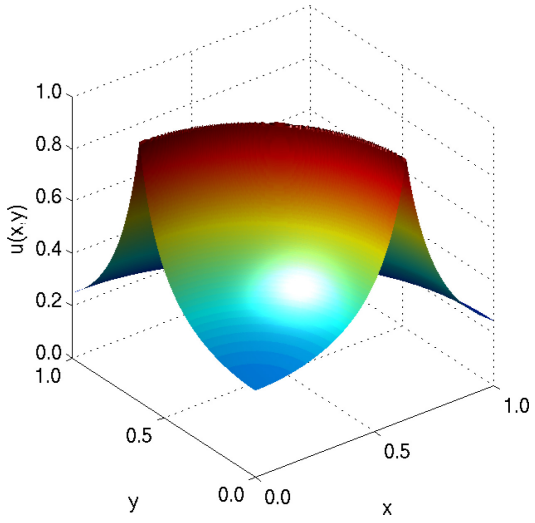
Wavelet Amplitudes

- Wavelet amplitude, $|d_{j,\lambda}|$, measures an error of the approximation of $u(\mathbf{x})$ by a local polynomial approximation at the point $\mathbf{x}_{j,\lambda}$.

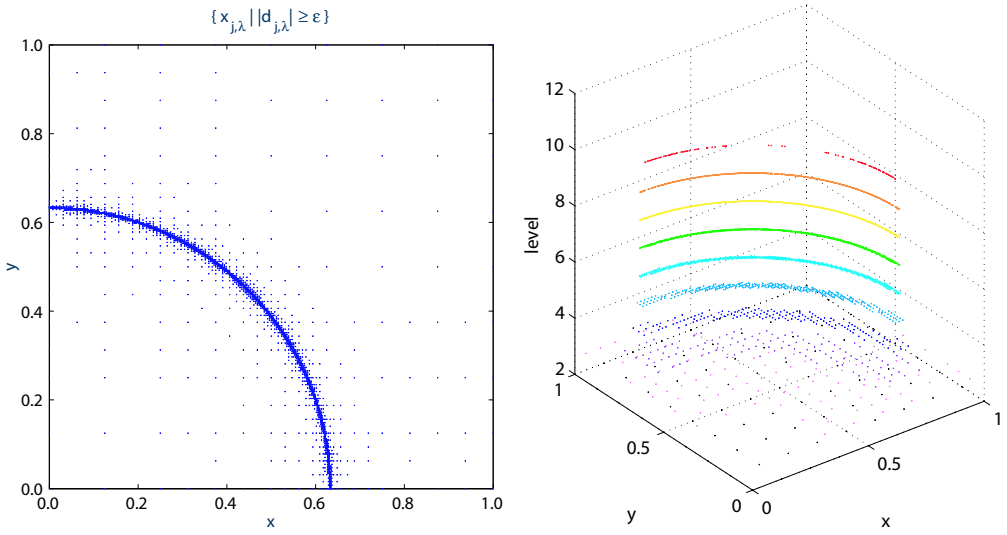
- In other words, wavelet amplitudes, $d_{j,\boldsymbol{\lambda}}$, indicate the local regularity of a function.

Example: Consider $u(x,y) = 0.2/(|0.4 - x^2 - y^2| + 0.2)$

$u(x,y)$



Associated sparse grid



Grid points correspond to wavelet amplitudes that are larger than $\varepsilon = 5 \times 10^{-3}$.

Sparse Wavelet Representation (SWR) and Irregular Sparse grid

- For a given threshold parameter ε, the multiscale approximation of a function $u(\mathbf{x})$ can be written as

$$u^J(\mathbf{x}) = \sum_{\mathbf{k}} u_{J_0,\mathbf{k}} \Phi_{j_0,\mathbf{k}}(\mathbf{x}) + \sum_{j=j_0}^{J-1} \sum_{\{\boldsymbol{\lambda} \,:\, |d_{j,\boldsymbol{\lambda}}| \geq \varepsilon\}} d_{j,\boldsymbol{\lambda}} \Psi_{j,\boldsymbol{\lambda}}(\mathbf{x})$$

$$+ \underbrace{\sum_{j=j_0}^{J-1} \sum_{\{\boldsymbol{\lambda} \,:\, |d_{j,\boldsymbol{\lambda}}| < \varepsilon\}} d_{j,\boldsymbol{\lambda}} \Psi_{j,\boldsymbol{\lambda}}(\mathbf{x})}_{R_\varepsilon^J}$$

- The Sparse Wavelet Representation (SWR) is obtained by discarding the term R_ε^J:

$$u_\varepsilon^J(\mathbf{x}) = \sum_{\mathbf{k}} u_{j_0,\mathbf{k}} \Phi_{j_0,\mathbf{k}}(\mathbf{x}) + \sum_{j=j_0}^{J-1} \sum_{\{\boldsymbol{\lambda} \,:\, |d_{j,\boldsymbol{\lambda}}| \geq \varepsilon\}} d_{j,\boldsymbol{\lambda}} \Psi_{j,\boldsymbol{\lambda}}(\mathbf{x}).$$

SWR and Irregular Sparse Grid (continued)

- For interpolating wavelets, each basis function is associated with one dyadic grid point, *i.e.*

$$\Phi_{j,\mathbf{k}}(\mathbf{x}) \quad \text{with} \quad \mathbf{x}_{j,\mathbf{k}} = (k_1 2^{-j}, \ldots, k_d 2^{-j})$$
$$\Psi_{j,\lambda}(\mathbf{x}) \quad \text{with} \quad \mathbf{x}_{j,\boldsymbol{\lambda}} = \mathbf{x}_{j+1,2\mathbf{k}+\mathbf{e}}$$

