An Adaptive Pseudo-Wavelet Approach for Solving Nonlinear Partial Di erential Equations

Gregory Beylkin and James M. Keiser

Wavelet Analysis and Applications, v.6, 1997, Academic Press.

Contents

An Adaptive Pseudo-Wavelet Approach for Solving Nonlinear Partial Di erential Equations

Gregory Beylkin **James M. Keiser**
Department of Applied Mathematics
University of Colorado
Boulder, CO 80309-0526

Abstract

We numerically solve nonlinear partial differential equations of the form $u_t = \mathcal{L}u + \mathcal{N}f(u)$ where $\mathcal{L}$ and $\mathcal{N}$ are linear differential operators and $f(u)$ is a nonlinear function. Equations of this form arise in the mathematical description of a number of phenomena including, for example, signal processing schemes based on solving partial differential equations or integral

oscillatory solutions and can exhibit shock-like behavior. Generally speaking, the approach takes advantage of the e cient representation of functions and operators in wavelet bases, and updates the solution by implementing two recently developed adaptive algorithms that operate on these representations. Speci cally, the algorithms involve the adaptive application of operators to functions (`special' matrix-vector multiplication) and the adaptive evaluation of nonlinear functions of the solution of the PDE, in particular, the pointwise product. These algorithms use the fact that wavelet expansions may be viewed as a localized Fourier analysis with multiresolution structure that automatically or adaptively distinguishes between smooth and shock-like behavior. The algorithms are adaptive since they update the solution using its representation in a wavelet basis, which concentrates significant coe cients near singular behaviour. Additionally, and as we will show, the algorithm for evaluating nonlinear functions is analogous to the approach used to update the solution of a PDE via pseudo-spectral type algorithms. These two features of the algorithms allow us to combine the desirable features of nite-di erence approaches, spectral methods and front-tracking or adaptive grid approaches into a collection of e cient, generic algorithms. We refer to the overall methodology for updating the solution of a nonlinear PDE via these algorithms as an *adaptive pseudo-wavelet method*.

In this Chapter we are concerned with computing numerical solutions of

$$u_t = Lu + N f(u), \tag{1.1}$$

with the initial condition

$$u(x,0) = u_0(x), \qquad 0 \quad x \quad 1, \tag{1.2}$$

and the periodic boundary condition

$$u(0,t) = u(1,t), \qquad 0 \quad t \quad T. \tag{1.3}$$

We explicitly separate the evolution Equation (1.1) into a linear part, Lu, and a nonlinear part, $N f(u)$, where the operators L and N are di erential operators that do not depend on time t. The function $f(u)$ is typically nonlinear, e.g. $f(u) = u^p$.

Examples of Equation (1.1) in 1+1 dimensions include reaction-di usion equations, e.g.

$$u_t = \nu u_{xx} + u^p, \qquad p > 1, \quad \nu > 0, \tag{1.4}$$

equations describing the buildup and propagation of shocks, e.g. Burgers' Equation

$$u_t + uu_x = \nu u_{xx}, \qquad \nu > 0, \tag{1.5}$$

[15], and equations having special $soliton$ solutions, e.g. the Korteweg-de Vries equation

$$u_t + \quad uu_x + \beta u_{xxx} = 0, \tag{1.6}$$

where and β are constant, [1, 24]. Finally, a simple example of Equation (1.1) is the classical di usion (or heat) equation

$$u_t = \nu u_{xx}, \qquad \nu > 0. \tag{1.7}$$

Although we do not address multi-dimensional problems in this Chapter, we note that the Navier-Stokes equations may also be written in the form (1.1). Consider

$$\mathbf{u}_t + \tfrac{1}{2}[\mathbf{u} \quad \mathbf{r}\,\mathbf{u} + \mathbf{r}\,(\mathbf{u} \quad \mathbf{u})] = \nu\,\mathbf{r}^2\mathbf{u} \quad \mathbf{r}\,p, \tag{1.8}$$

where

where $H(\ )$ is the Hilbert transform (see [18

straigh

tor) depends on the most singular behavior of the function. Since we are interested in solutions of partial differential equations that have regions of smooth, non-oscillatory behavior interrupted by a number of well-defined localized shocks or shock-like structures, using a basis of the eigenfunctions of differential operators would require a large number of terms due to the singular regions. Alternately, a localized representation of the solution, typified by front-tracking or adaptive grid methods, may be employed in order to distinguish between smooth and shock-like behavior. In our approach the number of operations is proportional to the number of significant coefficients in the wavelet expansions of functions and operators and, thus, is similar to that of adaptive grid methods.

