Detonation Simulation Using the Parallel Wavelet Adaptive Multiresolution Representation

By

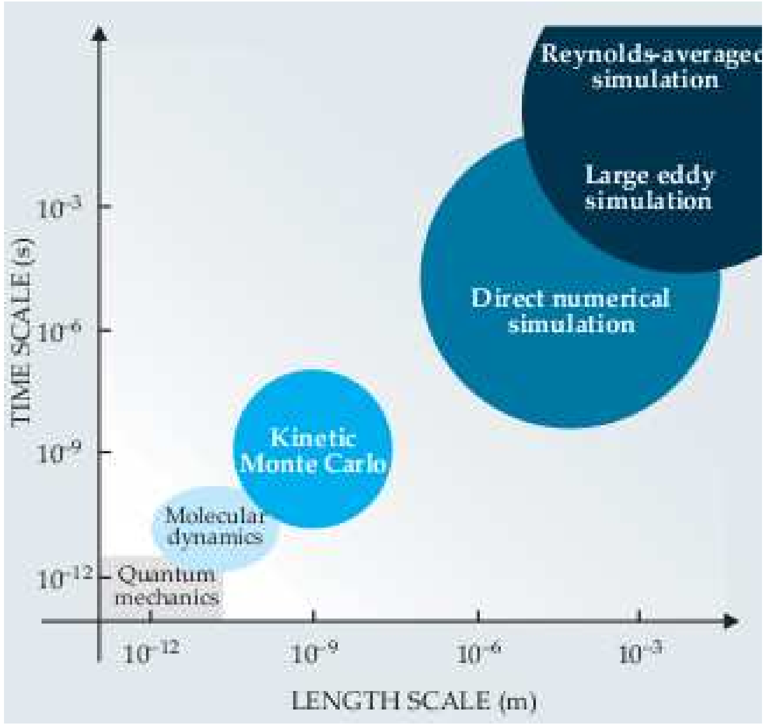
Z. J. Zikoski, S. Paolucci, and J. M. Powers

Aerospace and Mechanical Engineering
University of Notre Dame, Indiana 46556

13th International Conference on Numerical Combustion
Corfu, Greece   April 27-29, 2011

Project Summary

- An adaptive method is applied to the simulation of compressible reacting flow.
- Model includes detailed chemical kinetics, multi-species transport, momentum and energy diffusion.
- Problems are typically multi-dimensional and contain a wide range of spatial and temporal scales.
- Method resolves the range of scales present, while greatly reducing required computational effort and automatically produces verified solutions.



"Research needs for future internal combustion engines," *Physics Today*, Nov. 2008, pp 47-52.

Compressible Reactive Flow

Code solves the n-D compressible reactive Navier-Stokes equations:

$$\begin{aligned}
\frac{\partial \rho}{\partial t} &= -\frac{\partial}{\partial x_i}(\rho u_i) \\
\frac{\partial \rho u_i}{\partial t} &= -\frac{\partial}{\partial x_j}(\rho u_j u_i) - \frac{\partial p}{\partial x_i} + \frac{\partial \tau_{ij}}{\partial x_j} \\
\frac{\partial \rho E}{\partial t} &= -\frac{\partial}{\partial x_j}(u_j(\rho E + p)) + \frac{\partial u_j \tau_{ji}}{\partial x_i} - \frac{\partial q_i}{\partial x_i} \\
\frac{\partial \rho Y_k}{\partial t} &= -\frac{\partial}{\partial x_i}(u_i \rho Y_k) + M_k \dot{\omega}_k - \frac{\partial j_{i,k}}{\partial x_i}, \qquad k = 1, \ldots, K-1
\end{aligned}$$

Where ρ-density, u_i-velocity vector, E-specific total energy, Y_k-mass fraction of species k, τ_{ij}-viscous stress tensor, q_i-heat flux, $j_{i,k}$-species mass flux, M_k-molecular weight of species k, and $\dot{\omega}_k$-reaction rate of species k.

Compressible Reactive Flow (cont.)

Where,

$$
\begin{aligned}
\sum_{k=1}^{K} Y_k &= 1 \\
E &= e + \frac{1}{2} u_i u_i \\
\tau_{ij} &= -\frac{2}{3}\mu \frac{\partial u_l}{\partial x_l}\delta_{ij} + \mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) \\
q_i &= -\mathsf{k}\frac{\partial T}{\partial x_i} + \sum_{k=1}^{K}\left(h_k j_{i,k} - \frac{RT}{m_k X_k} D_k^T d_{i,k}\right) \\
j_{i,k} &= \frac{\rho Y_k}{X_k \overline{M}} \sum_{j=1, j\neq k}^{K} M_j D_{jk} d_{i,j} - \frac{D_k^T}{T}\frac{\partial T}{\partial x_i} \\
d_{i,k} &= \frac{\partial X_k}{\partial x_i} + (X_k - Y_k)\frac{1}{p}\frac{\partial p}{\partial x_i}
\end{aligned}
$$

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

Approximation of $u(\mathbf{x})$ by the interpolating wavelet, a multiscale basis, on $\mathbf{x} \in [0,1]^d$ 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 $\mathbf{x} \in \mathbb{R}^d$, $\lambda = (\mathbf{e}, \mathbf{k})$ 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 $\kappa_j^1 = \{0, \cdots, 2^j - 1\}$.