- Thus, for a given SWR, one has an associated grid composing of irregular points

$$\boldsymbol{\mathcal{V}} = \{\mathbf{x}_{j_0,\mathbf{k}}, \bigcup_{j \geq j_0} \mathbf{x}_{j,\boldsymbol{\lambda}} \ : \ \lambda \in \boldsymbol{\Lambda}_j\}, \quad \boldsymbol{\Lambda}_j = \{\lambda \ : \ |d_{j,\lambda}| \geq \varepsilon\}.$$

- There exists an adaptive fast wavelet transform ($AFWT$), with $O(N)$, $N = \dim\{\boldsymbol{\mathcal{V}}\}$ operations, mapping the function values on the irregular grid $\boldsymbol{\mathcal{V}}$ to the associated wavelet coefficients and *vice-versa*:

$$AFWT(\{u(\mathbf{x}) \ : \ \mathbf{x} \in \boldsymbol{\mathcal{V}}\}) \rightarrow \mathcal{D} = \{\{u_{j_0,\mathbf{k}}\}, \{d_{j,\boldsymbol{\lambda}}, \ \boldsymbol{\lambda} \in \boldsymbol{\Lambda}_j\}_{j>j0}\}.$$

SWR and Irregular Sparse grid (continued)

- Provided that the function $u(\mathbf{x})$ is continuous, the error in the SWR $u_\varepsilon^J(\mathbf{x})$ is bounded by

$$\|u - u_\varepsilon^J\|_\infty \leq C_1 \, \varepsilon.$$

- Furthermore, for the function that is smooth enough, the number of basis functions $N = \dim\{u_\varepsilon^J\}$ required for a given ε satisfies

$$N \leq C_2 \, \varepsilon^{-d/p},$$

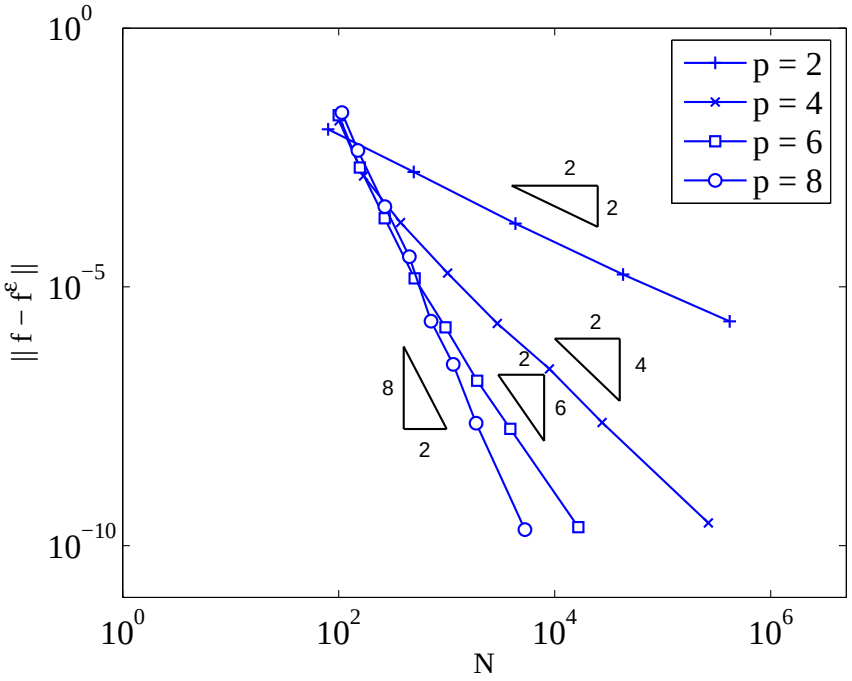
$$\|u - u_\varepsilon^J\|_\infty \leq C_2 \, N^{-p/d}$$

SWR and Irregular Sparse grid (continued)