The basic mechanism of refinement in wavelet-based algorithms is very simple. Due to the vanishing moments of wavelets, see e.g. [22], we know that (for a given accuracy) the wavelet transform of a function `automatically' places significant coefficients in a neighborhood of large gradients present in the function. We simply remove coefficients below a given accuracy threshold. This combination of basis expansion and adaptive thresholding is the foundation for our adaptive pseudo-wavelet approach.

In order to take advantage of this `adaptive transform' and compute solutions of (1.1) in wavelet bases using $O(N_s)$ operations, we have developed two algorithms: the adaptive application of operators to functions, and the adaptive pointwise product of functions. These algorithms are necessary ingredients of any fast, adaptive numerical scheme for computing solutions of partial differential equations. The algorithm for adaptively multiplying operators and functions is based on a `vanishing-moment property' associated with the B-blocks of the so-called Non-Standard Form representation of a class of operators (which includes differential operators and Hilbert transforms). The algorithm for adaptively computing $f(u)$, e.g. the pointwise product, is analogous to the method for evaluating nonlinear contributions in pseudo-spectral schemes. The spectral expansion of u is projected onto a `physical' subspace, the function $f(u)$ is evaluated, and the result is projected into the spectral domain. In our algorithm, contributions to $f(u)$ are adaptively computed in `pieces' on individual subspaces.

Each of our adaptive algorithms uses $O(N_s)$ operations, where N_s is the number of significant coefficients of the wavelet representation of the solution of (1.1). The adaptivity of our algorithms and the analogy with pseudo-spectral methods, prompts us to refer to our overall approach as an *adaptive pseudo-wavelet method*.

The outline of this Chapter is as follows. In Section 2 we use the semigroup approach to replace the nonlinear di erential equation (1.1) by an integral equation and describe a procedure for approximating the integral to any order of accuracy. We provide a brief review of wavelet \tools" relevant to our discussion in Section 3. In Section 4 we are concerned with the construction of and calculations with the operators appearing in the quadrature formulas derived in Section 2. Speci cally, we describe a method for constructing the wavelet representation, derive the vanishing-moment property, and describe a fast, adaptive algorithm for applying these operators to functions expanded in a wavelet basis. In Section 5 we introduce a new adaptive algorithm for computing the pointwise product of functions expanded in a wavelet basis, and discuss the calculation of general nonlinear functions. In Sections 4 and 5 we give simple numerical examples illustrating the algorithms. In Section 6 we illustrate the use of these algorithms by providing the results of a number of numerical experiments. Finally, in Section 7 we draw a number of conclusions based on our results and

In this Chapter we use Equation (2.14) as a starting point for an e cient numerical algorithm for solving (1.1). A signi cant di culty in designing numerical algorithms based directly on (2.14) is that the matrices representing these operators are dense in the ordinary representation. As far as we know, it is for this reason that the semigroup approach has had limited use in numerical calculations. We show in Sections 4.1 and 4.2 that in the wavelet system of coordinates these operators are sparse (for a xed but arbitrary accuracy) and have properties that allow us to develop fast, adaptive numerical algorithms. Discrete evolution schemes for (2.14) were used in [11], and further investigated in [12].

