

Adaptive wavelet algorithm for solving nonlinear initial-boundary value problems with error control

C. Harnish^{a,b}, K. Matouš^{a,b,*}, D. Livescu^c

^a*Department of Aerospace and Mechanical Engineering*

^b*Center for Shock Wave-processing of Advanced Reactive Materials,
University of Notre Dame, Notre Dame, IN 46556, USA*

^c*Computer and Computational Sciences Division, Los Alamos National Laboratory,
Los Alamos, NM 87545, USA*

Abstract

We present a numerical method which exploits the biorthogonal interpolating wavelet family, and second generation wavelets, to solve initial-boundary value problems on finite domains. Our predictor-corrector algorithm constructs a dynamically adaptive computational grid with significant data compression, and provides explicit error control. Error estimates are provided for the wavelet representation of functions, their derivatives, and the nonlinear product of functions. The method is verified on traditional nonlinear problems such as Burgers' equation and the Sod shock tube. Numerical analysis shows polynomial convergence with negligible global energy dissipation.

Keywords: multiresolution analysis, wavelets, adaptive algorithm, nonlinear PDEs, data compression

1. Introduction

As the field of computational physics has matured, the engineering applications which we seek to model have grown remarkably in size and complexity. The scope of modern simulations include: the global ocean Ringler et al. (2013), detonation combustion Cai et al. (2016), asteroid impacts Boslough et al. (2015), and supernova remnants Malone et al. (2014). As these problems are inherently interdisciplinary and multiscale, reliable numerical models must adaptively solve partial differential equations (PDEs) with multiphysics features on spatial and temporal scales across many orders of magnitude.

Several numerical methods have been developed to address the computational difficulty of these multiscale problems. For example, adaptive mesh refinement (AMR) constructs an irregular grid by recursively refining the mesh size in different locations Berger and Oliger (1984); Fatkullin and Hesthaven (2001). Similarly, multigrid methods use a hierarchy of grids to find a suitable spatial resolution Brandt (1977); Hackbusch (1978). Further adaptivity is available with finite element methods (FEM) by modifying the mesh size, changing the degree of the basis functions, relocating nodes, or any combination of such approaches Dong and Karniadakis (2003); Gui and

*Corresponding author.

Email addresses: charnish@nd.edu (C. Harnish), kmatus@nd.edu (K. Matouš), livescu@lanl.gov (D. Livescu)

Babuška (1986a,b); Rajagopal and Sivakumar (2007). Each of these techniques have merits and deficiencies. For example, AMR methods readily achieve variable resolution Klein (1999), multigrid methods are extremely efficient linear solvers Thekale et al. (2010), and complex geometries are amenable to FEM Schillinger and Rank (2011). However, both AMR and FEM require costly *a posteriori* analysis for adaption criteria Segeth (2010), and computationally efficient implementation of the necessary mesh repair, smoothing, or re-meshing is challenging Demkowicz et al. (1989). Furthermore, multigrid methods may require a major programming effort for each new grid configuration Dendy (1982).

Wavelet methodologies offer an alternative approach for numerically solving multiscale PDEs Schneider and Vasilyev (2010). These algorithms achieve spatial adaptivity with multiresolution wavelet basis functions Jawerth and Sweldens (1994). Notable accomplishments of wavelet solvers include: significant data compression Bertoluzza (1996); Beylkin and Keiser (1997); Liandrat and Tchamitchian (1990), bounded energy conservation Qian and Weiss (1993); Ueno et al. (2003), modeling stochastic systems Kong et al. (2016), and solving coupled systems of nonlinear PDEs Dubos and Kevlahan (2013); Nejadmalayeri et al. (2015); Paolucci et al. (2014a,b); Sakurai et al. (2017). While past solvers have had many successes, they are not without shortcomings. Many wavelet approaches only solve PDEs which are defined on infinite or periodic domains (e.g. Fröhlich and Schneider (1994); Goedecker (1998); Iqbal and Jeoti (2014)). Additionally, some algorithms do not exploit the data compression ability of wavelets, resulting in a computationally expensive uniform grid (e.g. Le and Caracoglia (2015); Lin and Zhou (2001); Qian and Weiss (1993)). Lastly, several wavelet methods use finite difference techniques to compute the spatial derivatives, requiring the PDEs to be solved in the physical domain rather than in the wavelet domain (e.g. Holmström (1999); Nejadmalayeri et al. (2015); Paolucci et al. (2014a,b)).

Our work advances the state of wavelet based methods with the development of a predictor-corrector algorithm which is designed to overcome the limitations of past solvers, while retaining their advantages. We solve nonlinear initial-boundary value problems on finite domains using differentiable wavelet bases and second generation wavelets near spatial boundaries. We maximize the data compression ability of these bases by populating the coarsest resolution with the minimum number of collocation points required for support of the wavelet basis function. Therefore, we define our bases with a modified support interval and derive special scaling relations to account for the variable grid spacing. Moreover, we compute spatial derivatives by operating directly on the wavelet bases. We derive error estimates for field values, their derivatives, and the aliasing errors associated with the nonlinear terms in a PDE. Then, our estimates are used to construct a sparse, dynamically adaptive, computational grid for each unknown that *a priori* guarantees the required accuracy. Our predictor-corrector procedure maintains the prescribed accuracy through time, and allows each field to adapt independently using its own wavelet grid. Our algorithm provides data compression on par with state-of-the-art wavelet solvers, has negligible global energy growth, and solves coupled systems of nonlinear PDEs in the wavelet domain.

Before presenting the mathematical and numerical concepts, a summary of wavelet discretization, differentiation, and corresponding *a priori* error estimation, is presented in Section 2. Then, the procedure of our algorithm is described in Section 3. Lastly, verification is provided in Section 4 with numerical solutions of nonlinear problems such as Burgers' equation and the Sod shock tube.

2. Wavelet Representation

For completeness of the presentation, we provide a brief review of wavelet theory. In particular, we summarize the formation of wavelet basis functions and explicitly define the mathematical operations needed to solve nonlinear PDEs with these bases. Additionally, we identify the known estimates for the spatial error associated with each of these operations and provide a new derivation of the error accumulated during wavelet based differentiation.

A multiresolution analysis (MRA) provides the formal mathematical definition of a wavelet family of basis functions Daubechies (1992). A MRA of a domain $\Omega \subset \mathbb{R}$ consists of a sequence of successive approximation spaces V_j and their associated dual spaces \widetilde{V}_j such that the union of these spaces is the $L^2(\Omega)$ space Cohen et al. (2000b)

$$V_j \subset V_{j+1}, \quad \widetilde{V}_j \subset \widetilde{V}_{j+1}, \quad \overline{\bigcup_{j=0}^{\infty} V_j} = L^2(\Omega). \quad (1)$$

The wavelet spaces W_j (\widetilde{W}_j) are then defined as the complements of the approximation spaces V_j (\widetilde{V}_j) in V_{j+1} (\widetilde{V}_{j+1}) Bacry et al. (1992); Qian and Weiss (1993)

$$V_{j+1} = V_j \oplus W_j, \quad \widetilde{V}_{j+1} = \widetilde{V}_j \oplus \widetilde{W}_j. \quad (2)$$

This multiresolution property requires the use of two indices: j the resolution level, and k the unique spatial locations on level j . The scaling functions $\phi_k^j(x)$ and dual scaling functions $\widetilde{\phi}_k^j(x)$ are the basis functions of the spaces V_j and \widetilde{V}_j respectively, whereas the wavelets $\psi_k^j(x)$ and dual wavelets $\widetilde{\psi}_k^j(x)$ are the basis functions of the spaces W_j and \widetilde{W}_j respectively. These bases are completely defined by the filter coefficients $h_i, \widetilde{h}_i, g_i$, and \widetilde{g}_i , derived in Goedecker (1998); Villiers et al. (2003).

Our proposed numerical method utilizes the biorthogonal interpolating wavelet family of basis functions, defined in Donoho (1992). These bases are sometimes referred to as the Deslauriers-Dubuc wavelets Burgos et al. (2013); Fujii and Hoefer (2003), or the auto-correlation of the Daubechies wavelets Bertoluzza and Naldi (1996). Since modified bases are required for wavelet representation on an interval Alpert et al. (2002); Sweldens (1998), we use the second generation wavelets defined in Villiers et al. (2003) near spatial boundaries. The remainder of this section summarizes operations specialized for this particular wavelet family.

2.1. Wavelet Discretization

We discretize in space by projecting a continuous function $f(x)$, defined on a finite interval $\{x \in \Omega \mid a \leq x \leq b\}$, onto the basis functions $\phi_k^0(x)$ and $\psi_k^j(x)$. The s_k^0 scaling function coefficients are equal to the field values calculated from

$$s_k^0 = f(a + k\Delta x), \text{ where } \Delta x = \frac{b-a}{2p}, \{k \in \mathbb{Z} : 0 \leq k \leq 2p\}. \quad (3)$$

The parameter p is an even integer which defines the properties of the basis functions (e.g. number of vanishing moments and interpolation order). Equation (3) departs from traditional wavelet methods by defining the coarsest grid spacing Δx with the minimum number of collocation points (i.e. $2p + 1$) required to satisfy the support of the wavelet basis function. This modifies the support

interval of all basis functions, maximizes data compression, and is unique to our algorithm as traditional wavelet methods usually define $\Delta x = 1$.

Next, the d_k^j wavelet coefficients are equal to the local interpolation error calculated from

$$d_k^j = \sum_{i=0}^{2p} \tilde{g}_i f_i, \text{ where } \{j, k \in \mathbb{Z} : 1 \leq j \leq \infty \wedge 1 \leq k \leq 2^j p\}, \quad (4)$$

and f_i is defined by,

$$f_i = \begin{cases} f\left(a + i \frac{\Delta x}{2^j}\right) & k \leq p/2 \\ f\left(b + (i - 2p) \frac{\Delta x}{2^j}\right) & k > 2^j p - p/2 \\ f\left(a + (i + 2k - p - 1) \frac{\Delta x}{2^j}\right) & \text{otherwise.} \end{cases} \quad (5)$$

It has been shown by many authors (e.g. Holmström (1999); Nejadmalayeri et al. (2015); Paolucci et al. (2014b)) that retaining only those d_k^j coefficients with a magnitude greater than or equal to some prescribed threshold ε , results in the discretization $f_\varepsilon(x)$ which approximates $f(x)$ with the spatial error

$$\|f(x) - f_\varepsilon(x)\|_\infty \leq O(\varepsilon). \quad (6)$$

Therefore, we calculate all of the d_k^j coefficients on resolution level $j = 1$ and refine locally around those $|d_k^j| \geq \varepsilon$ until we reach $j = j_{\max}$, where any further refinement would not produce any significant coefficients. In this way, we create the sparse representation

$$f_\varepsilon(x) = \sum_{k=0}^{2p} s_k^0 \phi_k^0(x) + \sum_{j=1}^{j_{\max}} \sum_{\{k: |d_k^j| \geq \varepsilon\}} d_k^j \psi_k^j(x). \quad (7)$$

For any continuous function, this discretization procedure only requires defining two parameters. The parameter p determines the properties of the bases and the parameter ε determines the accuracy of the discretization. For example, suppose we wish to discretize the function

$$f(x) = \arctan(100x - 50), \quad \text{on} \quad x \in [0, 1]. \quad (8)$$

If we choose $p = 8$ and $\varepsilon = 10^{-3}$ then Eqs. (3) to (7) completely define the projection onto wavelet basis functions. The resulting sparse multiresolution discretization is shown in Fig. 1. We will show in Section 2.3 the spatial error for field values (i.e. Eq. (6)) together with derivatives.

2.2. Backward and Forward Wavelet Transforms

With interpolating wavelets, a backward wavelet transform (BWT) maps the d_k^j wavelet coefficients back to their corresponding field values. A forward wavelet transform (FWT) does the inverse and returns these field values to their corresponding d_k^j wavelet coefficients. These operations are often referred to as wavelet synthesis and analysis respectively Farge (1992). The BWT is performed at each resolution level, from lowest to highest, by the matrix operator \mathbf{B} . Likewise, the FWT is performed at each resolution level, from highest to lowest, by the matrix operator \mathbf{F} .