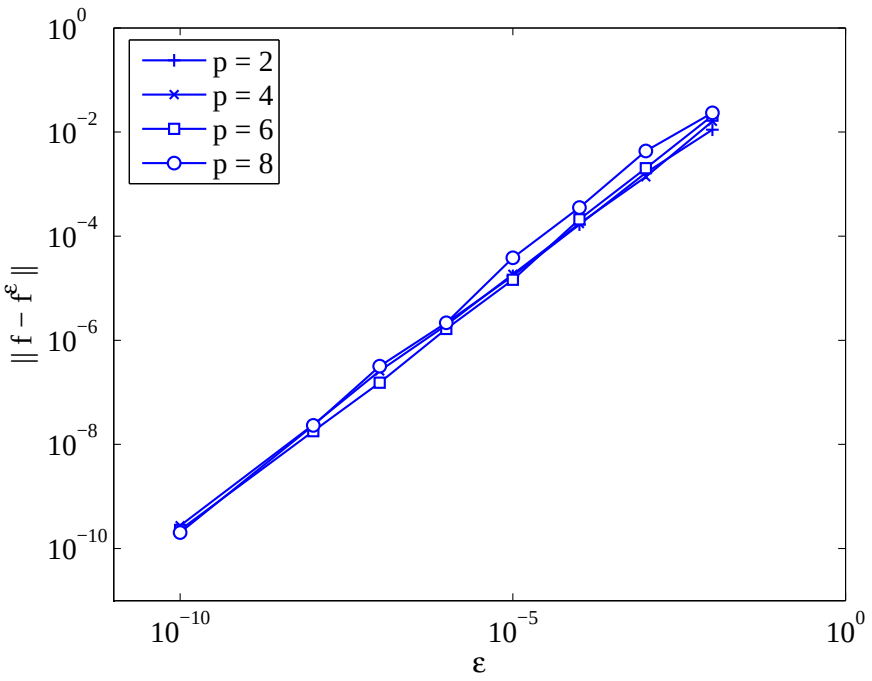
Example: Consider the SWR of

$$u(\mathbf{x}) = \frac{0.05(x + 0.05)}{(x + 0.05)^2 + (y - 0.4)^2}$$

$\|u - u_\varepsilon\|_\infty$ vs. N



ε vs. $\|u - u_\varepsilon\|_\infty$



Derivative Approximation of SWR

- ☞ Direct differentiation of wavelets is costly (with $O(p(J-j_0)N)$ operations) because of different support sizes of wavelet basis on different levels.
- ☞ Alternatively, we use finite differences to approximate the derivative on a grid of irregular points. The procedure can be summarized as follows:
  1. For a given SWR of a function, perform the inverse interpolating wavelet transform to obtain the function values at the associated irregular points.
  2. Apply locally a finite difference scheme of order n to approximate the derivative at each grid point.
- ☞ Estimate shows that the pointwise error of the derivative approximation has the following bound:

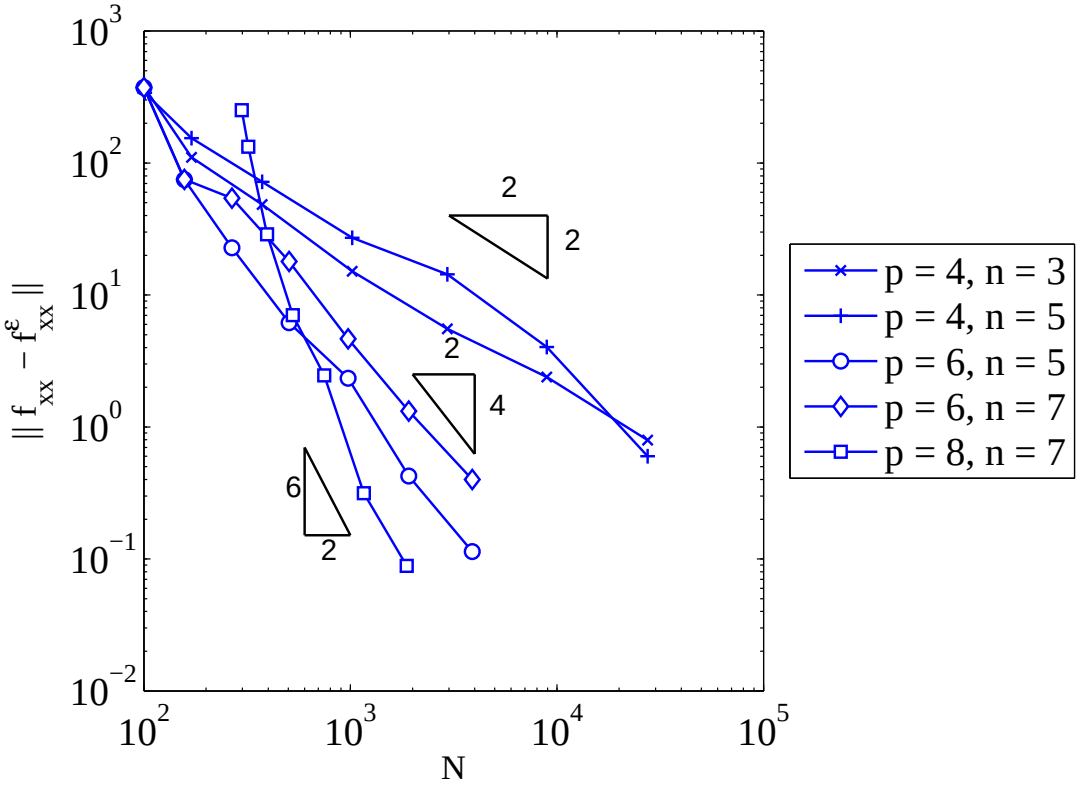
$$\|\partial^i u/\partial x^i - D_x^{(i)} u_\varepsilon^J\|_{\boldsymbol{\nu},\infty} \le C N^{-\min((p-i),n)/2}, \quad \|f\|_{\mathcal{G},\infty} = \max_{\mathbf{x}\in\boldsymbol{\nu}} |f(x)|.$$

Derivative Approximation of SWR

Example: Consider a derivative approximation of the SWR of

$$u(\mathbf{x}) = \frac{0.05(x + 0.05)}{(x + 0.05)^2 + (y - 0.4)^2}.$$

$\|u - D_x^{(2)} u_\varepsilon\|_\infty$ vs. N



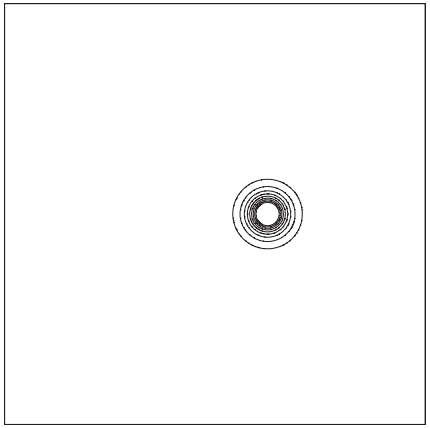
Snap shots for $\Gamma = 100$ with $p = 6$, $n = 4$ and $\varepsilon = 10^{-3}$