The starting point for our discrete evolution scheme is (2.14) where we consider the function $u(x,t)$ at the discrete moments of time t

I is the identity operator and where $u(t_i) = u_i$ and $v(t_i) = v_i$. Note that (2.17) is equivalent to the standard trapezoidal rule. For $m = 2$ our procedure yields an analogue of Simpson's rule

$$I(t) = \sum_{i=0}^{2} c_{i;i} u(t_i) u_x(t_i) + O((\quad t)^3), \tag{2.20}$$

where

$$c_{0;0} = \tfrac{1}{6} O_{\mathcal{L};2} \quad \tfrac{1}{3} L, \tag{2.21}$$

$$c_{1;1} = \tfrac{2}{3} O_{\mathcal{L};2}, \tag{2.22}$$

$$c_{2;2} = \tfrac{1}{6} O_{\mathcal{L};2} + \tfrac{1}{3} L, \tag{2.23}$$

For the derivation of higher order schemes ($m > 2$) and the stability analysis of these schemes we refer to [12], since our goals in this Chapter are limited to explaining how to make e ective use of such schemes in adaptive algorithms.

Preli inaries and Conventions of avelet Anal ysis

In this Section we review the relevant material associated with wavelet basis expansions of functions and operators. In Section 3.1 we set a system of notation associated with multiresolution analysis. In Section 3.2 we describe the representation of functions expanded in wavelet bases, and in Section 3.3 we describe the representation of operators in the standard and non-standard forms. In Section 3.4 we discuss the construction of the non-standard form of di erential operators, follonon-pp0.069 0 Td927 0 Td ited.02732 04 Td (adaptiv)Tj 3

not have to be nite and, by choosing $L_{\mathbf{f}} < 1$, we are selecting compactly supported wavelets, see, e.g. [22].

The function $\psi(\,)$ has M vanishing moments, i.e.,

$$\int_{-\infty}^{\infty} \psi(x$$

bases, via the two-scale di erence equations

where P_j denotes the projection operator onto subspace V_j. The set of coefficients $\{s_k^j\}_{k\in\mathbb{Z}}$, which we refer to as `averages', is computed via the inner product

$$s_k^j = \int_{-\infty}^{+\infty} f(x)\,\phi_{j;k}(x)\,dx. \tag{3.37}$$

Alternatively, it follows from (3.26) and (3.36) that we can also write $(P_j f)(x)$ as a sum of projections of $f(x)$ onto subspaces $W_{j'}, j' > j$

$$(P_j f)(x) = \sum_{j'>j}\sum_{k\in\mathbb{Z}} d_k^{j'}\psi_{j';k}(x), \tag{3.38}$$

where the set of coefficients $\{d_k^j\}_{k\in\mathbb{Z}}$, which we refer to as `differences', is computed via the inner product

$$d_k^j = \int_{-\infty}^{+\infty} f(x)\psi_{j;k}(x)\,dx. \tag{3.39}$$

The projection of a function on subspace W_j is denoted $(Q_j f)(x)$, where $Q_j = P_{j-1} - P_j$. Since we are considering a `periodized' MRA, on each subspace V_j and W_j the coefficients of the projections satisfy

$$\begin{aligned} s_k^j &= s_{k+2^{n-j}}^j, \\ d_k^j &= d_{k+2^{n-j}}^j, \end{aligned} \tag{3.40}$$

for each $j = 1, 2, \ldots, J$ and $k \in \mathbb{F}_{2^{n-j}} = \mathbb{Z}/2^{n-j}\mathbb{Z}$, i.e. $\mathbb{F}_{2^{n-j}}$ is the finite field of 2^{n-j} integers, e.g. the set $\{0, 1, \ldots, 2^{n-j} - 1\}$.

In our numerical algorithms, the expansion into the wavelet basis of $(P_0 f)(x)$ is given by a sum of successive projections on subspaces W_j, $j = 1, 2, \ldots, J$, and a final `coarse' scale projection on V_J,

$$(P_0 f)(x) = \sum_{j=1}^{J}\sum_{k\in\mathbb{F}_{2^{n-j}}} d_k^j\psi_{j;k}(x) + \sum_{k\in\mathbb{F}_{2^{n-J}}} s_k^J\,\phi_{J;k}(x). \tag{3.41}$$

Given the set of coefficients $\{s_k^0\}_{k\in\mathbb{F}_{2^n}}$, i.e. the coefficients of the projection of $f(x)$ on V_0, we use (3.27) and (3.28) to replace (3.37) and (3.39) by the following recursive definitions for s_k^j and d_k^j,

$$s_k^j = \sum_{l=1}^{L-1} h_l\, s_{l+2k+1}^{j-1}, \tag{3.42}$$

$$d_k^j = \sum_{l=1}^{L-1} g_l\, s_{l+2k+1}^{j-1}, \tag{3.43}$$

where $j = 1, 2, \ldots, J$ and $k \in \mathbb{F}_{2^{n-j}}$.