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

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

  and the SWR is obtained by discarding the term R_ε^J.

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

$$\Phi_{j,\mathbf{k}}(\mathbf{x}) \quad \text{with} \quad \mathbf{x}_{j,\mathbf{k}} = (k_1 2^{-j}, \ldots, k_d 2^{-j})$$

$$\Psi_{j,\lambda}(\mathbf{x}) \quad \text{with} \quad \mathbf{x}_{j,\boldsymbol{\lambda}} = \mathbf{x}_{j+1,2\mathbf{k}+\mathbf{e}}$$

SWR and Irregular Sparse Grid (continued)

- For a given SWR, one has an associated grid composed of *essential* points, whose wavelet amplitudes are greater than the threshold parameter ε

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

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

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

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

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

$$\boldsymbol{\mathcal{V}} = \mathbf{x}_{j_0,k} \cup \boldsymbol{\mathcal{V}}_e \cup \boldsymbol{\mathcal{V}}_b.$$

SWR and Irregular Sparse grid (continued)

- 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*:

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

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

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

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

$$N \le C_2\ \varepsilon^{-d/p}, \quad \text{and} \quad \|u - u_\varepsilon^J\|_\infty \le C_2\ N^{-p/d}.$$

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 the connection with Lagrange interpolating polynomials 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:

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

Dynamically Adaptive Algorithm for Solving Time-Dependent PDEs

Given the set of PDEs

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

with initial conditions

$$u(x, 0) = u^0.$$

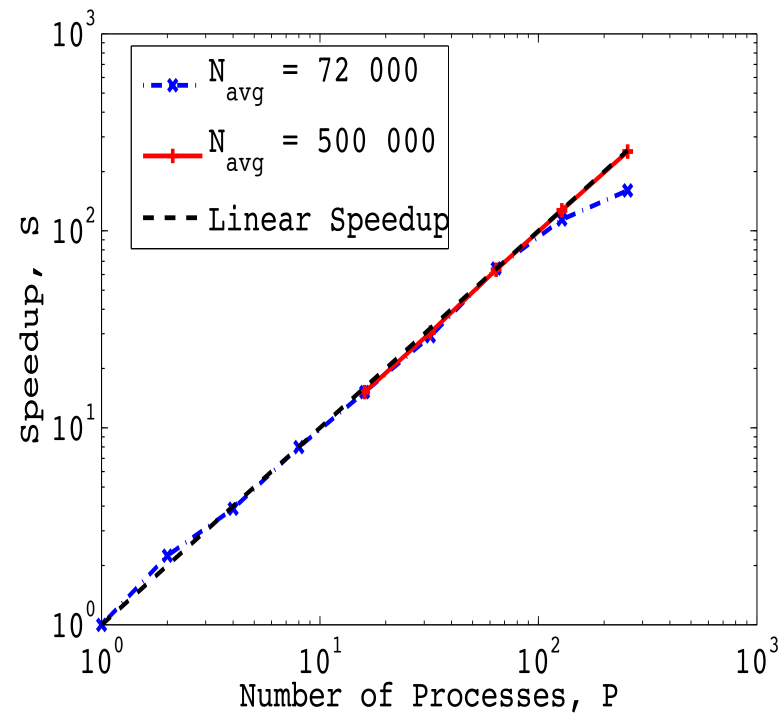
❶ Obtain sparse grid, $\boldsymbol{\mathcal{V}}^m$, based on thresholding of magnitudes of wavelet amplitudes of the approximate solution u^m.

❷ Integrate in time using an explicit time integrator with error control to obtain the new solution u^{m+1}.

❸ Assign $u^{m+1} \to u^m$ and return to step ❶.

Parallelization

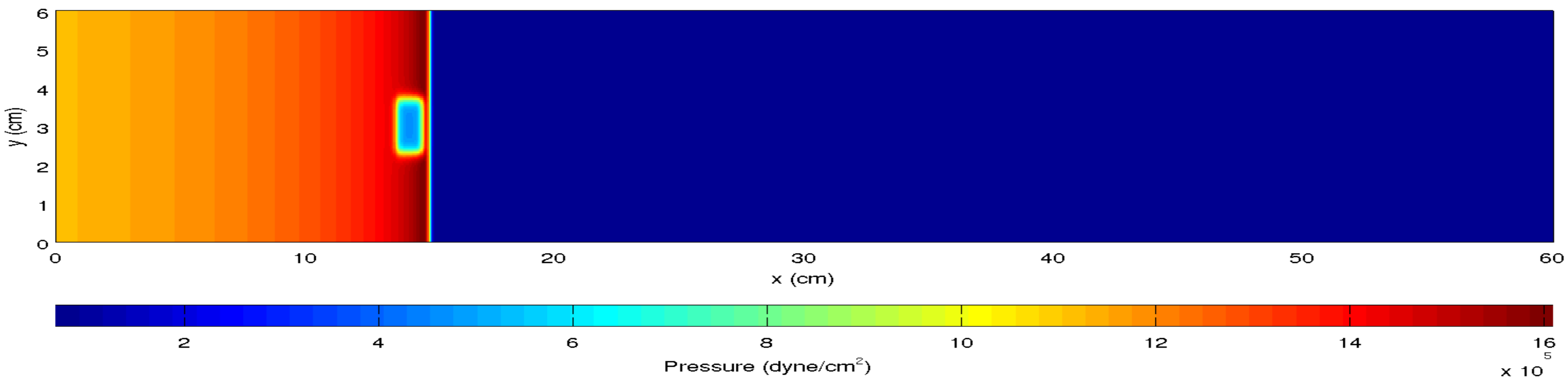
- Parallel algorithm uses an MPI-based domain decomposition.
- Hilbert space-filling curve used for partitioning and load-balancing.
- Strong scaling up to 256 cores with $> 90\%$ parallel efficiency.
- Chemkin-II and Transport Libraries used for evaluation of thermodynamics, transport properties, and reaction source terms.