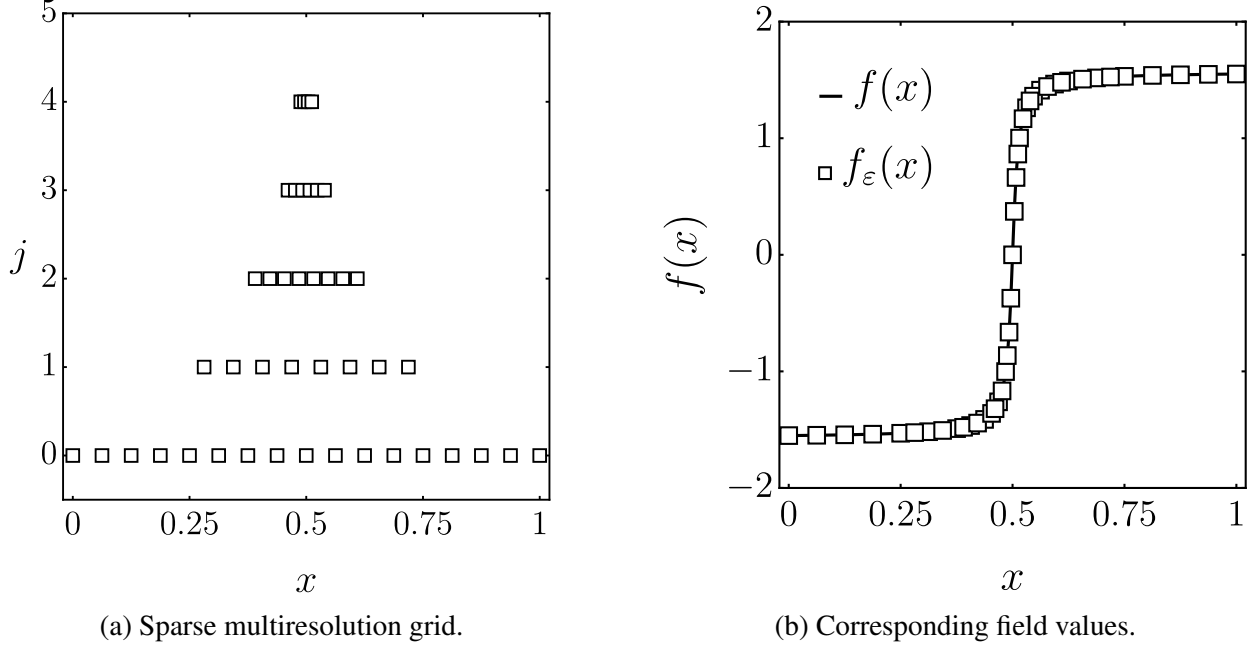


Figure 1: Wavelet spatial discretization of Eq. (8) with $p = 8$ and $\varepsilon = 10^{-3}$.

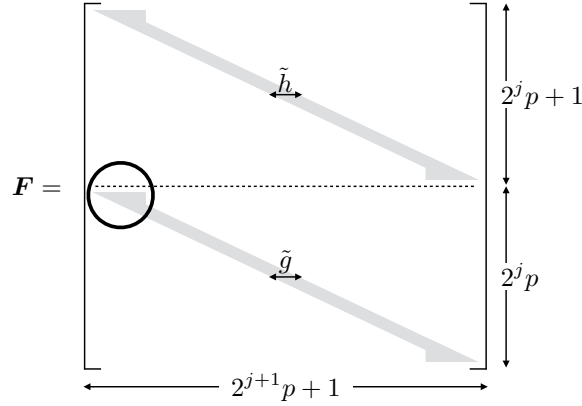


Figure 2: Definition of matrix \mathbf{F} , with modifications near spatial boundaries. An example of this modification is highlighted by the circle. The matrix \mathbf{B} is defined by the inverse of \mathbf{F} . Note that the matrix-free implementation of the FWT and BWT can be developed.

These matrices are sparse, banded, and constant in time. Due to these properties, the \mathbf{B} and \mathbf{F} matrices are never fully assembled, and only non-redundant non-zero entries are stored in memory. Therefore, the FWT and BWT operations have a matrix-free computational implementation.

The structure of these matrices are similar to those used in Goedecker (1998); Jameson (1993). Though like Dahmen et al. (1999), we modify these matrices with information of spatial boundaries (e.g. circled region of the matrix in Fig. 2). The matrix notation replaces the cumbersome indices and summations of Eq. (7) with:

$$f_\varepsilon(x) = \vec{f} \cdot \vec{\Phi} \quad \text{where} \quad \vec{f} = \mathbf{B} \cdot \vec{\mathfrak{d}}, \quad (9)$$

$$f_\varepsilon(x) = \vec{\mathfrak{d}} \cdot \vec{\Psi} \quad \text{where} \quad \vec{\mathfrak{d}} = \mathbf{F} \cdot \vec{f}. \quad (10)$$

2.3. Wavelet Derivatives

The smoothness of our wavelet family has been studied in Rioul (1992) and is summarized in Table 1. This continuity allows the spatial derivative operator to act directly on the basis functions,

$$\frac{d^m}{dx^m} f(x) \approx \frac{d^m}{dx^m} (\vec{d} \cdot \vec{\Psi}) = \vec{d} \cdot \frac{d^m \vec{\Psi}}{dx^m}. \quad (11)$$

p	Hölder Regularity	Continuity
2	\dot{C}^1	C^0
4	\dot{C}^2	C^1
6	$\dot{C}^{2.83}$	C^2
8	$\dot{C}^{3.55}$	C^3
10	$\dot{C}^{4.1935}$	C^4

Table 1: Regularity estimates from Rioul (1992) for the biorthogonal interpolating wavelet family.

As in Beylkin and Keiser (1997), we project the spatial derivative of $\vec{\Psi}$ onto the same wavelet basis functions and Eq. (11) becomes

$$\frac{d^m}{dx^m} f(x) \approx (\mathbf{D}^{(m)} \cdot \vec{d}) \cdot \vec{\Psi}, \quad (12)$$

where the matrix $\mathbf{D}^{(m)}$ is defined in Appendix A, and depicted in Fig. 3. Again, this matrix is sparse, banded, and constant in time. Therefore, the derivative operations also have a matrix-free computational implementation. This results in a discrete approximation $\mathbf{D}^{(m)} f_\varepsilon(x)$ of the m^{th} order derivative $f^{(m)}(x)$ with the spatial error

$$\left\| f^{(m)}(x) - \mathbf{D}^{(m)} f_\varepsilon(x) \right\|_\infty \leq O\left(\varepsilon^{1-\frac{m}{p}}\right). \quad (13)$$

Similar error estimates are found in Dubos and Kevlahan (2013); McCormick and Wells (1994). In this work, we rigorously derive the error bound on wavelet derivatives of any order, defined on a finite domain. Our derivation of Eq. (13) is a new contribution specific to the Deslauriers-Dubuc wavelet family, and is located in Appendix B.2.

Now we show the merits of the wavelet discretization and differentiation in more detail. Suppose we have a continuous field, such as in Eq. (8), and we need to calculate its first and second derivatives. Since we must choose bases that are at least twice differentiable, we choose $p = 6$ and $p = 8$ (using information from Table 1). Then, by choosing a small arbitrary value for ε , the process in Section 2.1 provides a sparse multiresolution discretization of the field. This wavelet representation guarantees *a priori* the spatial accuracy of $O(\varepsilon)$, as defined in Eq. (6), and shown in Fig. 4. Next, the spatial derivatives of the field are calculated on this sparse grid through the matrix-free implementation of $\mathbf{D}^{(1)}$, $\mathbf{D}^{(2)}$, and Eq. (12). Furthermore, we know *a priori* that such approximations of the first and second derivatives will have the spatial accuracy of $O\left(\varepsilon^{1-\frac{1}{p}}\right)$ and $O\left(\varepsilon^{1-\frac{2}{p}}\right)$ respectively, as defined by Eq. (13), and also shown in Fig. 4.

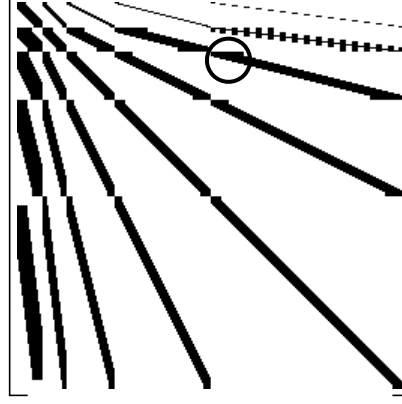


Figure 3: Structure of the wavelet derivative matrix operator $D^{(m)}$ with modifications near spatial boundaries. An example of this modification is highlighted by the circle. Due to its defined structure, the matrix-free implementation can be developed.

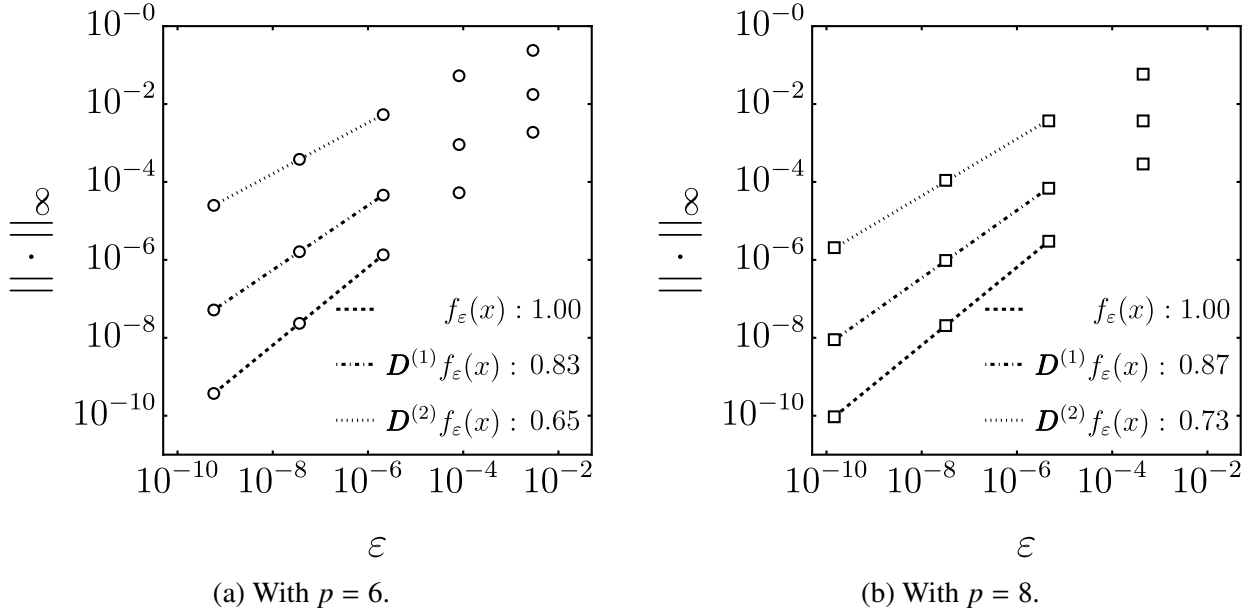


Figure 4: Spatial error for a field, Eq. (8), and its m^{th} derivatives is shown to be $O(\varepsilon)$ and $O(\varepsilon^{1-\frac{m}{p}})$ respectively. (a) Convergence rates with $p = 6$. (b) Convergence rates with $p = 8$.

2.4. Nonlinear Terms

Calculating the product of fields in wavelet space is computationally expensive because it requires a convolution operation. Therefore, we utilize the more efficient pseudo-spectral approach of point-wise multiplication in the physical domain. Specifically, we use Eq. (9) to perform a BWT and map the d_k^j wavelet coefficients to their corresponding field values. Then, we approximate the product of fields by multiplying the field values at each collocation point.

It is well known that this technique introduces aliasing errors. An estimate of the magnitude of such errors is provided in Holmström (1999), where it is shown that this process approximates the product of fields, $f_1(x)$ and $f_2(x)$, with the spatial error

$$\|f_1(x) \times f_2(x) - f_{1\varepsilon}(x) \times f_{2\varepsilon}(x)\|_\infty \leq O(\varepsilon). \quad (14)$$

Since these aliasing errors are bounded by ε , their influence remains of the same order as all other error sources.

3. Computational Implementation

We use the wavelet operations defined in Section 2 to solve nonlinear PDEs. Although these operations are well defined for multiple spatial dimensions Daubechies (1992), we present examples with one spatial dimension to provide better insight in the underlying steps of the algorithm. More detailed three-dimensional studies are needed to fully assess the general convergence estimates and algorithmic improvements. For example, consider the initial-boundary value problem:

$$\begin{aligned} \frac{\partial u}{\partial t} + (u + c) \frac{\partial u}{\partial x} &= \nu \frac{\partial^2 u}{\partial x^2} && \text{in } \Omega \times [0, t_f], \\ u &= u_d && \text{on } \partial\Omega_d \times [0, t_f], \\ \frac{\partial u}{\partial x} &= u_n && \text{on } \partial\Omega_n \times [0, t_f], \\ u &= u_0 && \text{in } \Omega \times (t = 0). \end{aligned} \quad (15)$$

Due to the presence of a second derivative, we specify p such that the bases are at least twice differentiable. This defines the matrix operators $\mathbf{D}^{(m)}$, \mathbf{B} and \mathbf{F} . Next, as is traditional for wavelet-based solvers, we use spatial discretizations from Eqs. (10) and (12) to transform the nonlinear PDE into a nonlinear ODE,

$$\frac{d}{dt} \vec{\mathfrak{d}} + (\vec{\mathfrak{d}} + c) \mathbf{D}^{(1)} \cdot \vec{\mathfrak{d}} = \nu \mathbf{D}^{(2)} \cdot \vec{\mathfrak{d}}. \quad (16)$$

We use the process defined in Section 2.1, to discretize the initial condition u_0 . Equation (6) provides *a priori* knowledge that the spatial accuracy associated with this approximation is explicitly controlled by the threshold parameter ε . Furthermore, as shown in Section 2.1, achieving the $O(\varepsilon)$ spatial error only requires the retention of those entries in $\vec{\mathfrak{d}}$ with $|\mathfrak{d}_k| \geq \varepsilon$. Associating these coefficients with their corresponding collocation points results in the multiresolution computational grid shown in Fig. 5(a).

Next, we use the process defined in Section 2.3 to approximate the spatial derivatives of the initial condition. Equation (13) relates the threshold parameter ε with the spatial accuracy of m^{th} order derivative approximations. As shown in Appendix B.2, achieving the $O(\varepsilon^{1-\frac{m}{p}})$ spatial error requires the retention of some entries in $\vec{\mathfrak{d}}$ with $0 < |\mathfrak{d}_k| < \varepsilon$. Therefore, comparable to the adjacent zone defined in Vasilyev and Paolucci (1996), we include additional points in the computational domain. However, this step is one of the novelties of our method, as we define the neighboring region such that Eq. (13) is satisfied.