Given the coefficients $s^0 = P_0 f \in V_0$ consisting of $N = 2^n$ `samples' the decomposition of f into the wavelet basis is an order N procedure, i.e. computing the coefficients d_k^j and s_k^j recursively using (3.42) and (3.43) is an order N algorithm. Computing the J-scale decomposition of f via (3.42) and (3.43) by the pyramid scheme is illustrated in Figure 1. Figure 2

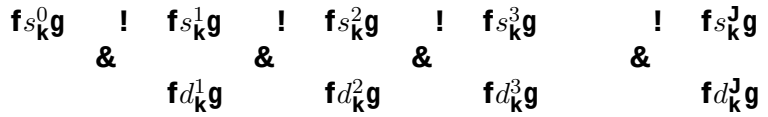


Figure 1: Projection of the coefficients $\{s_k^0\}$ into the multiresolution analysis via the pyramid scheme.

illustrates a typical wavelet representation of a function with $N = 2^n$, $n = 13$ and $J = 7$. We have generated this Figure using `coiflets', see e.g. [21], with $M = 6$ vanishing moments and an accuracy (cutoff) of $\varepsilon = 10^{-6}$, and note that a similar result is obtained for other choices of a wavelet basis. The top Figure is a graph of the projection of the function f on subspace V_0, which we note is a space of dimension 2^{13}. Each of the next $J = 7$ graphs represents the projection of f on subspaces W_j, for $j = 1, 2, \ldots 7$. Each W_j is a space of dimension 2^{13-j}, i.e. each consists of 2^{13-j} coefficients. Even though the width of the graphs is the same, we note that the number of degrees of freedom in W_j is twice the number of degrees of freedom in W_{j+1}. Since these graphs show coefficients d_k^j which are above the threshold of accuracy, ε, we note that the spaces W_1, W_2, W_3, and W_4 consist of no significant wavelet coefficients. This illustrates the `compression' property of the wavelet transform: regions where the function (or its

wavelet bases. First, we consider a two-dimensional wavelet basis which is arrived at by computing the tensor product of two one-dimensional wavelet basis functions, e.g.

$$\psi_{j;j'0;k;k'0}(x,y) = \psi_{j;k}(x)\psi_{j'0;k'0}(y), \tag{3.44}$$

where $j, j', k, k' \in \mathbb{Z}$. This choice of basis leads to the standard form (S-form) of an operator, [5, 8]. The projection of the operator T into the multiresolution analysis is represented in the S-form by the set of operators

$$T = \{A_j, \{B_j^{j'0}\}_{j'0 \geq j+1}, \{\Gamma_j^{j'0}\}_{j'0 \geq j+1}\}_{j \in \mathbb{Z}}, \tag{3.45}$$

where the operators $A_j, B_j^{j'0}$, and $\Gamma_j^{j'0}$ are projections of the operator T into the multiresolution analysis as follows

$$\begin{array}{rcl} A_j & = & Q_j T Q_j \quad : \quad W_j \to W_j, \\ B_j^{j'0} & = & Q_j T Q_{j'0} \quad : \quad W_{j'0} \to W_j, \\ \Gamma_j^{j'0} & = & Q_{j'0} T Q_j \quad : \quad W_j \to W_{j'0}, \end{array} \tag{3.46}$$

for $j = 1, 2, \ldots, n$ and $j' = j+1, \ldots, n$.

If n is the finite number of scales, as in (3.35), then (3.45) is restricted to the set of operators

$$T_0 = \{A_j, \{B_j^{j'0}\}_{j'0=j+1}, \{\Gamma_j^{j'0}\}_{j'0=j+1}, B_j^{n+1}, \Gamma_j^{n+1}, T_n\}_{j=1;\ldots;n}, \tag{3.47}$$

where T_0 is the projection of T on V_0. Here the operator T_n is the coarse scale projection of the operator T on V_n,

$$T_n = P_n T P_n : V_n \to V_n. \tag{3.48}$$

The subspaces V_j and W_j appearing in (3.46) and (3.48) can be periodized in the same fashion as described in Section 3.2.