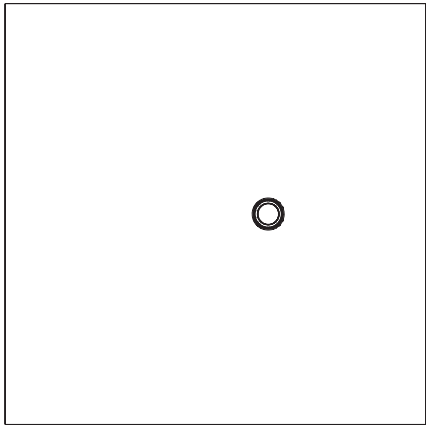
T

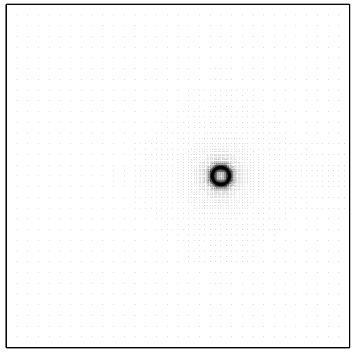
$\omega(T, Y, t)$

Adaptive grid

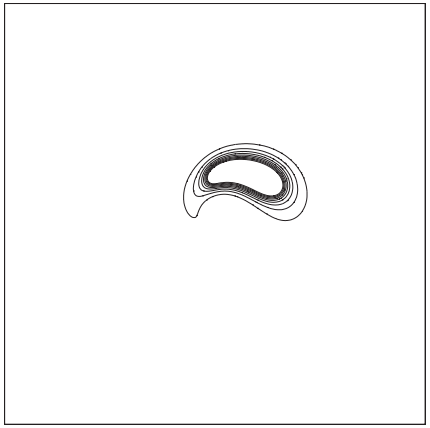
$t = 0$

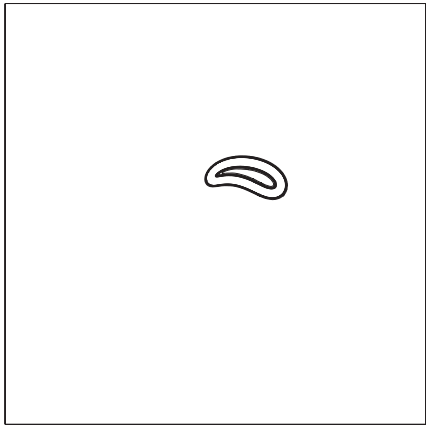


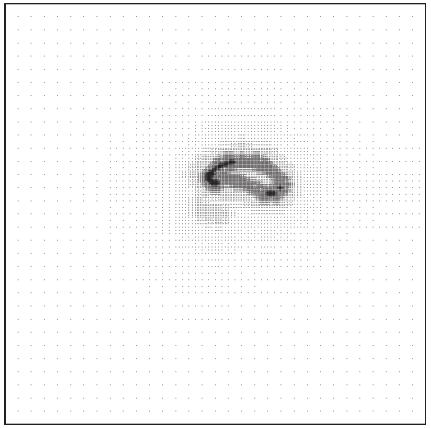




$t = 0.33$







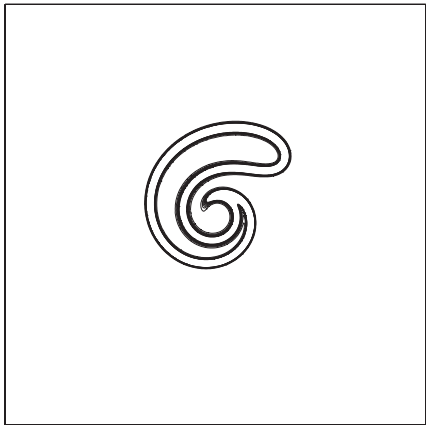
Snap shots for $\Gamma = 100$ with $p = 6$, $n = 5$ and $\varepsilon = 10^{-3}$

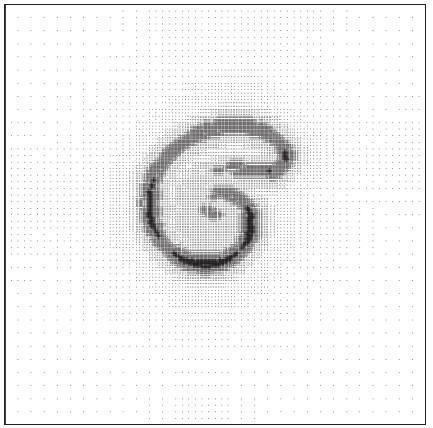
T

$\omega(T, Y, t)$

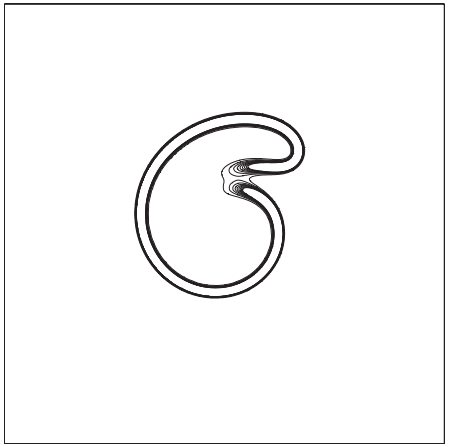
Adaptive grid

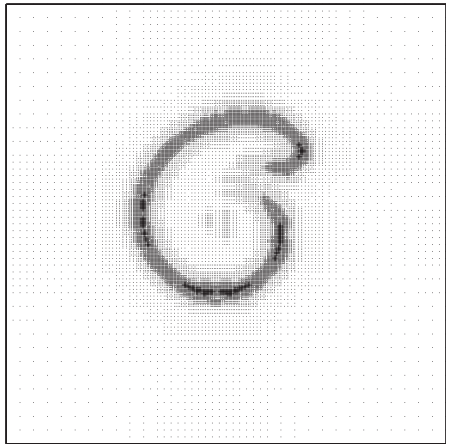
$t = 0.79$





$t = 1.0$





Model problem

Consider an advection-diffusion problem:

$$\frac{\partial u}{\partial t} + (\mathbf{V} \cdot \nabla)u = \nu \nabla^2 u + f(\mathbf{x}, t), \quad (\mathbf{x}, t) \in \Omega \times [0, +\infty), \Omega = (0, 1)^2$$

$$u = g, \quad \mathbf{x} \in \partial\Omega \quad \text{and} \quad u(\mathbf{x}, 0) = q(\mathbf{x})$$