For example, each point in Fig. 5(a) corresponds to an entry in $\vec{\mathfrak{d}}$, and a particular row in the matrix $\mathbf{D}^{(m)}$ (see Fig. 3). We examine the non-zero entries within a row to define a multiresolution wavelet stencil for each point. Then, this stencil is used to identify those points which influence the derivative calculations, but which were not retained by the initial discretization of the field with $|\mathfrak{d}_k| \geq \varepsilon$. Such points, shown as filled diamonds in Fig. 5(b), are included in the computational grid if their wavelet coefficients are $0 < |\mathfrak{d}_k| < \varepsilon$. This procedure ensures the validity of Eq. (13), and defines the sparse computational grid.

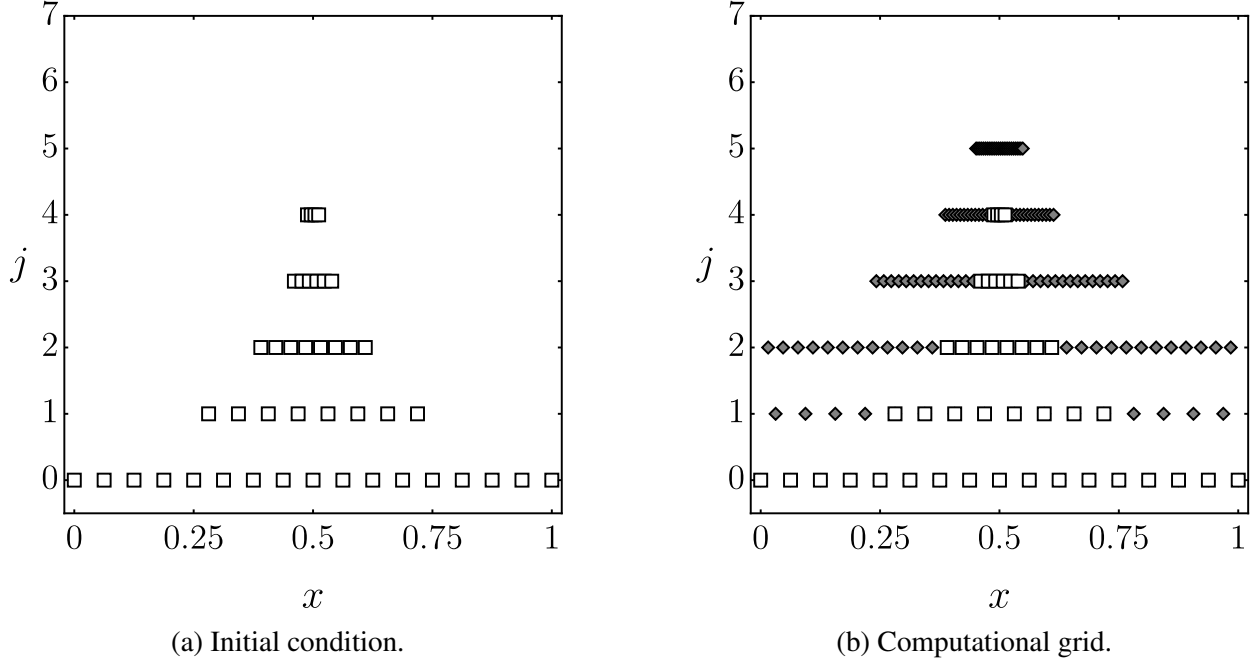


Figure 5: Wavelet spatial discretization of the initial condition with $p = 8$ and $\varepsilon = 10^{-3}$. (a) Collocation points with $|\phi_k| \geq \varepsilon$ (squares) are defined by the initial condition. (b) Computational grid contains the additional collocation points with $0 < |\phi_k| < \varepsilon$ (filled diamonds) that are needed for accurate derivative calculations.

Now the computational grid contains all of the collocation points that are required to approximate the solution of the PDE at time step n , with *a priori* knowledge of the spatial accuracy from Eqs. (6), (13) and (14). Since the solution of the PDE may evolve and advect, it is not clear if these collocation points will be sufficient at time step $n+1$. To resolve this issue, our algorithm combines ideas from Cohen et al. (2000a); Liandrat and Tchamitchian (1990) to define a predictor-corrector procedure. First we add trial points, shown as filled circles in Fig. 6, before advancing to a trial time step $n+1^*$. We utilize the procedure in Liandrat and Tchamitchian (1990) to define a trial grid by expanding the current computational grid by one resolution level and one point in each direction, as shown in Fig. 6(a). The trial grid serves as a prediction of the collocation points which will be required at the next time step.

We then use an explicit time integration scheme to advance the solution, on the trial grid, from the time step n to the trial time step $n+1^*$. The time step size Δt is determined from the traditional linear stability criteria, and adapts according to the highest resolution level present in the grid. This transforms the nonlinear ODE, in Eq. (16), into a system of algebraic equations which update \vec{d} to the trial time step $n+1^*$. Equation (12) is used to calculate spatial derivatives in the wavelet domain at each point in the trial grid. When nonlinear terms are present, the procedure in Section 2.4 is used to calculate the point-wise product on a collocation grid defined by the union of the fields involved. Then, a FWT returns the products to the wavelet domain and the update equations are evaluated.

At this point, we depart from traditional wavelet algorithms by verifying that our prediction of the grid modification was accurate. In one time step, it is possible that structures within the grid have advected more than one point or refined more than one level. Therefore, we check the magnitude of the coefficients on the highest resolution level at the trial time step. If $\|\vec{d}^{n+1^*}\|_\infty \geq \varepsilon$,

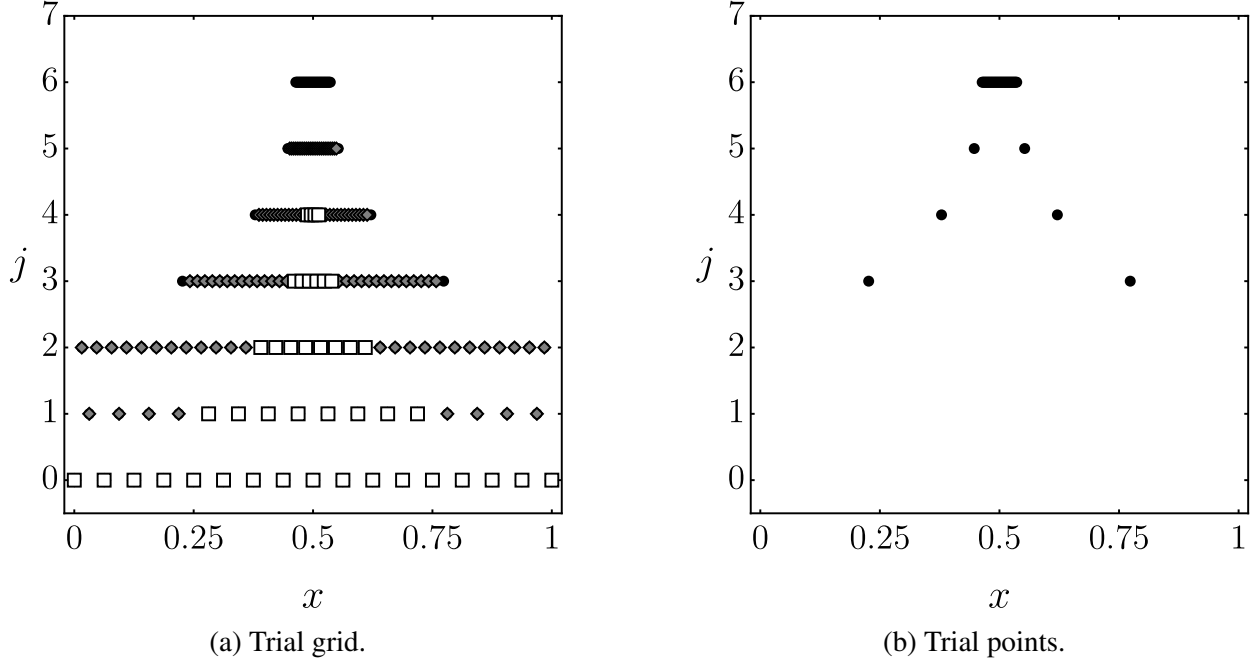


Figure 6: Collocation points associated with the predictor stage of the algorithm. (a) Trial grid contains additional trial points. (b) Location of the trial points.

then we cannot guarantee the accuracy of the solution of the PDE according to the estimates in Eqs. (6), (13) and (14). Consequently, we correct our prediction of the trial grid by discarding the trial time step and, similar to the growing procedure in Cohen et al. (2000a), we expand our prediction of the trial grid by one more resolution level and one more point in each direction. We repeat this process of correcting the trial grid and re-calculating the trial time step until the accuracy of the solution of the PDE can be guaranteed (i.e. $\|\vec{\mathfrak{d}}^{n+1*}\|_\infty < \varepsilon$ on the highest resolution level). This predictor-corrector procedure populates the sparse computational grid as it evolves with the solution of the PDE, and ensures that the spatial error remains bounded by Eqs. (6), (13) and (14) through time.

When a trial time step is accepted, we set $\vec{\mathfrak{d}}^{n+1} = \vec{\mathfrak{d}}^{n+1*}$, and many wavelet coefficients are no longer needed. Collocation points at the new time are retained only if $\|\vec{\mathfrak{d}}^{n+1}\|_\infty \geq \varepsilon$, or if they are used for calculating the spatial derivatives at such points. This procedure prunes the sparse computational grid as it evolves with the solution of the PDE.

Now that we have evolved the solution of the PDE to a new time, we enforce boundary conditions. Dirichlet conditions are handled by setting all collocation points on $\partial\Omega_d$ to the Dirichlet value u_d . Neumann conditions are handled by modifying Eq. (16) for all collocation points on $\partial\Omega_n$ to reflect the known derivatives u_n .

Figure 7 and Algorithm 1 summarize our predictor-corrector algorithm to solve nonlinear PDEs on a sparse, dynamically adaptive, computational grid with explicit error control.

Algorithm 1 Explicit solver for initial-boundary value problems

```
1: procedure INITIALIZE
2: procedure CREATE THE INITIAL SPARSE GRID
3:   Discretize the initial condition with wavelet basis           ▶ Eqs. (3) to (7)
4:   Include non-zero coefficients for derivative calculations     ▶ Eq. (12)
5: while  $t < t_{\text{final}}$  do
6:   procedure CHOOSE  $\Delta t$ 
7:   repeat
8:     procedure ADD TRIAL POINTS
9:     procedure CALCULATE DERIVATIVES                           ▶ Eq. (12)
10:    procedure CALCULATE PRODUCTS                               ▶ Eqs. (9) and (10)
11:    procedure PERFORM A TRIAL TIME STEP
12:  until  $\|\vec{c}^{n+1*}\|_{\infty} < \varepsilon$  on highest resolution level
13:  procedure ACCEPT TIME STEP
14:    Retain coefficients based on  $\|\vec{c}^{n+1}\|_{\infty} \geq \varepsilon$  and derivative calculations
15:    Enforce B.C.
```

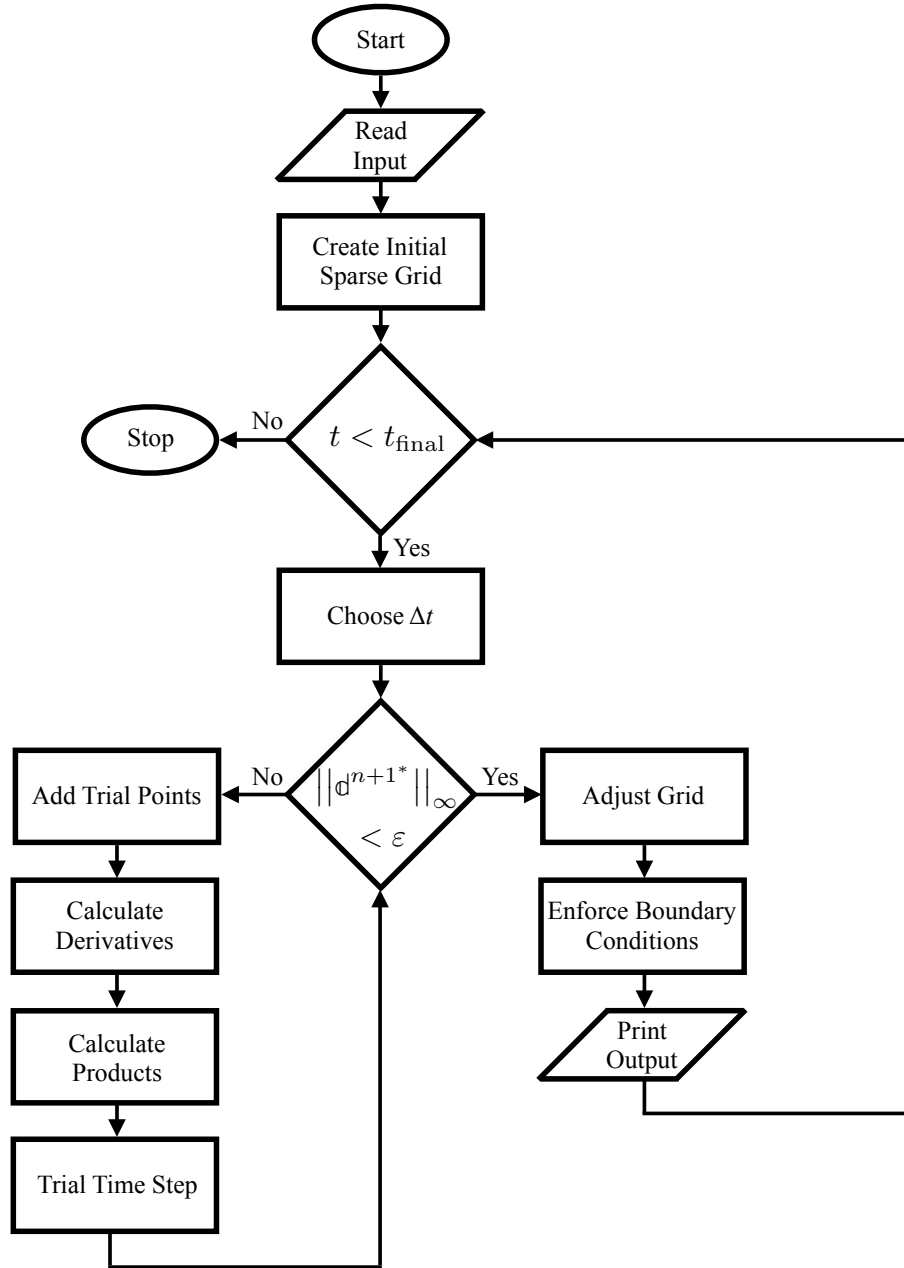


Figure 7: Dynamically adaptive wavelet solver for nonlinear PDEs.