The operators $A_j, B_j^{j'0}$, $\Gamma_j^{j'0}$, and T_n appearing in (3.45) and (3.47) are represented by matrices $\alpha^j, \beta^{j;j'0}$, $\gamma^{j;j'0}$ and s^n with entries defined by

$$\begin{array}{rcl} \alpha_{k;k'0}^j & = & \int\int \psi_{j;k}(x) K(x,y) \psi_{j;k'0}(y) dx dy, \\ \beta_{k;k'0}^{j;j'0} & = & \int\int \psi_{j;k}(x) K(x,y) \psi_{j'0;k'0}(y) dx dy, \\ \gamma_{k;k'0}^{j;j'0} & = & \int\int \psi_{j;k}(x) K(x,y) \psi_{j'0;k'0}(y) dx dy, \\ s_{k;k'0}^n & = & \int\int \phi_{n;k}(x) K(x,y) \phi_{n;k'0}(y) dx dy, \end{array} \tag{3.49}$$

where $K(x, y)$ is the kernel of the operator T. The operators in (3.47) are organized as blocks of a matrix as shown in Figure 3.3.

In [8] it is observed that if the operator T is

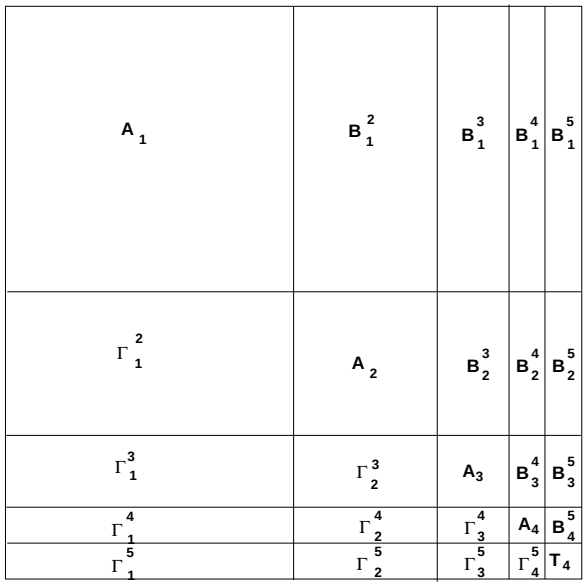


Figure 3: Organization of the standard form of a matrix.

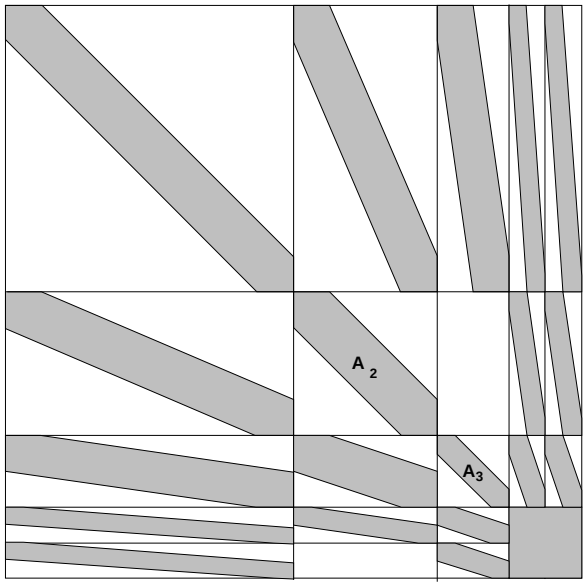
An alternative to forming two-dimensional wavelet basis functions using the tensor product (which led us to the S-form representation of operators) is to consider basis functions which are combinations of the wavelet, $\psi(\ )$, and the scaling function, $\phi(\ )$. We note that such an approach to forming basis elements in higher dimensions is speci c to wavelet bases (tensor products as considered above can be used with any basis, e.g. Fourier basis).