- $\mathbf{V} = (1, 0)^T$, $\nu = 1/100$ and f, g, and q are chosen so that the exact solution is defined by

$$u(\mathbf{x}, t) = \frac{0.05(x + 0.05)}{(x + 0.05)^2 + (y - t)^2}$$

Numerical experiments

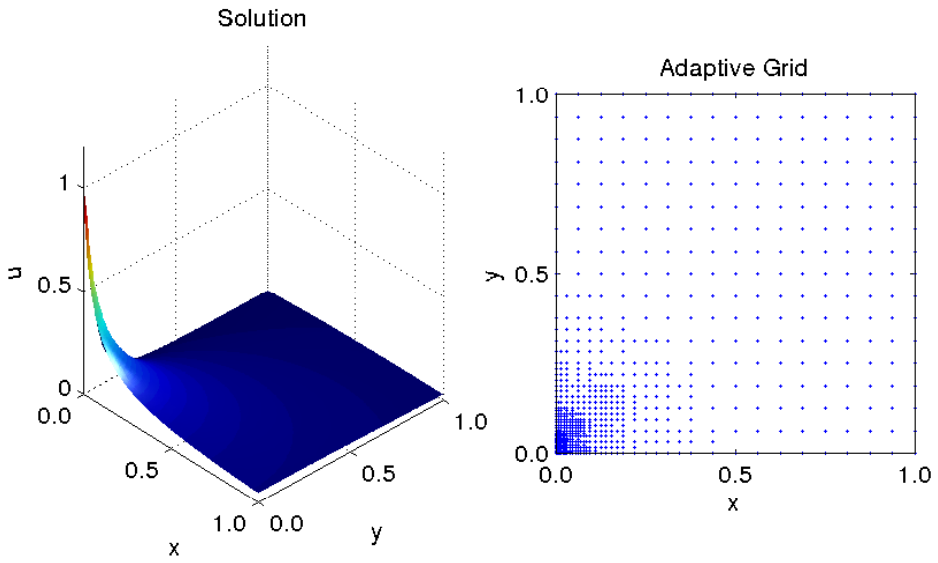
Time discretization	: Trapezoidal scheme (Crank-Nicolson)
	: with small Δt, $t_f = 1$
Adaptive method	: $p = 4$ and $n = 4$, $p = 6$ and $n = 4$,
	: and $p = 8$ and $n = 7$
Coarsest levels	: $j_0 = 3$
Maximum levels of refinement	: $J - j_0 = 11$
Threshold	: $\varepsilon = 5 \times 10^{-3}$ to 1×10^{-7}

☞ The numerical solution as ε varied is expected to conform reasonably with

$$N(t_m) = \dim\{\boldsymbol{\mathcal{V}}^{t_m}\} = O(\varepsilon^{-d/p})$$

$$\|u_{ext}(t^m) - u_\varepsilon(t^m)\|_{\boldsymbol{\mathcal{V}}^{t_m},\infty} = O(N^{-\min(p-2,n)/d}) = O(\varepsilon^{\min(p-2,n)/p})$$