4. Numerical Examples

This section provides verification examples of the adaptive algorithm described in Section 3. Burgers' equation is solved in two separate cases to subject the algorithm to shock wave evolution and shock wave advection. Then, the Sod shock tube problem is solved to subject the algorithm to a coupled system of nonlinear equations.

4.1. Burgers' equation

The general form of Burgers' equation was given in Eq. (15) and this section uses the following dimensionless values: $\nu = 10^{-2}$, $t_f = 1/2$, and $\Omega = (-1, 1)$, with no Neumann conditions. A shock evolution problem is defined by setting $c = 0$, with the following initial and Dirichlet conditions:

$$u_0 = -\sin \pi x, \quad (17)$$

$$u(-1, t) = 0, \quad u(1, t) = 0, \quad (18)$$

and has the exact solution,

$$u(x, t) = -\frac{\int_{-\infty}^{\infty} \sin(\pi x - \pi \eta) \exp\left(\frac{-\cos(\pi x - \pi \eta)}{2\pi\nu}\right) \exp\left(\frac{-\eta^2}{4\nu t}\right) d\eta}{\int_{-\infty}^{\infty} \exp\left(\frac{-\cos(\pi x - \pi \eta)}{2\pi\nu}\right) \exp\left(\frac{-\eta^2}{4\nu t}\right) d\eta}. \quad (19)$$

A shock advection problem is defined by setting $c = 2$, with the following initial and Dirichlet conditions:

$$u_0 = -\tanh\left(\frac{x + 1/2}{2\nu}\right), \quad (20)$$

$$u(-1, t) = 1, \quad u(1, t) = -1, \quad (21)$$

and has the exact solution,

$$u(x, t) = -\tanh\left(\frac{x + 1/2 - ct}{2\nu}\right). \quad (22)$$

Second order accurate explicit Runge-Kutta time integration is used with $p = 6$, and $\varepsilon = 10^{-3}$, to obtain the approximate solutions shown in Figs. 8 and 9.

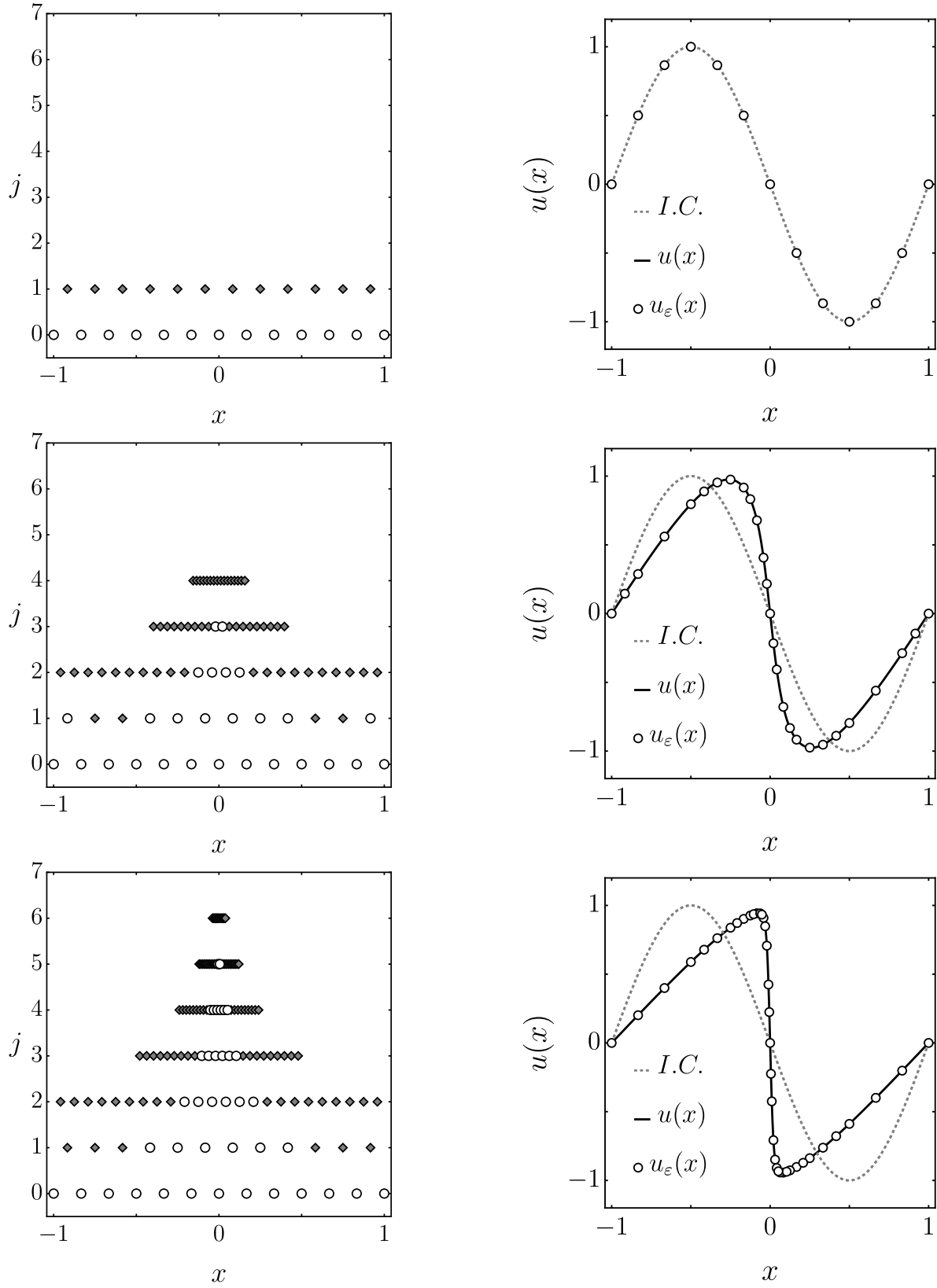


Figure 8: Sparse multiresolution grid and corresponding field values for the evolution of a shock at times $t = 0$ (top), $t = 1/4$ (middle), and $t = 1/2$ (bottom) with $p = 6$ and $\varepsilon = 10^{-3}$.

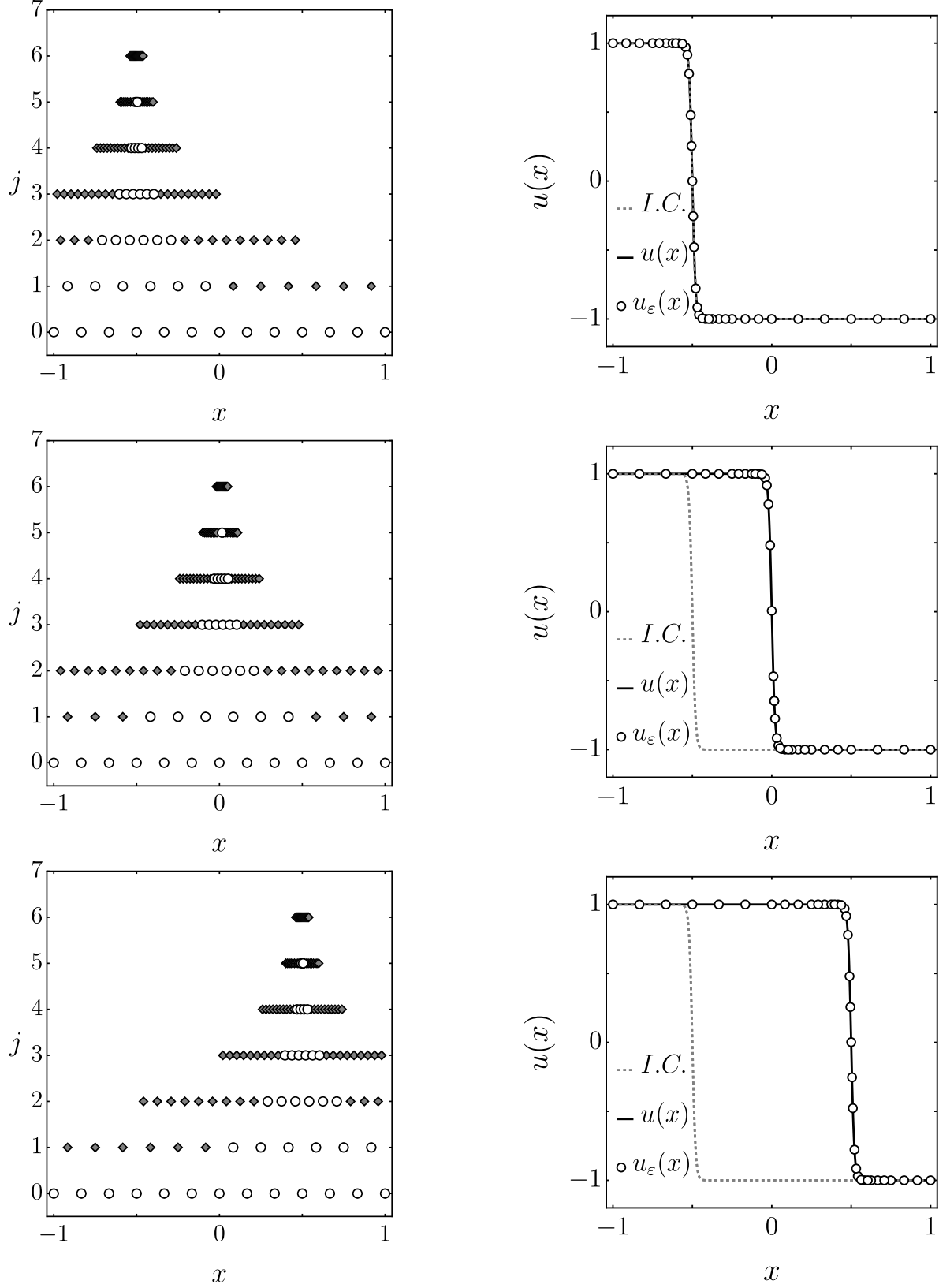


Figure 9: Sparse multiresolution grid and corresponding field values for the advection of a shock at times $t = 0$ (top), $t = 1/4$ (middle), and $t = 1/2$ (bottom) with $p = 6$ and $\varepsilon = 10^{-3}$.

We use the exact solutions in Eqs. (19) and (22) to provide quantitative error analysis. The error at each time step n is shown in Fig. 10 for both the shock evolution and shock advection problems. As predicted by *a priori* error estimates Eqs. (6), (13) and (14), the error at any time step is bounded by $\max\{O(\varepsilon), O(\varepsilon^{1-\frac{m}{p}})\}$.

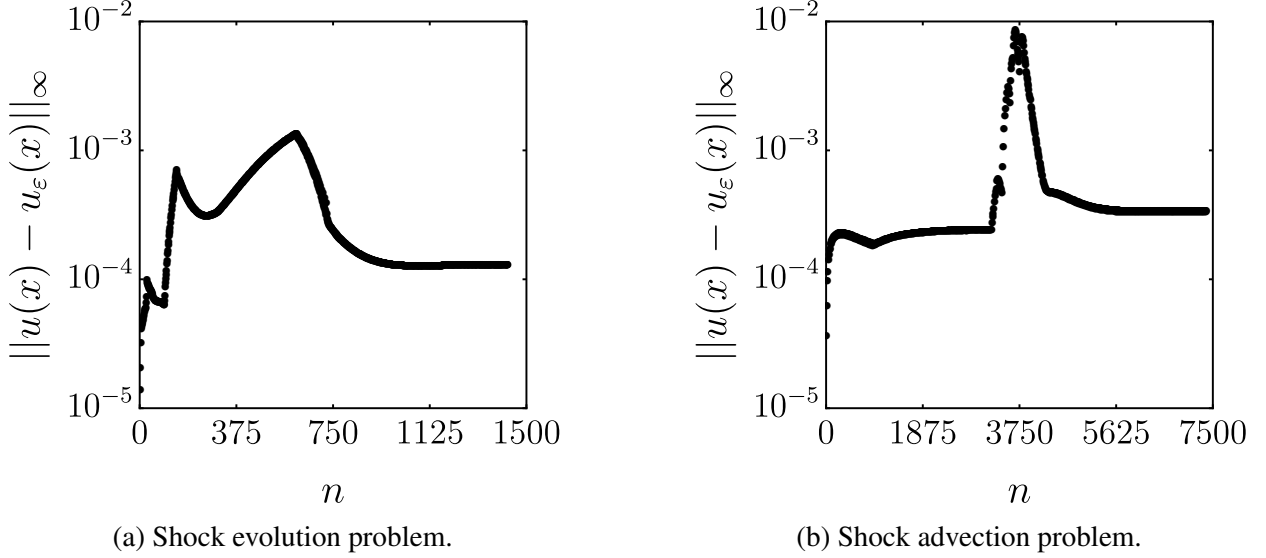


Figure 10: Spatial error at each time step n with $p = 6$ and $\varepsilon = 10^{-3}$. The *a priori* estimate of the error bound is $\max\{O(\varepsilon), O(\varepsilon^{\frac{3}{6}}), O(\varepsilon^{\frac{2}{3}})\} = 10^{-2}$.