We will consider representations of operators in the non-standard form (NS-form), following [8] and [5]. Recall that the wavelet representation of an operator in the NS-form is arrived at using bases formed by combinations of wavelet and scaling functions, for example, in $L^2(\mathbb{R}^2)$

$$\begin{array}{l} \psi_{j;k}(x)\ \psi_{j;k'}(y), \\ \psi_{j;k}(x)\ \phi_{j;k'}(y), \\ \phi_{j;k}(x)\ \psi_{j;k'}(y), \end{array} \tag{3.53}$$

where $j, k, k' \in \mathbb{Z}$. The NS-form of an operator T is obtained by expanding T in the `telescopic' series

$$T = \sum_{j \in \mathbb{Z}} (Q_j T Q_j + Q_j T P_j + P_j T Q_j), \tag{3.54}$$


A 2
A3

where

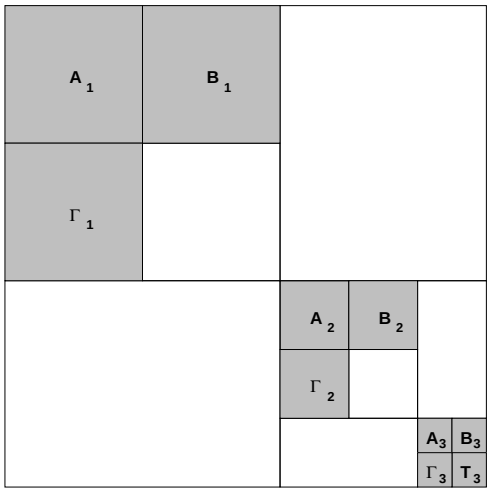


Figure 5: Organization of the non-standard form of a matrix. A_j, B_j, and Γ_j, $j = 1, 2, 3$, and T_3 are the only non-zero blocks.

using the decomposition algorithm described by (3.42) and (3.43) as follows. Given the coefficients $\{s^J_k\}_{k=1}$ and $\{d^j_k\}^J_{j=1}$, we decompose $\{s^1\}$ into $\{s^2\}$ and $\{d^2\}$ and form the sums $\{s^2\} = \{s^2_{\text{from}}\} + \{s^2\}$ and $\{d^2\} = \{d^2_{\text{from}}\} + \{d^2\}$. Then on each scale $j = 2, 3, \ldots, J - 1$, we decompose $\{s^j\} = \{s^j_{\text{from}}\} + \{s^j\}$ into $\{s^{j+1}\}$ and $\{d^{j+1}\}$ and form the sums $\{s^{j+1}\} = \{s^{j+1}_{\text{from}}\} + \{s^{j+1}\}$ and $\{d^{j+1}\} = \{d^{j+1}_{\text{from}}\} + \{d^{j+1}\}$. The sets $\{s^J\}$ and $\{d^j\}^J_{j=1}$ are the coefficients of the wavelet expansion of $(T_0 f_0)(x)$, i.e. the coefficients appearing in (3.63). This procedure is illustrated in Figure 7.

An alternative to projecting the representation (3.62) into the wavelet basis is to reconstruct (3.62) to space V_0, i.e. form the representation (3.36)

$$(P_0 f)(x) = \sum_{k \in \mathbb{Z}} s^0_k \phi_{0;k}(x), \tag{3.64}$$

using the reconstruction algorithm described in Section 3 as follows. Given the coefficients $\{s^J_k\}_{k=1}$ and $\{d^j_k\}^J_{j=1}$, we reconstruct $\{d^J_{\text{from}}\}$ and $\{s^J_{\text{from}}\}$ into $\{s^{J-1}\}$ and form the sum $\{s^{J-1}\} = \{s^{J-1}_{\text{from}}\} + \{s^{J-1}\}$. Then on each scale $j = J - 1, J - 2, \ldots, 1$ we reconstruct $\{s^j_{\text{from}}\}$ and $\{d^j_{\text{from}}\}$ into $\{s^{j-1}\}$ and form the sum $\{s^{j-1}\} = \{s^{j-1}_{\text{from}}\} + \{s^{j-1}\}$. The final reconstruction (of $\{d^1\}$ and $\{s^1\}$) forms the coefficients $\{s^0\}$ appearing in (3.64). This procedure is illustrated