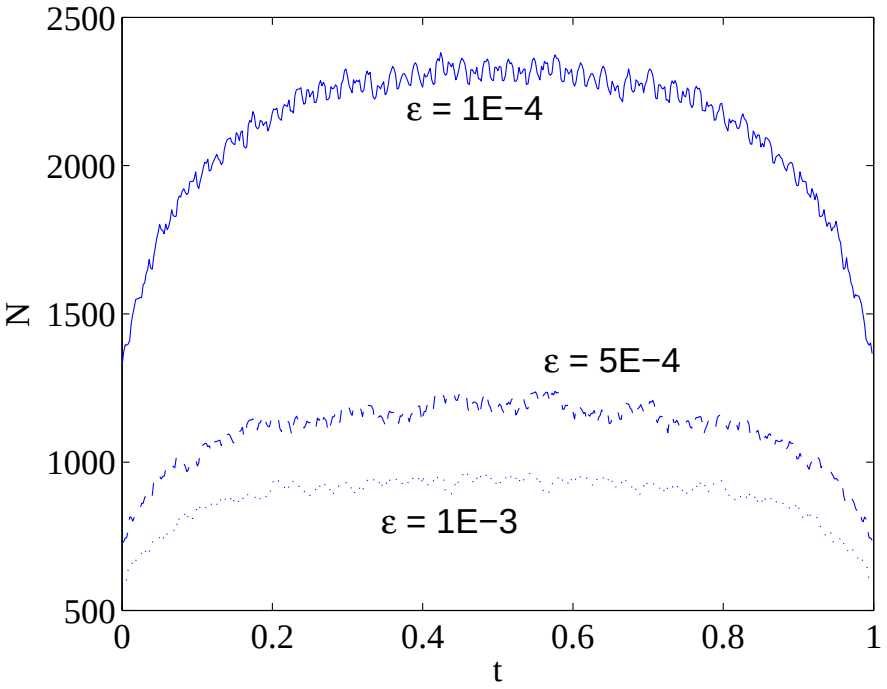
Results of simulation



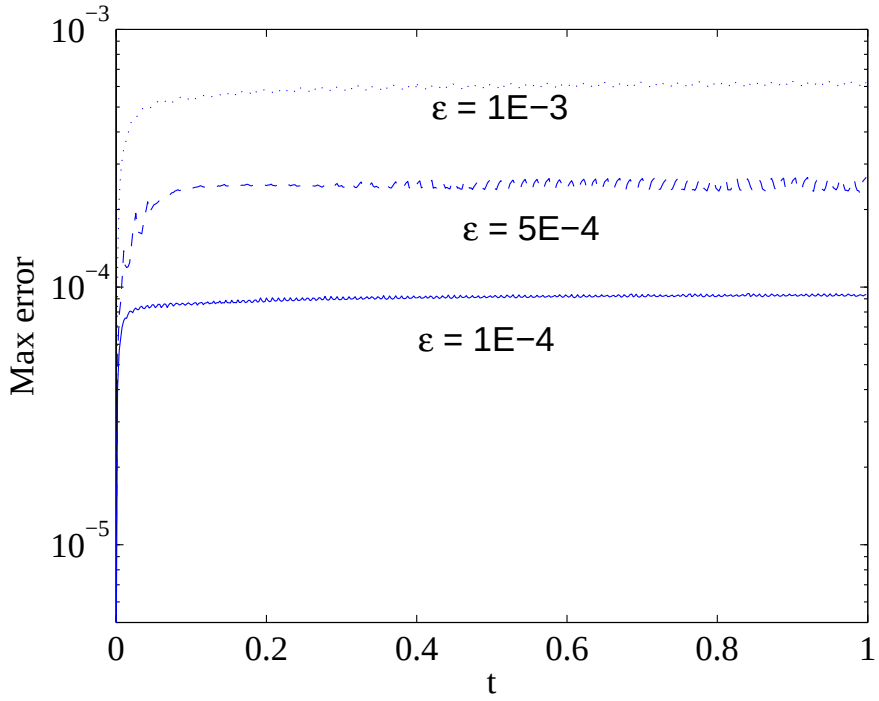
Threshold	: $\varepsilon = 5 \times 10^{-4}$
Adaptive method	: $p = 4$ and $n = 4$
Coarsest level	: $j_0 = 3$

Results for $p = 4$ and $n = 4$

N vs. t



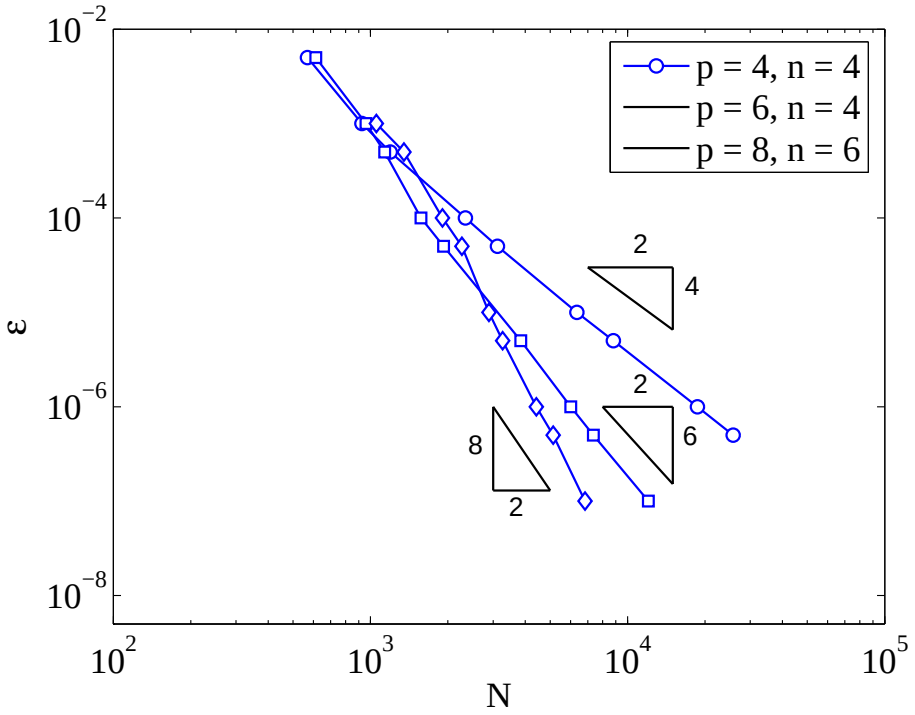
$\|u_{ext} - u_\varepsilon\|_{\boldsymbol{\nu},\infty}$ vs. t



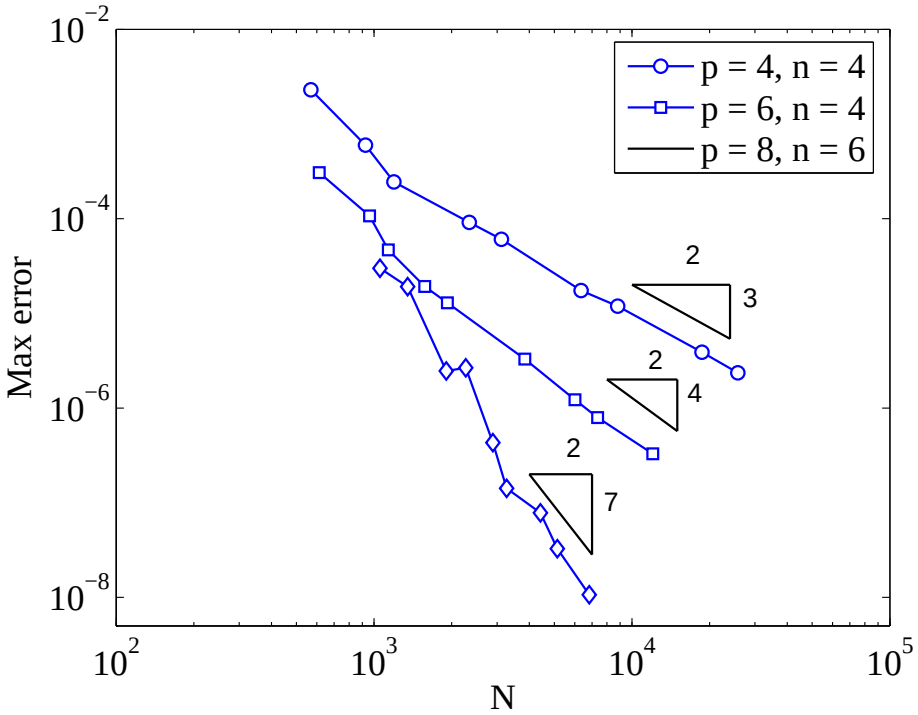
- As threshold value ε decrease, the number of grid points demanded by the algorithm increase automatically.
- The discrete maximum error in the numerical solution stay almost constant in time.