Solving each form of Burgers' equation with various values for ε verifies that the spatial convergence rate approaches the theoretical estimates, as shown in Fig. 11.

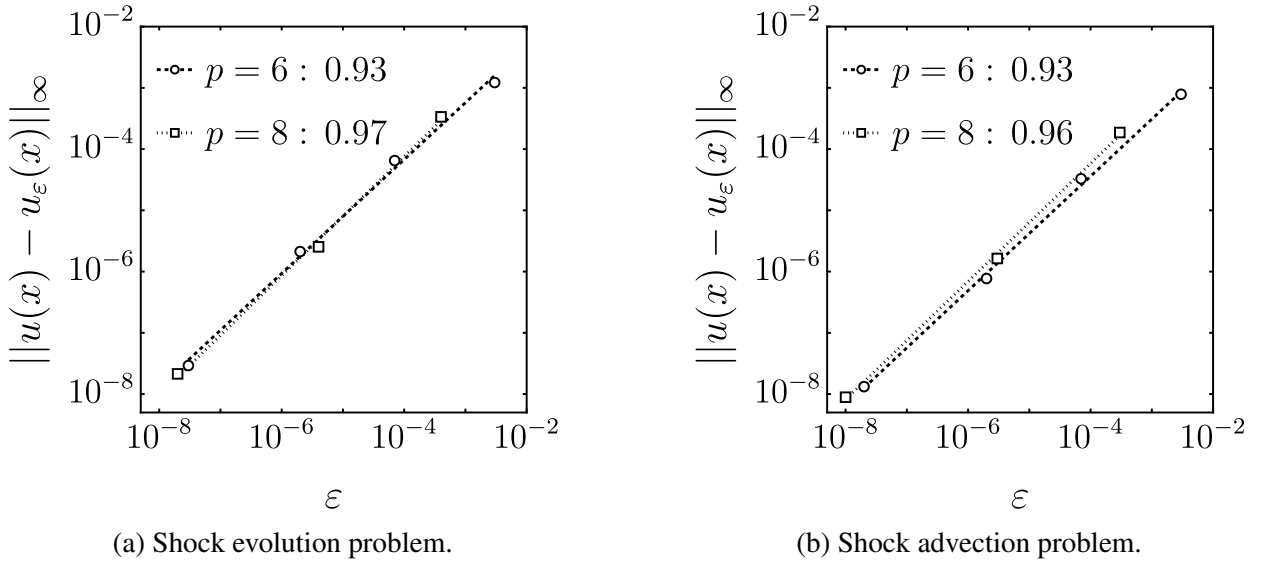


Figure 11: Spatial convergence for Burgers' equation.

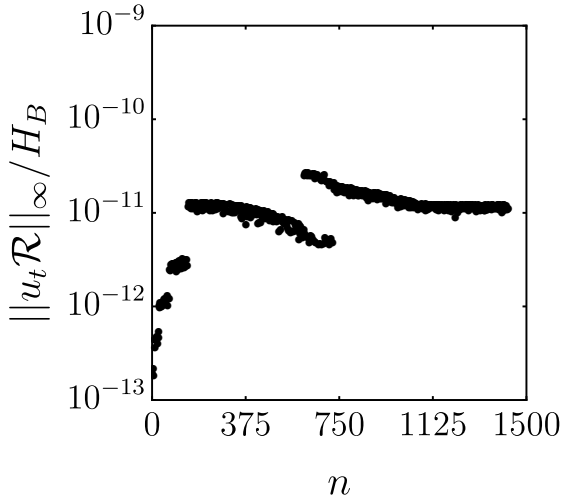
In general, collocation methods are not energy conserving. However, the strict error control of our method results in negligible changes to the global energy at each time step. Specifically, we quantify the global energy growth of our algorithm by showing that the generalized energy integral H_B is approximately time invariant. It has been shown in Ueno et al. (2003) that

$$H_B = \int_{-1}^1 \left\{ \int_0^1 \left[\left(\frac{\partial u}{\partial \xi} \right)^2 + u \frac{\partial u}{\partial \xi} \frac{\partial u}{\partial x} \right] d\xi + \frac{\nu}{2} \left(\frac{\partial u}{\partial x} \right)^2 \right\} dx, \quad (23)$$

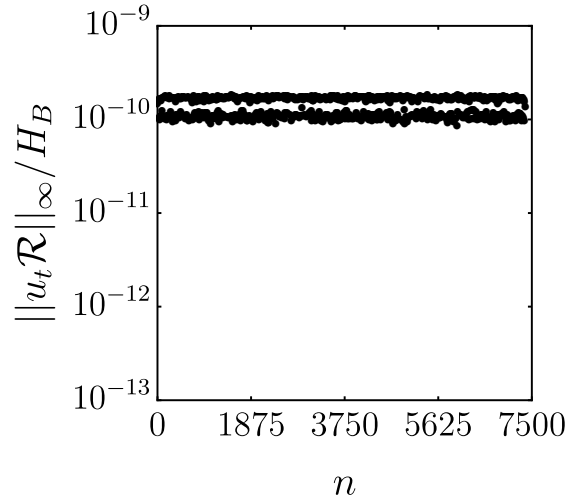
$$\frac{d}{dt} H_B = \int_{-1}^1 \frac{\partial u}{\partial t} \left(\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} - \nu \frac{\partial^2 u}{\partial x^2} \right) dx, \quad (24)$$

$$\frac{d}{dt} H_B = \int_{-1}^1 u_t \mathcal{R} dx. \quad (25)$$

Instead of evaluating this integral directly, Fig. 12 shows the magnitude of the integrand normalized by the value of H_B computed from the exact solution. Since this integrand is approximately zero for every time step, our adaptive wavelet algorithm conserves global energy approximately, but with a high degree of accuracy.



(a) Shock evolution problem.



(b) Shock advection problem.

Figure 12: Magnitude of the integrand in Eq. (25) at each time step n .

4.2. Sod Shock Tube

The Sod problem, as defined in Kamm and et al. (2008), is a type of Riemann problem with a shock wave and a contact discontinuity that move to the right, and a rarefaction wave that moves to the left. The governing equations for this problem are the one dimensional Navier-Stokes equations:

$$\frac{\partial \rho}{\partial t} = -\frac{\partial}{\partial x}(\rho v), \quad (26)$$

$$\frac{\partial}{\partial t}(\rho v) = -\frac{\partial}{\partial x}(\rho v^2 + p - \tau), \quad (27)$$

$$\frac{\partial}{\partial t}(\rho E) = -\frac{\partial}{\partial x}(\rho E v + p v - v \tau + q). \quad (28)$$

The following closure equations arise from assuming a calorically perfect ideal gas, with zero bulk viscosity, and Fourier heat conduction:

$$\begin{aligned} \tau &= \frac{4}{3}\mu \frac{\partial v}{\partial x}, & q &= -k \frac{\partial T}{\partial x}, & e &= c_v T, \\ p &= (\gamma - 1)\rho e, & E &= e + \frac{1}{2}v^2. \end{aligned} \quad (29)$$

The ratio of the specific heats is $\gamma = 7/5$, and the other material properties are taken from tabulated values for dry air at 250 °K Heldman (2003). Table 2 lists the initial conditions, domain, and interface location x_i for this Riemann problem. The initial conditions are made continuous by using a hyperbolic tangent function.

	ρ [g/cm ³]	u [cm/s]	p [dyn/cm ²]
Left	1.0	0.0	1.0
Right	0.125	0.0	0.1
$x \in [0, 1]$ cm; $x_i = 0.5$ cm; $t \in [0, 0.2]$ s			

Table 2: Initial conditions and domain for the Sod shock tube from Kamm and et al. (2008).

The boundary conditions are set to maintain the initial conditions at each time step. The inviscid (i.e. $\mu = 0$) Sod problem has an analytical solution and it is shown against the viscid numerical solution for qualitative comparison. First order accurate forward Euler time integration is used with parameters $p = 8$ and $\varepsilon = 10^{-3}$ to obtain the approximate solutions (Figs. 13 to 15).

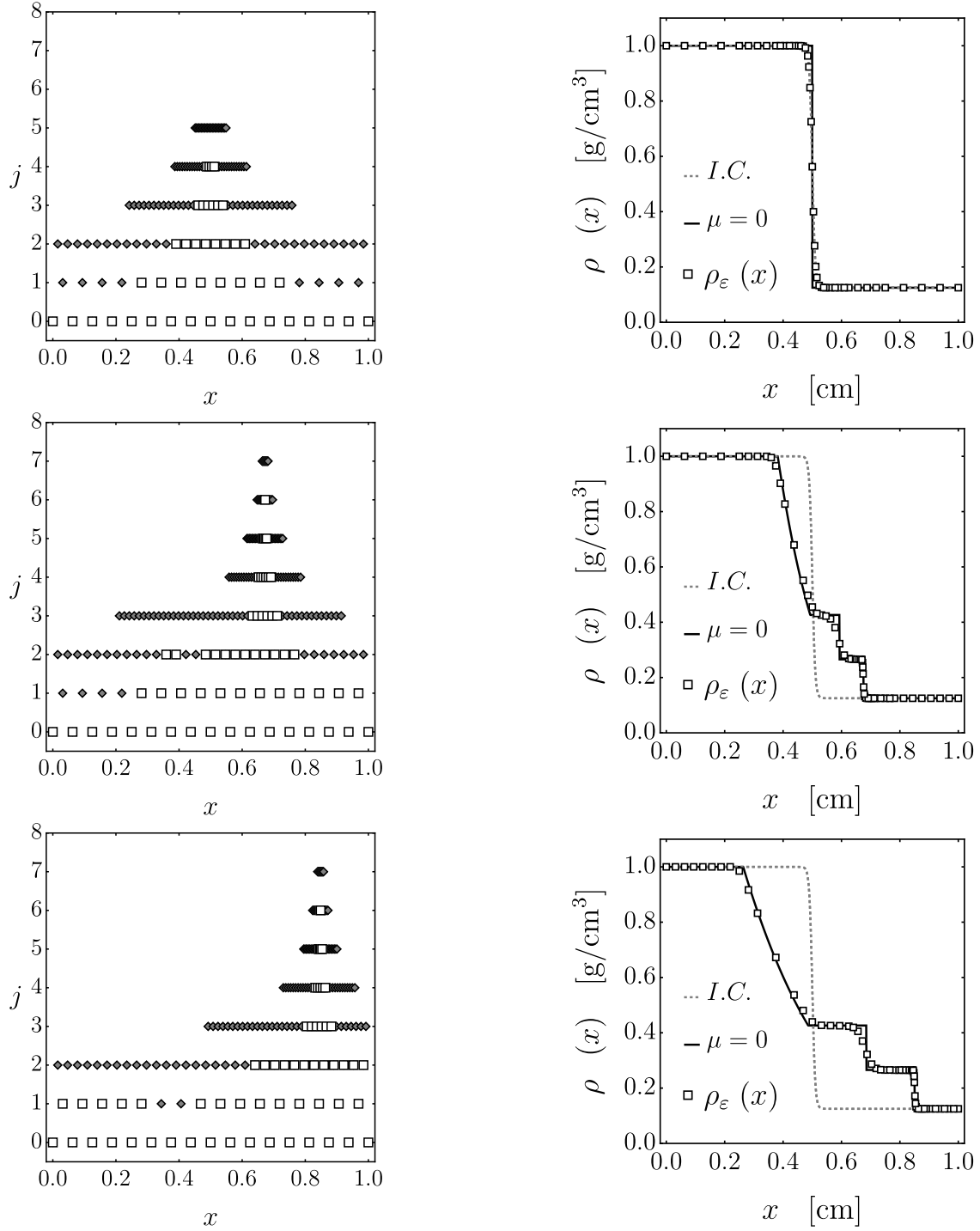


Figure 13: Sparse multiresolution grid and corresponding density values at times $t = 0.0$ s (top), $t = 0.1$ s (middle), and $t = 0.2$ s (bottom) with $p = 8$ and $\varepsilon = 10^{-3}$.