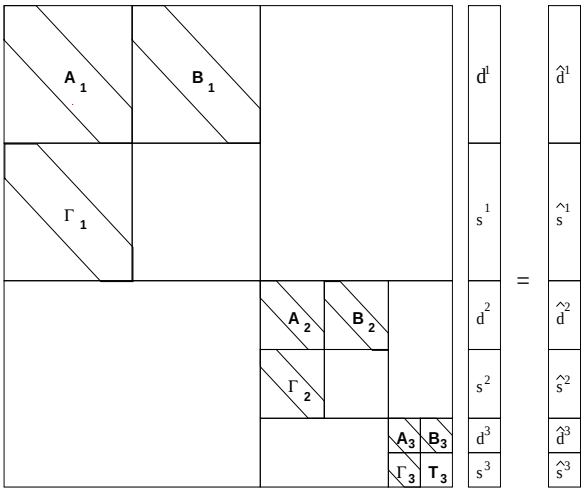


Figure 6: Illustration of the application of the non-standard form to a vector.

in Figure 8.

Following [5], in this Section we recall the wavelet representation of di erential operators $\frac{p}{x}$ in the NS-form. The rows of the NS-form of di erential operators may be viewed as nite-di erence approximations on subspace V_0 of order $2M$ 1, where M is the number of vanishing moments of the wavelet $\psi(x)$.

The NS-form of the operator $\frac{p}{x}$ consists of matrices $A^j, B^j,$ ^{j}, for $j = 0, 1, \ldots, J$ and a `coarse scale' approximation T^J. We denote the elements

$$\begin{array}{ccccc} \{s^0\} & ! & \{A^1 + s^1\} = \{s^1\} & ! & \\ & \& & & \& & \\ & & \{B^1 + d^1\} = \{d^1\} & & \end{array} \qquad \begin{array}{ccc} ! & \{A^J + s^J\} = \{s^J\} \\ \& & \\ & \{B^J + d^J\} = \{d^J\} \end{array}$$

Figure 7: Projection of the product of the NS-form and a function into a wavelet basis.

where a_{2k-1} are the autocorrelation coefficients of H defined by

$$a_n = 2 \sum_{i=0}^{L_f - 1 - n} h_i h_{i+n}, \quad n = 1, \ldots, L_f - 1. \tag{3.69}$$

We note that the autocorrelation coefficients a_n with even indices are zero,

$$a_{2k} = 0, \quad k = 1, \ldots, L_f/2 - 1, \tag{3.70}$$

and $a_0 = \sqrt{2}$. The resulting coefficients s_l^0 corresponding to the projection of the operator $\frac{d}{dx}$ on V_0 may be viewed as a finite-difference approximation of order $2M - 1$. Further details are found in [5].

We are interested in developing adaptive algorithms, i.e. algorithms such that the number of operations performed is proportional to the number of significant coefficients in the wavelet expansion of solutions of partial differential equations. The S-form has `built-in' adaptivity, i.e. applying the S-form of an operator to the wavelet expansion of a function, (3.38), is a matter of multiplying a sparse vector by a sparse matrix. On the other hand, as we have mentioned before, the S-form is not a very efficient representation (see, e.g., our discussion of convolution operators in Section 3.3).

In the following Sections we address the issue of adaptively multiplying the NS-form and a vector. Since the NS-form of a convolution operator remains a convolution, the A^j, B^j, and Γ^j blocks may be thought of as being represented by short filters. For example, the NS-form of a differential operator in any dimension requires $O(C)$ coefficients as it would for any finite difference scheme. We can exploit the efficient representation afforded us by the NS-form and use the vanishing-moment property of the B^j and Γ^j blocks of the NS-form of differential operators and the Hilbert transform to develop an adaptive algorithm. In Section 4.1 we describe two methods for constructing the NS-form representation of operator functions. In Section 4.2 we establish the vanishing-moment property which we later use to develop an adaptive algorithm for multiplying operators and functions expanded in a wavelet basis. Finally, in Section 4.3 we present an algorithm for adaptively multiplying the NS-form representation of an operator and a function expanded in the wavelet system of coordinates.