Dual Quad-Core, 2.7 GHz L5520 Intel Nehalem nodes (8 cores/node), 12 GB RAM, Infiniband interconnect

2-D Viscous Detonation

Initial Conditions:



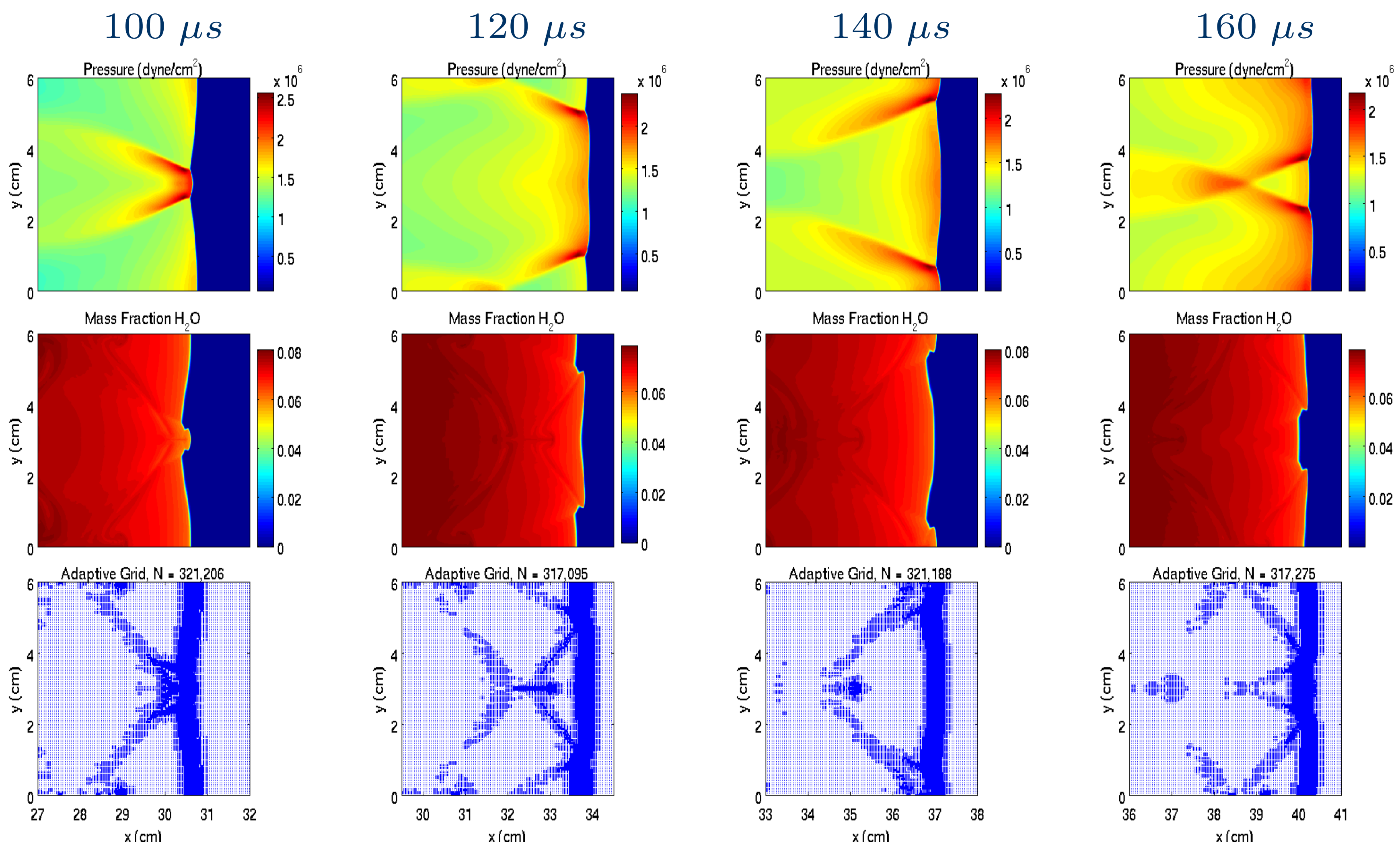
Domain: $[0, 60] \times [0, 6]$ cm
Front: $x = 15.0$ cm
Unreacted pocket:
$[1.05 \times 1.43]$ cm
at $x = 14.7$ cm
$P = 4.7 \times 10^5$ dyne/cm^2
$T = 2100$ K
128 cores
391 hrs runtime