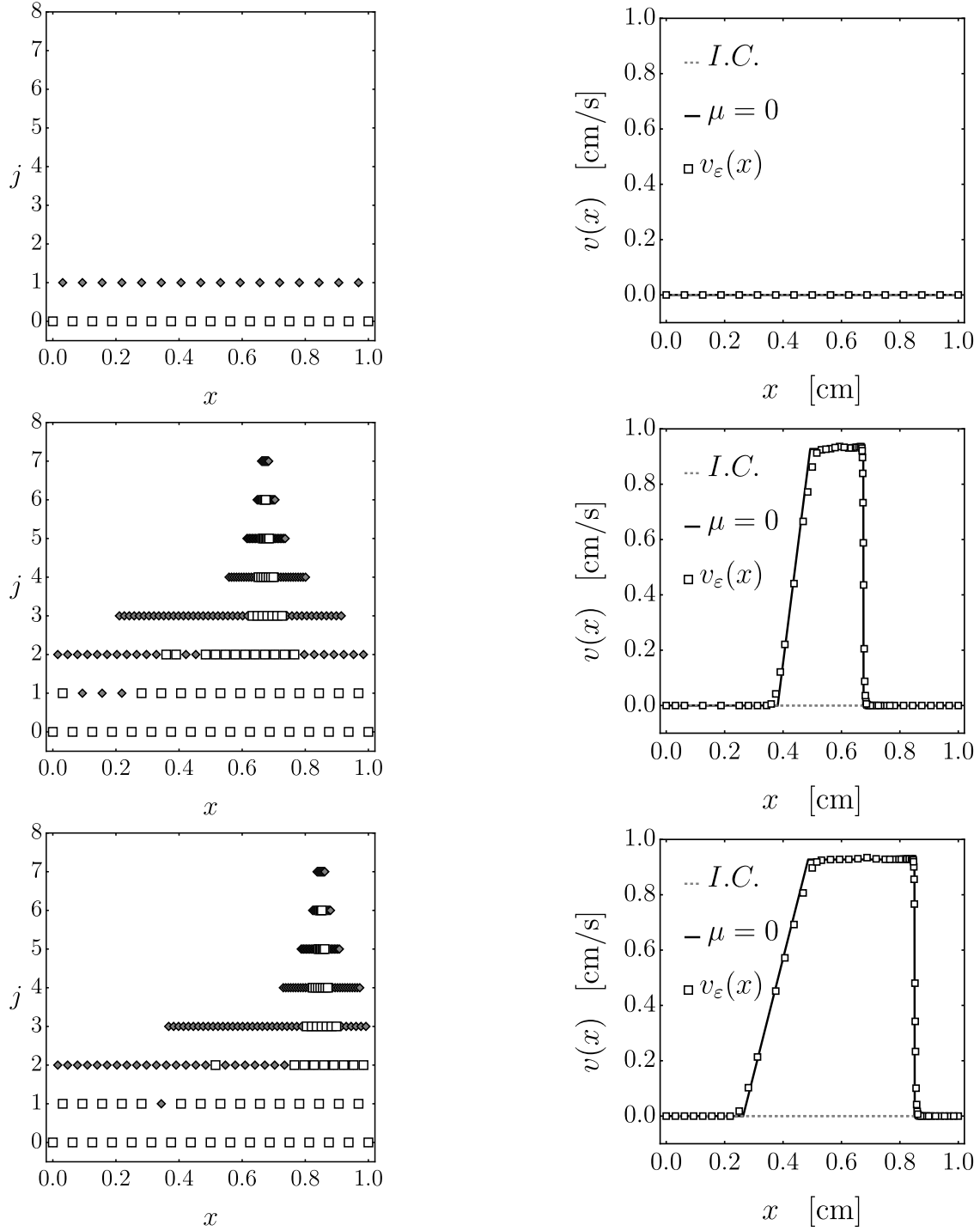


Figure 14: Sparse multiresolution grid and corresponding velocity values at times $t = 0.0$ s (top), $t = 0.1$ s (middle), and $t = 0.2$ s (bottom) with $p = 8$ and $\varepsilon = 10^{-3}$.

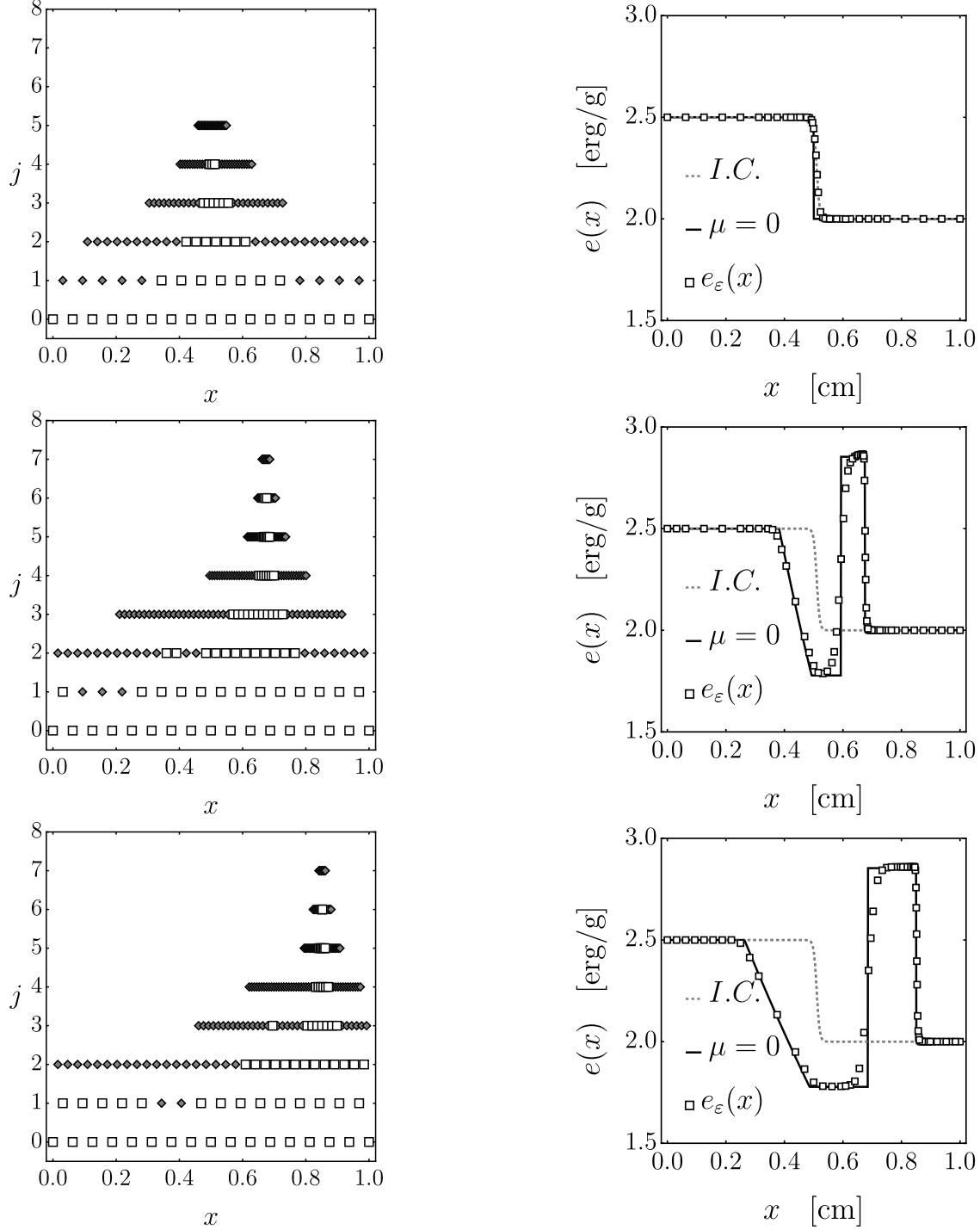


Figure 15: Sparse multiresolution grid and corresponding energy values at times $t = 0.0$ s (top), $t = 0.1$ s (middle), and $t = 0.2$ s (bottom) with $p = 8$ and $\varepsilon = 10^{-3}$.

As shown in Fig. 16, we allow each field to advance using its own sparse collocation grid. By retaining only the non-redundant information from grid to grid (field to field), a high degree of data compression is obtained. This is especially evident when comparing the initial computational grid of velocity against the initial grids for density and energy. Figure 16 shows the first 100

steps to highlight the transient development of each field. In total, 9,834 time steps were taken. On average, 200, 232, and 239 collocation points were needed to represent the density, velocity, and energy fields respectively. Vasilyev and Bowman (2000) defines a compression coefficient by dividing the number of points used in a uniform grid with comparable resolution by the number of points in the adaptive grid. This work achieves average compression coefficients of approximately 10.25, 8.83, and 8.57 for the density, velocity, and energy fields respectively.

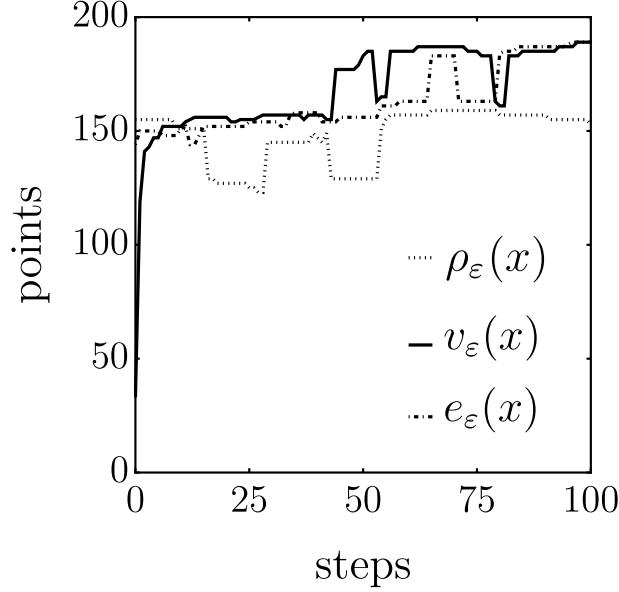


Figure 16: Number of collocation points needed for each field at each time step. Note, each grid adapts independently, as needed to satisfy the error bounds in Eqs. (6), (13) and (14).

5. Conclusions

In this work, we have developed an adaptive algorithm for solving nonlinear PDEs. We have incorporated a matrix notation to simplify the fundamental wavelet operations, and utilized a matrix-free computational implementation. We have shown that our numerical method is capable of solving initial-boundary value problems on finite domains with an explicit error control and negligible global energy growth. The algorithm takes advantage of the regularity of the biorthogonal interpolating wavelet family and evaluates spatial derivatives directly on the wavelet basis functions. We have advanced the state of wavelet based algorithms by deriving bounds on the spatial error of PDE solutions and developing a predictor-corrector strategy to ensure that the spatial error stays bounded at each time step. We have verified these error estimates through numerical analysis of nonlinear shock problems with analytical solutions. Furthermore, we have defined each field in the governing equations on its own dynamically adaptive computational grid and fine scale features, such as shock waves, are well resolved with no spurious numerical oscillations.

6. Acknowledgements

This work was supported by the Department of Energy, National Nuclear Security Administration, under the award No. DE-NA0002377 as part of the Predictive Science Academic Alliance

Program II. We would also like to acknowledge support from Los Alamos National Laboratory under the award No. 369229. Cale Harnish and Karel Matouš would like to thank Dr. S. Paolucci for fruitful discussions regarding the wavelet solutions of PDEs.

Appendices

A. Vector and Matrix Definitions

The vectors and matrices described in Section 2 are similar to those in Goedecker (1998); Jameson (1993), though in this work they are modified to account for finite domains. Spatial derivative calculations require the matrix operator $\mathbf{D}^{(m)}$, defined by:

$$\mathbf{D}^{(m)} = \mathbf{\Gamma}^{(m)} \quad \text{single resolution,} \quad (30)$$

$$\mathbf{D}^{(m)} = \mathbf{F} \cdot \mathbf{\Gamma}^{(m)} \cdot \mathbf{B} \quad \text{multiresolution,} \quad (31)$$

$$\mathbf{\Gamma}_{kl}^{(m)} = \int \tilde{\phi}_k^j(x) \frac{d^m}{dx^m} \phi_l^j(x) dx. \quad (32)$$

As in Qian and Weiss (1993), evaluating Eq. (32) for the interior scaling functions can be accomplished by solving an eigenvector problem. The eigenvector χ_i is then normalized according to

$$\sum_i i^m \chi_i = \left(\frac{-1}{\Delta x} \right)^m m!. \quad (33)$$

We note that Eq. (33) is derived in Beylkin (1992), and further scaled for a variable Δx . This scaling is unique to our method since we have modified the support interval of the basis in Section 2.1.

In Villiers et al. (2003), the modified boundary basis functions are defined as linear combinations of the interior bases. Therefore, evaluating Eq. (32) for a basis function near the boundary is accomplished by calculating an appropriate linear combination of the normalized eigenvector χ_i . With $\mathbf{\Gamma}^{(m)}$ fully defined, $\mathbf{D}^{(m)}$ is calculated in the “standard form” Goedecker (1998) by applying Eq. (31) at each resolution level.

B. Mathematical Derivations

The following mathematical formulations are used to estimate the spatial error associated with evaluating derivatives of the wavelet bases. Much of the literature on this subject has been focused on orthogonal wavelet families, with infinite or periodic domains, and a coarse grid spacing of unity (i.e. $\Delta x = 1$). Therefore, it is necessary to derive identities which pertain to our biorthogonal interpolating wavelet family, with modified bases on finite domains, and a variable coarse grid spacing (i.e. $\Delta x = \frac{b-a}{2^p}$).

The derivations consider a polynomial of an arbitrary order P in the domain $x \in [0, 1]$,

$$f(x) = x^P. \quad (34)$$

For non-trivial derivatives, $P \geq m$ and,

$$f^{(m)}(x) = \frac{P!}{(P-m)!} x^{P-m}. \quad (35)$$

Let p be the order of the wavelet basis functions, J be a maximum resolution level, and $N + 1$ is the number of collocation points on resolution level J where $N = 2^J(2p)$.