Error in solutions at $t = 0.5$

N vs. ε



N vs. $\|u_{ext} - u_\varepsilon\|_{\boldsymbol{\nu},\infty}$



- N is proportional $O(\varepsilon^{-p/d})$
- $\|u_{ext} - u_\varepsilon\|_{\boldsymbol{\nu},\infty} = O(N^{-c_1})$, $c_1 \geq \min(p-2, n)/2$.
- Furthermore, $\|u_{ext} - u_\varepsilon\|_{\boldsymbol{\nu},\infty} = O(\varepsilon^{c_2})$, $c_2 = c_1 d/p$

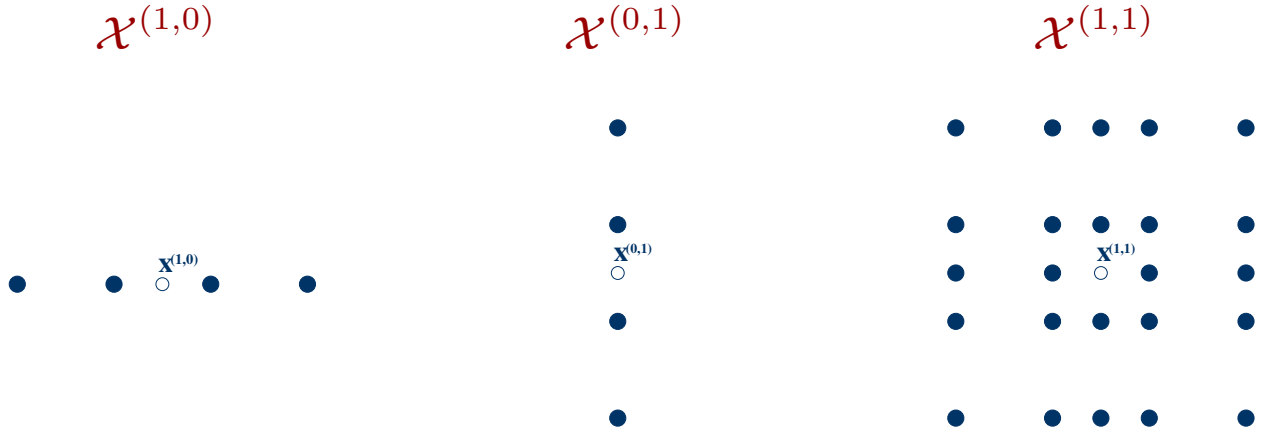
Wavelet Amplitudes

• Let $\mathbf{x}_{j,\lambda} = \mathbf{x}^{\mathbf{e}}_{j,\mathbf{k}} = (x^{e_1}_{j,k_1}, x^{e_2}_{j,k_2}, \cdots, x^{e_d}_{j,k_d})$, where $x^0_{j,k} = k2^{-j}$ and $x^1_{j,k} = (2k+1)2^{-j}$, the interpolating wavelet coefficients is defined by

$$d_{j,\lambda} = d^{\mathbf{e}}_{j,\mathbf{k}} = f(\mathbf{x}_{j,\lambda}) - (\mathcal{P}_{\mathbf{x}^e_{j,\lambda}} f)(\mathbf{x}_{j,\lambda}) \tag{2}$$

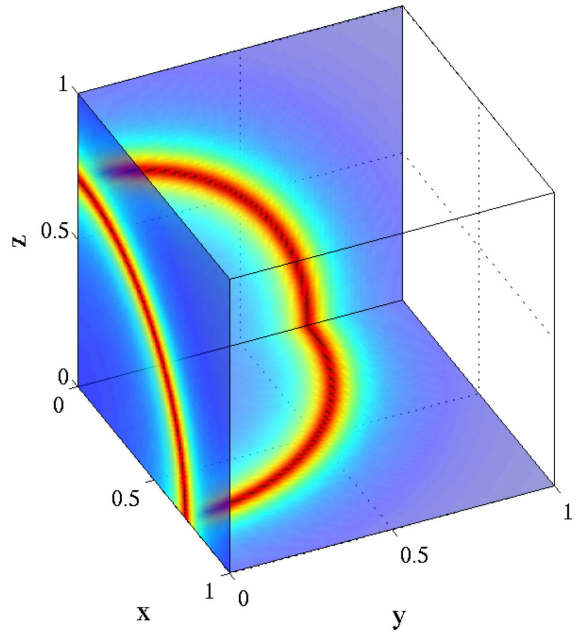
• $(\mathcal{P}_{\mathcal{X}^e_{j,\lambda}} f)(\mathbf{x}_{j,\lambda})$ is a prediction of $f(\mathbf{x}_{j,\lambda})$ with a polynomial defined from a certain set of points $\mathcal{X}^{\mathbf{e}}_{j,\mathbf{k}} \in \{\mathbf{x}_{j,\mathbf{k}}\} \backslash \mathbf{x}_{j,\lambda}$, where $\mathbf{x}_{j,\mathbf{k}} = (k_1, k_2, \cdots, k_d)2^{-j}$.