$2H_2 : O_2 : 7Ar$ mixture
9 species, 37 reactions

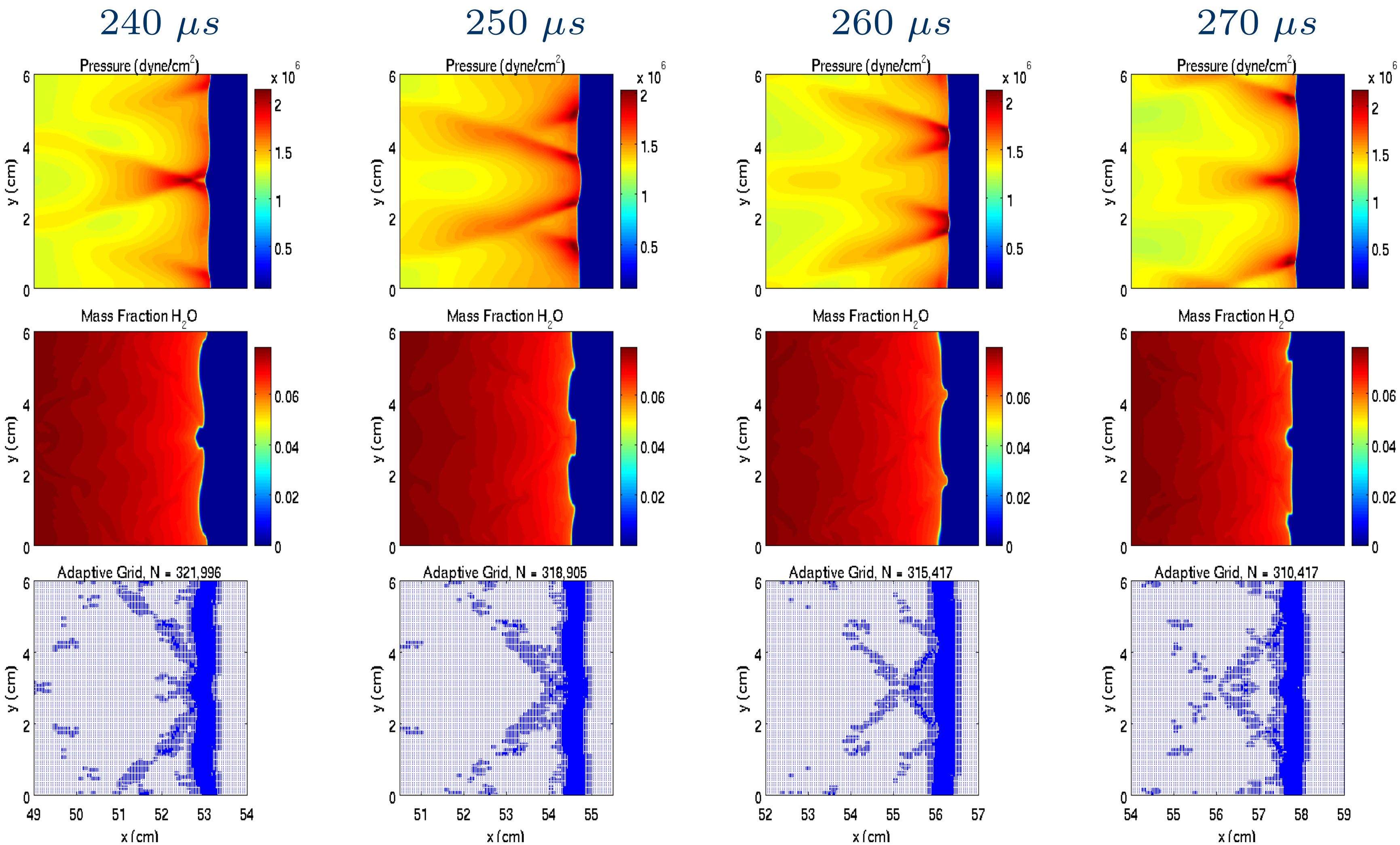
Wavelet parameters:

$\epsilon = 1 \times 10^{-3}$
$p = 6, \quad n = 5$
$[N_x \times N_y]_{j_0} = [600 \times 60]$
$J - j_0 = 10$

2-D Viscous Detonation (cont.)

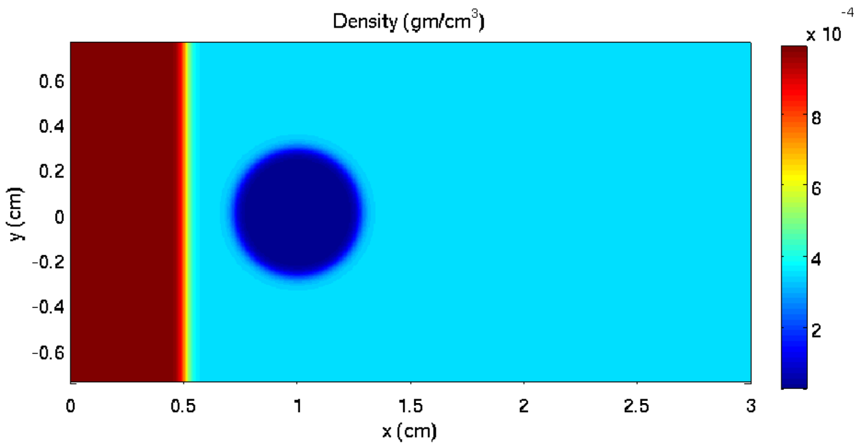


2-D Viscous Detonation (cont.)