B.1. Moment Property of the Derivative Matrix

The following identity is required in Appendix B.2. Starting with,

$$\int \tilde{\phi}_k^J(x) \frac{d^m}{dx^m} x^P dx = \frac{P!}{(P-m)!} \left(\frac{k\Delta x}{2^J} \right)^{P-m}, \quad (36)$$

two cases develop, depending on the power of the polynomial.

- For P less than the order of the wavelet basis, (i.e. $P < p$):

$$x^P = \sum_{l=0}^N \left(l \frac{\Delta x}{2^J} \right)^P \phi_l^J(x), \quad (37)$$

$$\int \tilde{\phi}_k^J(x) \frac{d^m}{dx^m} x^P dx = \int \tilde{\phi}_k^J(x) \frac{d^m}{dx^m} \left(\sum_{l=0}^N \left(l \frac{\Delta x}{2^J} \right)^P \phi_l^J(x) \right) dx, \quad (38)$$

$$\int \tilde{\phi}_k^J(x) \frac{d^m}{dx^m} x^P dx = \left(\frac{\Delta x}{2^J} \right)^P \sum_{l=0}^N l^P \int \tilde{\phi}_k^J(x) \frac{d^m}{dx^m} \phi_l^J(x) dx, \quad (39)$$

$$\int \tilde{\phi}_k^J(x) \frac{d^m}{dx^m} x^P dx = \left(\frac{\Delta x}{2^J} \right)^P \sum_{l=0}^N l^P D_{kl}^{(m)}. \quad (40)$$

Setting Eqs. (36) and (40) equal yields,

$$\sum_{l=0}^N l^P D_{kl}^{(m)} = \frac{P!}{(P-m)!} \left(\frac{\Delta x}{2^J} \right)^{P-m}, \quad \text{for } P < p. \quad (41)$$

- For P greater than or equal to the order of the wavelet basis, (i.e. $P \geq p$):

$$x^P = \sum_{l=0}^N \left(l \frac{\Delta x}{2^J} \right)^P \phi_l^J(x) + \sum_{j=J+1}^{\infty} \sum_{l=1}^{2^j p} d_l^j \psi_l^j(x), \quad (42)$$

$$\int \tilde{\phi}_k^J(x) \frac{d^m}{dx^m} x^P dx = \int \tilde{\phi}_k^J(x) \frac{d^m}{dx^m} \left(\sum_{l=0}^N \left(l \frac{\Delta x}{2^J} \right)^P \phi_l^J(x) + \sum_{j=J+1}^{\infty} \sum_{l=1}^{2^j p} d_l^j \psi_l^j(x) \right) dx, \quad (43)$$

$$\int \tilde{\phi}_k^J(x) \frac{d^m}{dx^m} x^P dx = \left(\frac{\Delta x}{2^J} \right)^P \sum_{l=0}^N l^P D_{kl}^{(m)} + \int \tilde{\phi}_k^J(x) \frac{d^m}{dx^m} \left(\sum_{j=J+1}^{\infty} \sum_{l=1}^{2^j p} d_l^j \psi_l^j(x) \right) dx, \quad (44)$$

$$\int \tilde{\phi}_k^J(x) \frac{d^m}{dx^m} x^P dx = \left(\frac{\Delta x}{2^J} \right)^P \sum_{l=0}^N l^P D_{kl}^{(m)} + \sum_{j=J+1}^{\infty} \sum_{l=1}^{2^j p} d_l^j \int \tilde{\phi}_k^J(x) \frac{d^m}{dx^m} \psi_l^j(x) dx. \quad (45)$$

Setting Eqs. (36) and (45) equal yields,

$$\sum_{l=0}^N l^P D_{kl}^{(m)} = \frac{P!}{(P-m)!} k^{P-m} \left(\frac{\Delta x}{2^J} \right)^{-m} - \left(\frac{\Delta x}{2^J} \right)^{-P} \sum_{j=J+1}^{\infty} \sum_{l=1}^{2^j P} d_l^j \int \tilde{\phi}_k^J(x) \frac{d^m}{dx^m} \psi_l^j(x) dx. \quad (46)$$

B.2. Error Estimate for the Wavelet Derivative

Let $f_\varepsilon(x)$ be the wavelet representation of $f(x)$ and $f_\varepsilon^{(m)}(x)$ be the wavelet representation of the m^{th} derivative of $f(x)$. Then, subtract the discrete representation from the continuous to obtain,

$$\begin{aligned} & \left\| f^{(m)}(x) - \mathbf{D}^{(m)} f_\varepsilon(x) \right\|_\infty = \\ & \left\| f^{(m)}(x) - f_\varepsilon^{(m)}(x) + f_\varepsilon^{(m)}(x) - \mathbf{D}^{(m)} f_\varepsilon(x) \right\|_\infty \leq \\ & \left\| f^{(m)}(x) - f_\varepsilon^{(m)}(x) \right\|_\infty + \left\| f_\varepsilon^{(m)}(x) - \mathbf{D}^{(m)} f_\varepsilon(x) \right\|_\infty. \end{aligned} \quad (47)$$

The norms on the right hand side of Eq. (47) have different expressions depending on the power P of the polynomial. The first norm corresponds to the interpolation error of a wavelet representation truncated at resolution level J . This error estimate has been derived in Donoho (1992) to be,

$$\left\| f^{(m)}(x) - f_\varepsilon^{(m)}(x) \right\|_\infty = 0, \quad \text{for } P < p, \quad (48)$$

$$\left\| f^{(m)}(x) - f_\varepsilon^{(m)}(x) \right\|_\infty \leq C_1 \left(\frac{\Delta x}{2^J} \right)^p, \quad \text{for } P \geq p. \quad (49)$$

The second norm on the right hand side of Eq. (47) corresponds to the error from projecting the derivatives of the basis functions onto the same basis functions.

$$f_\varepsilon(x) = \sum_{k=0}^N \left(k \frac{\Delta x}{2^J} \right)^P \phi_k^J(x), \quad (50)$$

$$f_\varepsilon^{(m)}(x) = \sum_{k=0}^N \frac{P!}{(P-m)!} \left(k \frac{\Delta x}{2^J} \right)^{P-m} \phi_k^J(x), \quad (51)$$

$$\begin{aligned} & \left\| f_\varepsilon^{(m)}(x) - \mathbf{D}^{(m)} f_\varepsilon(x) \right\|_\infty = \\ & \left\| \sum_{k=0}^N \frac{P!}{(P-m)!} \left(k \frac{\Delta x}{2^J} \right)^{P-m} \phi_k^J(x) - \sum_{k=0}^N \sum_{l=0}^N D_{kl}^{(m)} \left(l \frac{\Delta x}{2^J} \right)^P \phi_k^J(x) \right\|_\infty, \end{aligned} \quad (52)$$

$$\begin{aligned} & \left\| f_\varepsilon^{(m)}(x) - \mathbf{D}^{(m)} f_\varepsilon(x) \right\|_\infty = \\ & \left\| \sum_{k=0}^N \left\{ \frac{P! k^{P-m}}{(P-m)!} \left(\frac{\Delta x}{2^J} \right)^{P-m} - \left(\frac{\Delta x}{2^J} \right)^P \sum_{l=0}^N l^P D_{kl}^{(m)} \right\} \phi_k^J(x) \right\|_\infty. \end{aligned} \quad (53)$$

Proceeding, there are two cases,

- For $P < p$, using Eq. (41), Eq. (53) becomes:

$$\left\| f_{\varepsilon}^{(m)}(x) - \mathbf{D}^{(m)} f_{\varepsilon}(x) \right\|_{\infty} = \left\| \sum_{k=0}^N \left\{ \frac{P! k^{P-m}}{(P-m)!} \left(\frac{\Delta x}{2^j} \right)^{P-m} - \frac{P! k^{P-m}}{(P-m)!} \left(\frac{\Delta x}{2^J} \right)^{P-m} \right\} \phi_k^J(x) \right\|_{\infty}, \quad (54)$$

$$\left\| f_{\varepsilon}^{(m)}(x) - \mathbf{D}^{(m)} f_{\varepsilon}(x) \right\|_{\infty} = 0. \quad (55)$$

- For $P \geq p$, using Eq. (46), Eq. (53) becomes:

$$\begin{aligned} \left\| f_{\varepsilon}^{(m)}(x) - \mathbf{D}^{(m)} f_{\varepsilon}(x) \right\|_{\infty} = & \left\| \sum_{k=0}^N \left\{ \frac{P! k^{P-m}}{(P-m)!} \left(\frac{\Delta x}{2^j} \right)^{P-m} - \frac{P! k^{P-m}}{(P-m)!} \left(\frac{\Delta x}{2^J} \right)^{P-m} \dots \right. \right. \\ & \left. \left. - \sum_{j=J+1}^{\infty} \sum_{l=1}^{2^j p} d_l^j \int \tilde{\phi}_k^J(x) \frac{d^m}{dx^m} \psi_l^j(x) dx \right\} \phi_k^J(x) \right\|_{\infty}, \end{aligned} \quad (56)$$

$$\left\| f_{\varepsilon}^{(m)}(x) - \mathbf{D}^{(m)} f_{\varepsilon}(x) \right\|_{\infty} = \left\| \sum_{k=0}^N \sum_{j=J+1}^{\infty} \sum_{l=1}^{2^j p} -d_l^j \left(\int \tilde{\phi}_k^J(x) \frac{d^m}{dx^m} \psi_l^j(x) dx \right) \phi_k^J(x) \right\|_{\infty}, \quad (57)$$

$$\left\| f_{\varepsilon}^{(m)}(x) - \mathbf{D}^{(m)} f_{\varepsilon}(x) \right\|_{\infty} = \left\| \sum_{k=0}^N \sum_{j=J+1}^{\infty} \sum_{l=1}^{2^j p} d_l^j \left(\int \tilde{\phi}_k^J(x) \frac{d^m}{dx^m} \phi_{2l+1}^{j+1}(x) dx \right) \phi_k^J(x) \right\|_{\infty}, \quad (58)$$

$$\left\| f_{\varepsilon}^{(m)}(x) - \mathbf{D}^{(m)} f_{\varepsilon}(x) \right\|_{\infty} = \left\| \sum_{k=0}^N \sum_{j=J+1}^{\infty} \sum_{l=1}^{2^j p} d_l^j \frac{2^{jm} 2^m}{2^J} \left(\int \tilde{\phi}_k^J(x) \frac{d^m}{dx^m} \phi_{2l+1}^J(x) dx \right) \phi_k^J(x) \right\|_{\infty}, \quad (59)$$

$$\left\| f_{\varepsilon}^{(m)}(x) - \mathbf{D}^{(m)} f_{\varepsilon}(x) \right\|_{\infty} = 2^{m-J} \left\| \sum_{k=0}^N \sum_{j=J+1}^{\infty} 2^{jm} \sum_{i=3}^{1+p2^{j+1}} d_{\frac{i-1}{2}}^j D_{ki}^{(m)} \phi_k^J(x) \right\|_{\infty}. \quad (60)$$

As shown in Goedecker (1998), the d_i^j coefficients are identical to the Lagrange remainder (defined in Abramowitz and Stegun (1964)).

$$\left\| f_{\varepsilon}^{(m)}(x) - \mathbf{D}^{(m)} f_{\varepsilon}(x) \right\|_{\infty} = 2^{m-J} \left\| \sum_{k=0}^N \sum_{j=J+1}^{\infty} 2^{jm} \sum_{i=3}^{1+p2^{j+1}} C_i \left(\frac{\Delta x}{2^j} \right)^p D_{ki}^{(m)} \phi_k^J(x) \right\|_{\infty}. \quad (61)$$

Then, since the norm is less than or equal to the sum of the norms for each i , and the components of the $\mathbf{D}^{(m)}$ matrix come from the normalized eigenvectors χ_i in Eq. (33), we obtain

$$\left\| f_{\varepsilon}^{(m)}(x) - \mathbf{D}^{(m)} f_{\varepsilon}(x) \right\|_{\infty} \leq 2^{m-J} \left\| \sum_{k=0}^N \sum_{j=J+1}^{\infty} \left(\frac{\Delta x}{2^j} \right)^p 2^{jm} C_3 \left(\frac{1}{\Delta x} \right)^m \phi_k^J(x) \right\|_{\infty}. \quad (62)$$

Again, since the norm is less than or equal to the sum of the norms for each j , we obtain

$$\left\| f_\varepsilon^{(m)}(x) - \mathbf{D}^{(m)} f_\varepsilon(x) \right\|_\infty \leq \frac{2^{2m+Jm-J}}{2^p - 2^m} \left(\frac{\Delta x}{2^J} \right)^p \left(\frac{1}{\Delta x} \right)^m \left\| \sum_{k=0}^N C_3 \phi_k^J(x) \right\|_\infty, \quad (63)$$

$$\left\| f_\varepsilon^{(m)}(x) - \mathbf{D}^{(m)} f_\varepsilon(x) \right\|_\infty \leq C_2 \left(\frac{\Delta x}{2^J} \right)^{p-m}. \quad (64)$$

Now Eq. (47) has the following forms,

- For $P < p$, use Eqs. (48) and (55) to obtain

$$\left\| f^{(m)}(x) - \mathbf{D}^{(m)} f_\varepsilon(x) \right\|_\infty = 0. \quad (65)$$

- For $P \geq p$, use Eqs. (49) and (64) to obtain

$$\left\| f^{(m)}(x) - \mathbf{D}^{(m)} f_\varepsilon(x) \right\|_\infty \leq C_2 \left(\frac{\Delta x}{2^J} \right)^{p-m}. \quad (66)$$