Examples of sets of points defining prediction polynomials $(\mathcal{P}_{\mathcal{X}^e_{j,\lambda}} f)$ for 2-D wavelet with $p = 4$



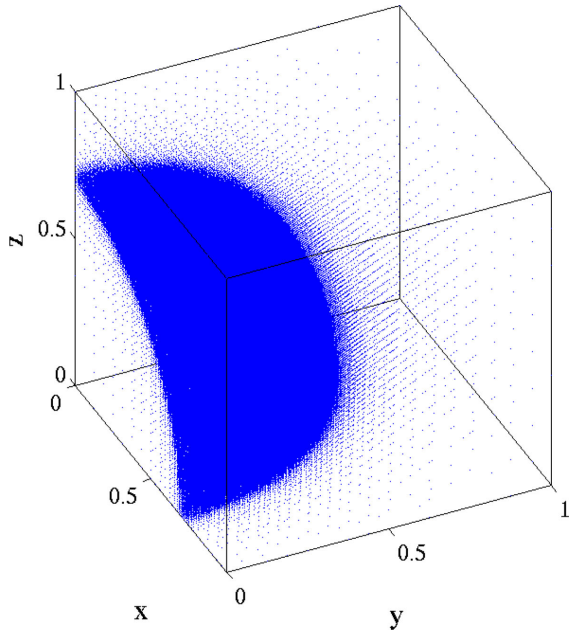
Wavelet Amplitudes

$u = 1/(|0.5 - x^2 - y^2 - z^2| + 0.1)$



Sections of test function

$FWT(u)$



Grid points corresponding to $|d_{j,\lambda}| \geq 5 \times 10^{-3}$.

Note on Algebra of Function Operation

- Addition/subtraction of two functions u_ε^J and v_ε^J can be obtained easily by simply either adding/subtracting their coefficients. The error estimate for this operation is bounded by

$$\|(u \pm v) - (u_\varepsilon^J \pm (v_\varepsilon^J)\| \leq \|u - u_\varepsilon^J\| + \|v - v_\varepsilon^J\| \leq C\varepsilon$$

- Multiplication of two functions u_ε^J and v_ε^J is computed by following
  - use IWT on $\mathcal{D}_u$ and $\mathcal{D}_v$ to get the corresponding functional values on $\mathcal{G}_u \cup \mathcal{G}_v$.
  - Evaluate the multiplication pointwise in the set of points $\mathcal{G}_u \cup \mathcal{G}_v$.

  The error $\|uv - u_\varepsilon^J v_\varepsilon^J\|$ of this procedure is bounded by

$$C_1(\|u\| + \|v_\varepsilon^J\|)\varepsilon \quad \text{and} \quad C_2(\|v\| + \|u_\varepsilon^J\|)\varepsilon.$$

SWR and Irregular Sparse Grid (continued)

- If $\{u(\mathbf{x}), \mathbf{x} \in \boldsymbol{\mathcal{V}}\}$ is the set of exact function values at irregular grid points, $AFWT$ yields the exact values of the associated wavelet coefficients.
- To find interpolated values on an irregular grid $\widetilde{\boldsymbol{\mathcal{V}}}$ that is different from $\boldsymbol{\mathcal{V}}$, simply set up an extended grid $\widehat{\boldsymbol{\mathcal{V}}} = \widetilde{\boldsymbol{\mathcal{V}}} \cup \boldsymbol{\mathcal{V}}$ and an augmented set of coefficients

$$\widehat{\boldsymbol{\mathcal{D}}} = \{\{u_{j_0}\}, \{\widehat{d}_{j,\lambda}\}\} \text{ where } d_{j,\lambda} = \begin{cases} d_{j,\lambda} & \text{for } \mathbf{x}_{j,\lambda} \in \boldsymbol{\mathcal{V}} \\ 0 & \text{for } \mathbf{x}_{j,\lambda} \in \widehat{\boldsymbol{\mathcal{V}}} \setminus \boldsymbol{\mathcal{V}} \end{cases}$$

and run fast inverse wavelet transform ($AIWT$)

$$AIWT(\widehat{\boldsymbol{\mathcal{D}}}) \rightarrow \{\widehat{u}(\mathbf{x}) \ : \ \mathbf{x} \in \widehat{\boldsymbol{\mathcal{V}}}\}$$

Grid Adaption Strategy

- In each refinement step, determine the *essential* grid points, which are points whose associated wavelet amplitudes are larger than the threshold parameter ε:

$$\widehat{\boldsymbol{\nu}}_e = \{\mathbf{x}_{j,\lambda} \ : \ j \geq j_0, \ \lambda \in \boldsymbol{\Lambda}_j, \ |d_{j,\boldsymbol{\lambda}}| \geq \varepsilon\},$$

- To accommodate the possible advection and sharpening of solution features, we determine the *neighboring* grid points:

$$\widehat{\boldsymbol{\nu}}_b = \bigcup_{\{j,\lambda \in \boldsymbol{\Lambda}\}} \mathcal{N}_{j,\lambda},$$

  where $\mathcal{N}_{j,\lambda}$ is a set of neighboring points to $x_{j,\lambda}$.

- The new sparse grid, $\mathcal{V}$, is then given by

$$\boldsymbol{\nu} = \{x_{j_0,\mathbf{k}}\} \cup \widehat{\boldsymbol{\nu}}_e \cup \widehat{\boldsymbol{\nu}}_b.$$