Shock/H_2-Bubble Interaction

Initial Conditions:



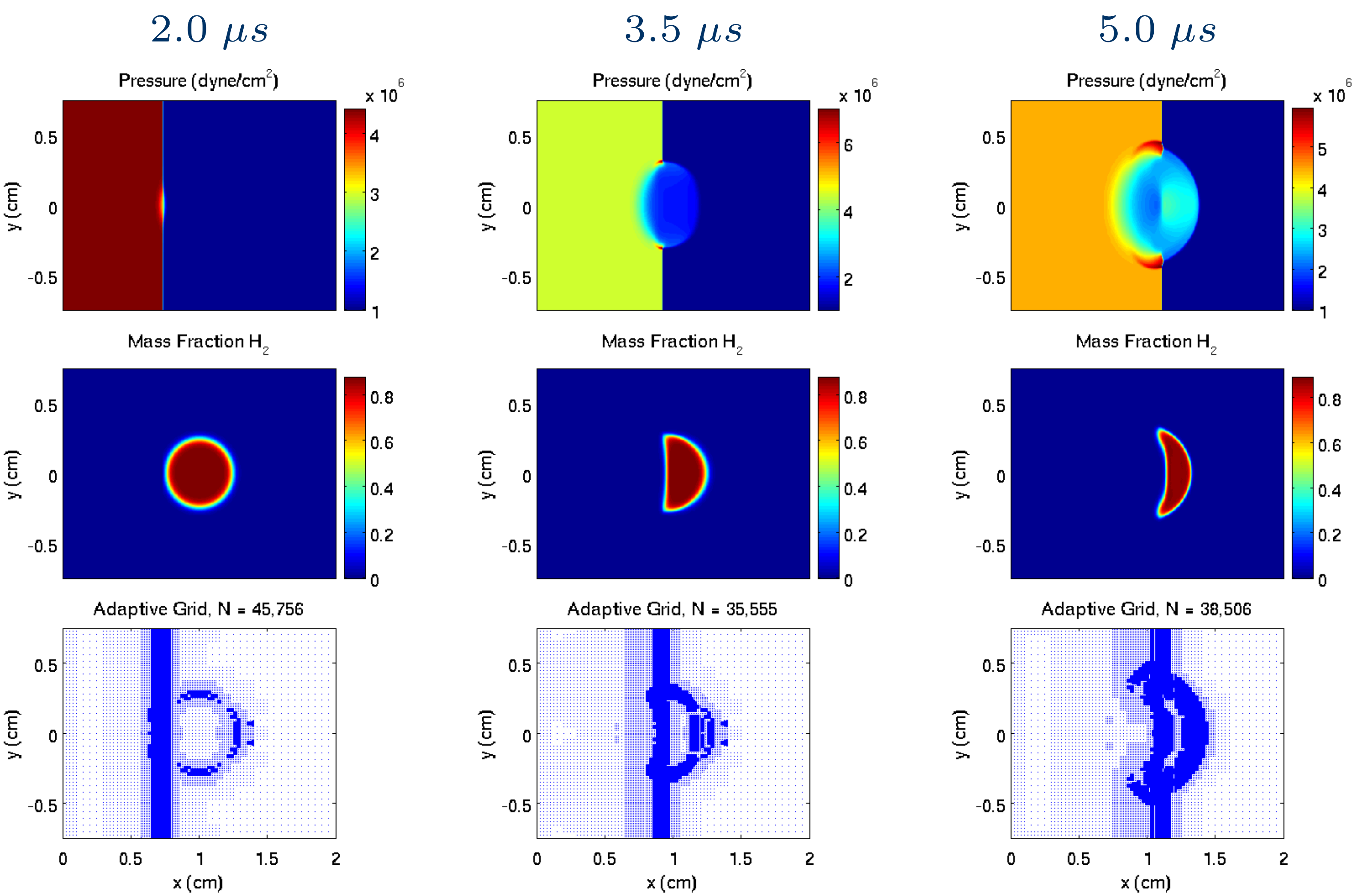
Domain: $[0,3] \times [0,0.75]$ cm
Mach 2 shock: $x = 0.5$ cm
$P_\infty = 1.0 \times 10^6$ dyne/cm^2
$T_\infty = 1000$ K
$r = \sqrt{(x-1)^2 + y^2}$
$r < 0.28$ cm: $83H_2 : 17N_2$
$r > 0.28$ cm: $22O_2 : 78N_2$
64 cores
runtime

$H_2 : O_2 : N_2$ mixture
9 species, 37 reactions

Wavelet parameters:

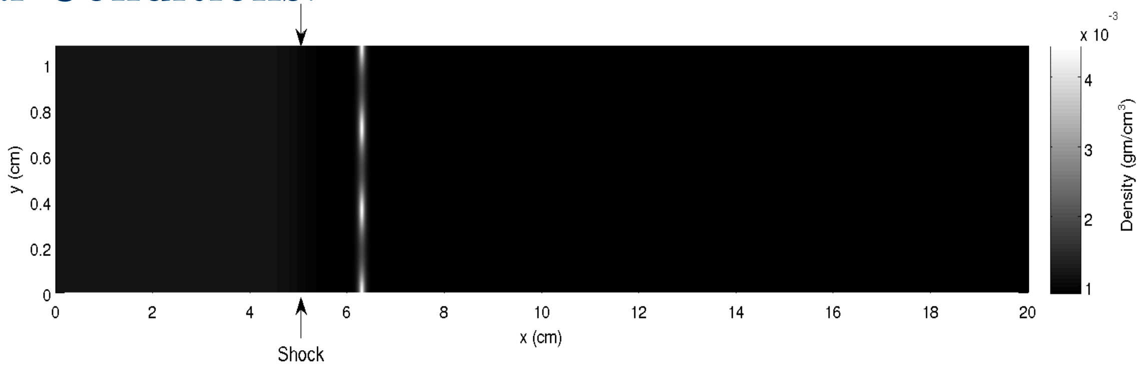
$\epsilon = 1 \times 10^{-3}$
$p = 6, \quad n = 5$
$[N_x \times N_y]_{j_0} = [30 \times 8]$
$J - j_0 = 10$

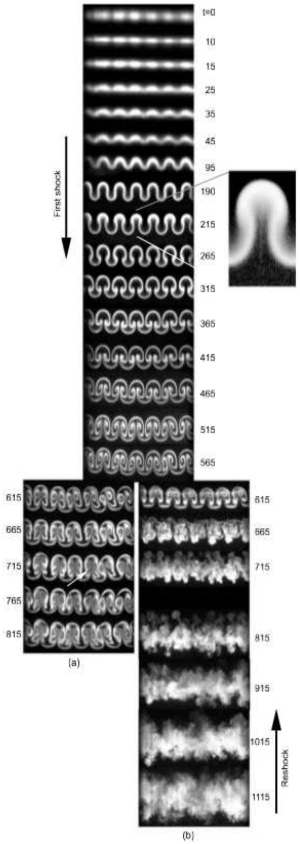
SHOCK/H_2-BUBBLE INTERACTION (CONT.)



Richtmeyer-Meshkov Instability

Initial Conditions:





Domain:

$[0, 20] \times [0, 1.08]$ cm

Ambient mixture:

$Y_{N_2} = 0.99, \; Y_{SF_6} = 0.01$

$P = 79.5$ kPa

$T = 300$ K

$M_s = 1.2$ shock

at $x = 5.0$ cm

64 cores

118 hrs runtime

Varicose sheet at $x = 6.3$ cm

$Y_{N_2} = 0.01, \; Y_{SF_6} = 0.99$

Balakumar *et al.*

Phys. Fluids **20**, 2008

Wavelet parameters:

$\epsilon = 1 \times 10^{-4}$

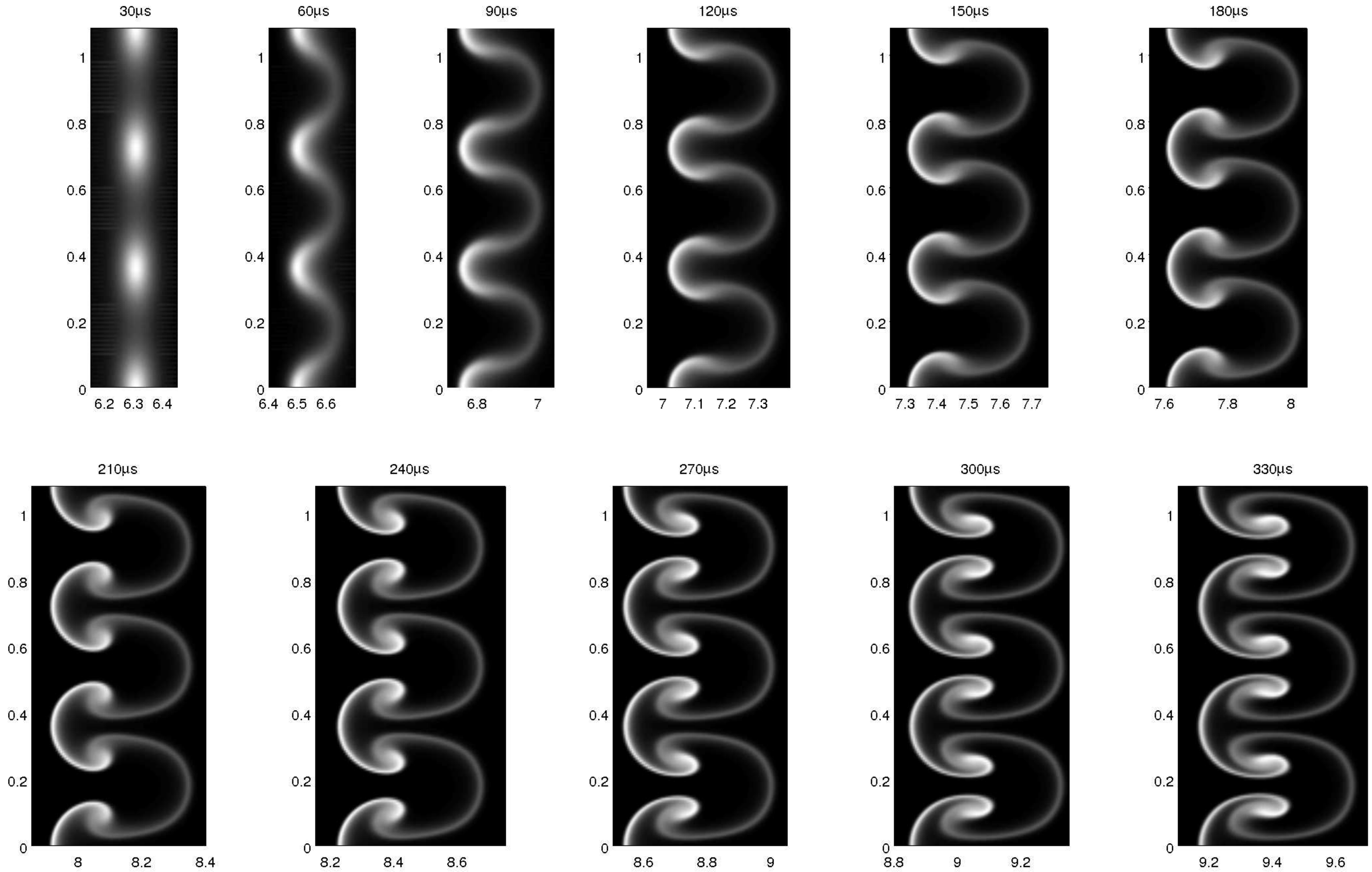
$p = 6, \quad n = 5$

$[N_x \times N_y]_{j_0} = [200 \times 10]$

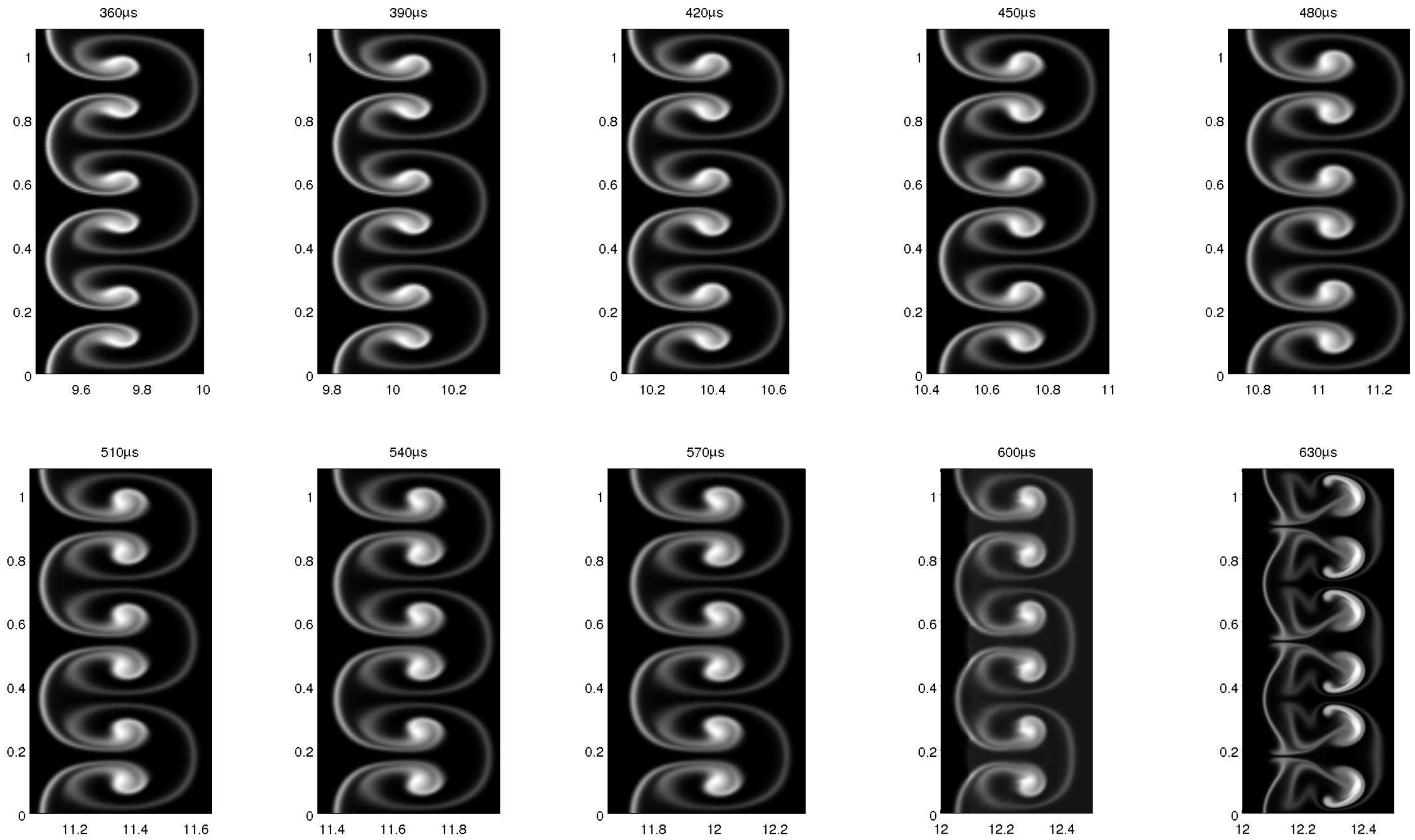
$J - j_0 = 10$

Richtmeyer-Meshkov Instability (cont.)