This estimate takes into account the modified support interval of our bases and the modified bases near the spatial boundaries. As shown in Holmström (1999), the grid spacing at the highest resolution level, $h = \Delta x/2^J$, may be related to the thresholding parameter ε with

$$O(h) \approx O(\varepsilon^{\frac{1}{p}}), \quad (67)$$

to obtain

$$\left\| f^{(m)}(x) - \mathbf{D}^{(m)} f_\varepsilon(x) \right\|_\infty \leq O\left(\varepsilon^{1-\frac{m}{p}}\right). \quad (68)$$

Bibliography

- M. Abramowitz and I. Stegun. *Handbook of Mathematical Functions*. United States Department of Commerce, 1964.
- B. Alpert, G. Beylkin, D. Hines, and L. Vozovoi. Adaptive solution of partial differential equations in multiwavelet bases. *Journal of Computational Physics*, 182:149–190, 2002. doi:10.1006/jcph.2002.7160.
- E. Bacry, S. Mallat, and B. Papanicolaou. A wavelet based space-time adaptive numerical method for partial differential equations. *Mathematical Modelling and Numerical Analysis*, 26(7):793–834, 1992.
- M. Berger and J. Oliger. Adaptive mesh refinement for hyperbolic partial differential equations. *Journal of Computational Physics*, 53:484–512, 1984. doi:10.1016/0021-9991(84)90073-1.
- S. Bertoluzza. Adaptive wavelet collocation method for the solution of burgers equation. *Transport Theory and Statistical Physics*, 25:339–352, 1996. doi:10.1080/00411459608220705.
- S. Bertoluzza and G. Naldi. A wavelet collocation method for the numerical solution of partial differential equations. *Applied and Computational Harmonic Analysis*, 3:1–9, 1996. doi:10.1006/acha.1996.0001.
- G. Beylkin. On the representation of operators in bases of compactly supported wavelets. *SIAM J. Numer. Anal.*, 6(6):1716–1740, 1992. doi:10.1137/0729097.
- G. Beylkin and J. Keiser. On the adaptive numerical solution of nonlinear partial differential equations in wavelet bases. *Journal of Computational Physics*, 132:233–259, 1997. doi:10.1006/jcph.1996.5562.
- M. Boslough, B. Jennings, B. Carvey, and W. Fogleman. FEMA asteroid impact tabletop exercise simulations. *Procedia Engineering*, 103:43–51, 2015. doi:10.1016/j.proeng.2015.04.007.
- A. Brandt. Multi-level adaptive solutions to boundary-value problems. *Mathematics of Computation*, 31(138):333–390, 1977. doi:10.1090/S0025-5718-1977-0431719-X.
- R. B. Burgos, M. A. C. Santos, and R. R. e Silva. Deslauriers-dubuc interpolating wavelet beam finite element. *Journal of Elements in Analysis and Design*, 75:71–77, 2013. doi:10.1016/j.fincl.2013.07.004.
- X. Cai, J. Liang, R. Deiterding, Y. Che, and Z. Lin. Adaptive mesh refinement based simulations of three-dimensional detonation combustion in supersonic combustible mixtures with a detailed reaction model. *International Journal of Hydrogen Energy*, 41:3222–3239, 2016. doi:10.1016/j.ijhydene.2015.11.093.
- A. Cohen, W. Dahmen, and R. DeVore. Adaptive wavelet methods for elliptic operator equations: Convergence rates. *Mathematics of Computation*, 70(233):27–75, 2000a. doi:10.1090/S0025-5718-00-01252-7.

- A. Cohen, W. Dahmen, and R. DeVore. Multiscale decompositions on bounded domains. *Transactions of the American Mathematical Society*, 352(8):3651–3685, 2000b. doi:10.1090/S0002-9947-00-02112-7.
- W. Dahmen, A. Kunoth, and K. Urban. Biorthogonal spline wavelets on the interval - stability and moment conditions. *Applied and Computational Harmonic Analysis*, 6:132–196, 1999. doi:10.1006/acha.1998.0247.
- I. Daubechies. *Ten Lectures on Wavelets*. SIAM, Philadelphia, 1992.
- L. Demkowicz, J. T. Oden, W. Rachowicz, and O. Hardy. Toward a universal $h - p$ adaptive finite element strategy, part 1. constrained approximation and data structure. *Computer Methods in Applied Mechanics and Engineering*, 77:79–112, 1989. doi:10.1016/0045-7825(89)90129-1.
- J. E. Dendy. Black box multigrid. *Journal of Computational Physics*, 48:366–386, 1982. doi:10.1016/0021-9991(82)90057-2.
- S. Dong and G. E. Karniadakis. P-refinement and p-threads. *Computer Methods in Applied Mechanics and Engineering*, 192:2191–2201, 2003. doi:10.1016/S0045-7825(02)00653-9.
- D. Donoho. Interpolating wavelet transforms. *Preprint, Department of Statistics, Stanford University*, 2(3), 1992.
- T. Dubos and N. K-R. Kevlahan. A conservative adaptive wavelet method for the shallow-water equations on staggered grids. *Quarterly Journal of the Royal Meteorological Society*, 139:1997–2020, 2013. doi:10.1002/qj.2097.
- M. Farge. Wavelet transforms and their applications to turbulence. *Annual review of fluid mechanics*, 24:395–457, 1992. doi:10.1146/annurev.fl.24.010192.002143.
- I. Fatkullin and J. S. Hesthaven. Adaptive high-order finite-difference method for nonlinear wave problems. *Journal of Scientific Computing*, 16(1):47–67, 2001. doi:10.1023/A:1011198413865.
- J. Fröhlich and K. Schneider. An adaptive wavelet galerkin algorithm for one-and two-dimensional flame computations. *European Journal of Mechanics B-Fluids*, 13:439–471, 1994.
- M. Fujii and W. J. R. Hoefer. Interpolating wavelet collocation method of time dependent maxwell’s equations: Characterization of electrically large optical waveguide discontinuities. *Journal of Computational Physics*, 186:666–689, 2003. doi:10.1016/S0021-9991(03)00091-3.
- S. Goedecker. *Wavelets and their application for the solution of partial differential equations in physics*. Presses polytechniques et universitaires romandes, Lausanne, Switzerland, 1998.
- W. Gui and I. Babuška. The h , p , and h - p versions of the finite element method in 1 dimension, part i. *Numer. Math.*, 49:577–612, 1986a. doi:10.1007/BF01389733.
- W. Gui and I. Babuška. The h , p , and h - p versions of the finite element method in 1 dimension, part ii. *Numer. Math.*, 49:613–657, 1986b. doi:10.1007/BF01389734.

- W. Hackbusch. On the multi-grid method applied to difference equations. *Computing*, 20:291–306, 1978. doi:10.1007/BF02252378.
- D. Heldman. *Encyclopedia of Agricultural, Food, and Biological Engineering*. Marcel Dekker, New York, NY, 2003.
- M. Holmström. Solving hyperbolic PDEs using interpolating wavelets. *SIAM J. Sci. Comput.*, 21(2):405–420, 1999. doi:10.1137/S1064827597316278.
- A. Iqbal and V. Jeoti. An improved split-step wavelet transform method for anomalous radio wave propagation modeling. *Radio Engineering*, 23(4):987–996, 2014.
- L. Jameson. On the daubechies-based wavelet differentiation matrix. ICASE Report 93-95, NASA, Langley Research Center, Hapton, Virginia, December 1993. NASA Contractor Report 191583.
- B. Jawerth and W. Sweldens. An overview of wavelet based multiresolution analyses. *SIAM Review*, 36(3):377–412, 1994. doi:10.1137/1036095.
- J. R. Kamm and et al. *Enhanced Verification Test Suite for Physics Simulation Codes*. Los Alamos National Laboratory, LANL report no. LA-14379, 2008.
- R. Klein. Star formation with 3-D adaptive mesh refinement: the collapse and fragmentation of molecular clouds. *Journal of Computational and Applied Mathematics*, 109:123–152, 1999. doi:10.1016/S0377-0427(99)00156-9.
- F. Kong, I. Kougoumtzoglou, P. Spanos, and S. Li. Nonlinear system response evolutionary power spectral density determination via a harmonic wavelets based galerkin technique. *International Journal for Multiscale Computational Engineering*, 14(3):255–272, 2016. doi:10.1615/IntJMultCompEng.2016016464.
- T. Le and L. Caracoglia. Reduced-order wavelet-galerkin solution for the coupled, nonlinear stochastic response of slender buildings in transient winds. *Journal of Sound and Vibration*, 344:179–208, 2015. doi:10.1016/j.jsv.2015.01.007.
- J. Liandrat and P. Tchamitchian. Resolution of the 1d regularized burgers equation using a spatial wavelet approximation. ICASE Report 90-83, NASA, Langley Research Center, Hapton, Virginia, December 1990. NASA Contractor Report 187480.
- E. B. Lin and X. Zhou. Connection coefficients on an interval and wavelet solutions of burgers equation. *Journal of Computational and Applied Mathematics*, 135:63–78, 2001. doi:10.1016/S0377-0427(00)00562-8.
- C. Malone, A. Nonaka, S. Woosley, A. Almgren, J. Bell, S. Dong, and M. Zingale. The deflagration stage of chandrasekhar mass models for type 1a supernovae. i. early evolution. *The Astrophysical Journal*, 782(1):1–24, 2014. doi:10.1088/0004-637X/782/1/11.
- K. McCormick and R. O. Wells. Wavelet calculus and finite difference operators. *Mathematics of Computation*, 63(207):155–173, 1994. doi:10.2307/2153567.

- A. Nejadmalayeri, A. Vezolainen, E. Brown-Dymkoski, and O. Vasilyev. Parallel adaptive wavelet collocation method for PDEs. *Journal of Computational Physics*, 298:237–253, 2015. doi:10.1016/j.jcp.2015.05.028.
- S. Paolucci, Z. Zikoski, and T. Gredga. WAMR: An adaptive wavelet method for the simulation of compressible reacting flow. part ii. the parallel algorithm. *Journal of Computational Physics*, 272:842–864, 2014a. doi:10.1016/j.jcp.2014.03.059.
- S. Paolucci, Z. Zikoski, and D. Wirasaet. WAMR: An adaptive wavelet method for the simulation of compressible reacting flow. part i. efficiency and accuracy of algorithm. *Journal of Computational Physics*, 272:814–841, 2014b. doi:10.1016/j.jcp.2014.01.025.
- S. Qian and J. Weiss. Wavelets and the numerical solution of partial differential equations. *Journal of Computational Physics*, 106:155–175, 1993. doi:10.1006/jcph.1993.1100.
- A. Rajagopal and S. M. Sivakumar. A combined r-h adaptive strategy based on material forces and error assessment for plane problems and bimaterial interfaces. *Comput. Mech.*, 41:49–72, 2007. doi:10.1007/s00466-007-0168-8.
- T. Ringler, M. Petersen, R. Higdon, D. Jacobsen, P. Jones, and M. Maltrud. A multi-resolution approach to global ocean modeling. *Ocean Modelling*, 69:211–232, 2013. doi:10.1016/j.ocemod.2013.04.010.
- O. Rioul. Simple regularity criteria for subdivision schemes. *SIAM J. Math. Anal.*, 23(6):1544–1576, 1992. doi:10.1137/0523086.
- T. Sakurai, K. Yoshimatsu, K. Schneider, M. Farge, K. Morishita, and T. Ishihara. Coherent structure extraction in turbulent channel flow using boundary adapted wavelets. *Journal of Turbulence*, 18(4):352–372, 2017. doi:10.1080/14685248.2017.1284326.
- D. Schillinger and E. Rank. An unfitted hp-adaptive finite element method based on hierarchical b-splines for interface problems of complex geometry. *Comput. Methods Appl. Mech. Engrg.*, 200:3358–3380, 2011. doi:10.1016/j.cma.2011.08.002.
- K. Schneider and O. Vasilyev. Wavelet methods in computational fluid dynamics. *Annual Review of Fluid Mechanics*, 42:473–503, 2010. doi:10.1146/annurev-fluid-121108-145637.
- K. Segeth. A review of some a posteriori error estimates for adaptive finite element methods. *Mathematics and Computers in Simulation*, 80:1589–1600, 2010. doi:10.1016/j.matcom.2008.12.019.
- W. Sweldens. The lifting scheme: a construction of second generation wavelets. *SIAM J. Math. Anal.*, 29(2):511–546, 1998. doi:10.1137/S0036141095289051.
- A. Thekale, T. Gradl, K. Klamroth, and U. Rüde. Optimizing the number of multigrid cycles in the full multigrid algorithm. *Numerical Linear Algebra with Applications*, 17:199–210, 2010. doi:10.1002/nla.697.

- T. Ueno, T. Ide, and M. Okada. A wavelet collocation method for evolution equations with energy conservation property. *Bulletin des Sciences Mathematiques*, 127:569–583, 2003. doi:10.1016/S0007-4497(03)00044-7.
- O. Vasilyev and C. Bowman. Second-generation wavelet collocation method for the solution of partial differential equations. *Journal of Computational Physics*, 165:660–693, 2000. doi:10.1006/jcph.2000.6638.
- O. Vasilyev and S. Paolucci. A dynamically adaptive multilevel wavelet collocation method for solving partial differential equations in a finite domain. *Journal of Computational Physics*, 125:498–512, 1996. doi:10.1006/jcph.1996.0111.
- J. M. De Villiers, K. M. Goosen, and B. M. Herbst. Dubuc-deslauriers subdivision for finite sequences and interpolation wavelets on an interval. *SIAM J. Math. Anal.*, 35(2):423–452, 2003. doi:10.1137/S0036141001386830.