$\Longrightarrow$ Shock Direction $\Longrightarrow$



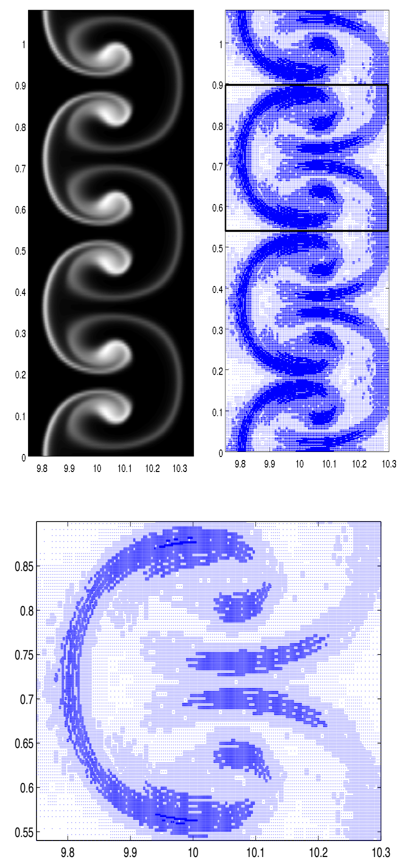
Richtmeyer-Meshkov Instability (cont.)



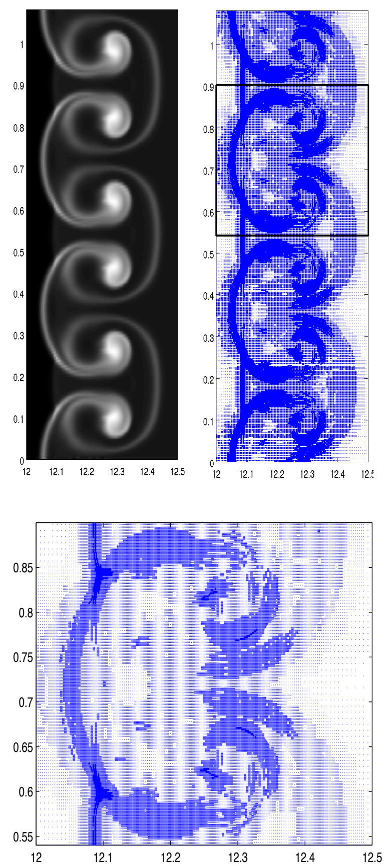
$\Leftarrow$ Reshock $\Leftarrow$

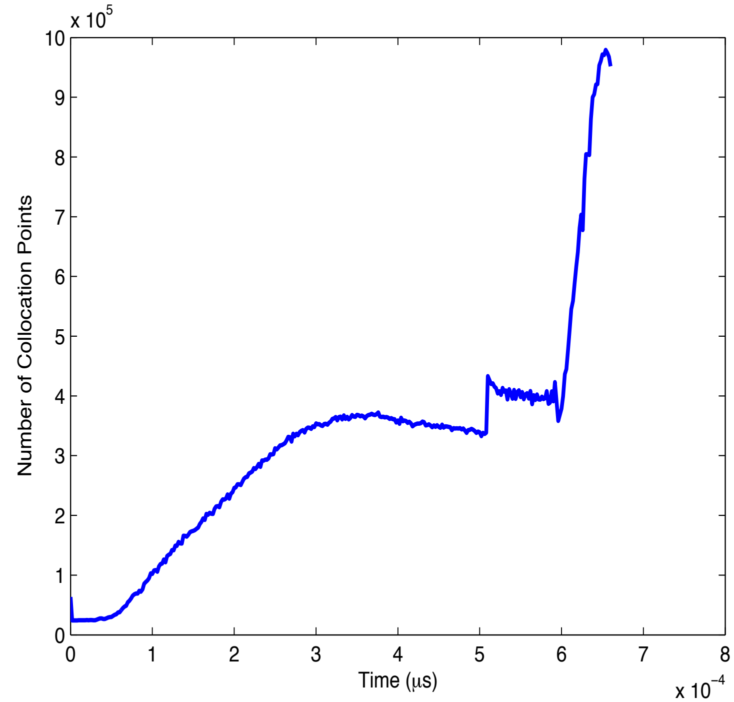
Richtmeyer-Meshkov Instability – Grid

$t = 390\ \mu s$



$t = 600\ \mu s$





Summary

- The wavelet adaptive multiresolution method provides a means to capture a wide range of scales present in multidimensional reactive compressible flows.
- The parallel algorithm shows excellent scaling up to the maximum number of cores tested.
- Resolved (verified) solutions in large geometries require large computational resources even with an adaptive method.

1-D Interpolating Scaling Function and Wavelet

Some properties of $\phi_{j,k}$ and $\psi_{j,k}$ of order p ($p \in \mathbb{N}$, 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.* $\mathrm{supp}\{\phi_{j,k}\} \sim |O(2^{-j})|$.
- $\phi_{j,k}(x_{j,n} = n2^{-j}) = \delta_{k,n}$, *i.e.* satisfies the *interpolation property.*
- $\psi_{j,k} = \phi_{j+1,2k+1}$.
- $\mathrm{span}\{\phi_{j,k}\} = \mathrm{span}\{\{\phi_{j-1,k}\},\{\psi_{j-1,k}\}\}$.
- $\{1, x, \cdots, x^{p-1}\}$, 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 $[0,1]$.