Non Standard For tor Functions Representation of Opera

In this

for example. In the following we assume that the function f is

where

$$g(\ ) = \sum_{k\in\mathbb{Z}} f(\ i2^{-j}(\ + 2\pi k))|\hat{\phi}(\ + 2\pi k)|^2. \tag{4.82}$$

We now observe that for a given accuracy the function $|\hat{\phi}(\ )|^2$ acts as a cuto function in the Fourier domain, i.e. $|\hat{\phi}(\ )|^2 <$ for $|\ | >$ for some > 0. Therefore, Equation (4.80) is approximated to within by

$$g(\ ) = \sum_{k=-K}^{K} f(\ i2^{-j}(\ + 2\pi k))|\hat{\phi}(\ + 2\pi k)|^2, \tag{4.83}$$

for some K. Using (4.83) (in place of $g(\ )$) in (4.81) we obtain an approximation to the coe cients s_l^j,

$$s_l^j = \frac{1}{N} \sum_{n=0}^{N-1} g(\ n) e^{i \cdot} \tag{4.84}$$

The coe cients s_l^j are computed by applying the FFT to the sequence $fg(\ n)g$ computed via (4.83).

In order to compute the NS-form of an operator function via (4.72), we

We now establish the vanishing-moment property of the B-blocks of the NS-form representation of functions of a di erential operator described in Section 4.1 and the Hilbert transform. We note that a similar result also holds for the B-blocks of some classes of pseudo-di erential operators, see e.g. [31]. Additionally, we note that these results do not require compactly supported wavelets and we prove the results for the general case. In Section 4.3 we use the vanishing-moment property to design an adaptive algorithm for multiplying the NS-form of an operator and the wavelet expansion of a function.

Proposition 1. If the wavelet basis has M vanishing moments, then the B-blocks of the NS-form of the analytic operator function $f(x)$, described in Section 4.1, satisfy

$$\sum_{l=-\infty}^{+\infty} l^m \beta_l^j = 0, \tag{4.88}$$

for $m = 0, 1, 2, \ldots, M-1$ and $j = 1, 2, \ldots J$.

Proof. Using the de nition (3.49), we obtain

$$\sum_{l=-\infty}^{+\infty} l^m \beta_l = \int_{-\infty}^{+\infty} \psi(x-k) f(x) P_m(x) dx. \tag{4.89}$$

We have used the fact that $\int_{-\infty}^{+\infty} \psi(x-k) P_m$ and m 1/R67 10.90990Tf 4.1898 0 014913 [illegible] /R37 190 Td (-fo Tf 5.84741 -1.68 Td 9d ( ) 10.

We have used the fa[illegible]/R67 4291 0liqtt65j -3524451 -16.08 Td (+)Tj /R89 7.97011 Tf 6.59924 0

(where p.v. indicates the principle value), satisfy

$$\sum_{l=-\infty}^{+\infty} l^m \beta_l^j = 0, \tag{4.93}$$

for $0 \leq m \leq M-1$ and $j = 1, 2, \ldots J$.

Proof. The β_l elements of the NS-form of the Hilbert transform are given by

$$\beta_l = \int_{-\infty}^{+\infty} \psi(x-l)(H_{\mathrm{scf}})(x)dx, \tag{4.94}$$

and proceeding as in Proposition 1, we find

$$\begin{aligned}
\sum_{l=-\infty}^{+\infty} l^m \beta_l &= \sum_{l=-\infty}^{+\infty} l^m \int_{-\infty}^{+\infty} \psi(x-l)(H_{\mathrm{scf}})(x)dx \\
&= \sum_{l=-\infty}^{+\infty} l^m \int_{-\infty}^{+\infty} (H\psi)(x)_{\mathrm{scf}}(x+l)dx \\
&= \int_{-\infty}^{+\infty} (H\psi)(x) P_m(x)dx,
\end{aligned} \tag{4.95}$$

where, once again, we have used (4.90).

To show that the integrals in (4.95) are zero, we establish that $(H\psi)(x)$ has at least M vanishing moments. Let us consider the generalized function

$$\int_{-\infty}^{\infty} (H\psi)(x) x^m e^{i\omega x} d\omega = i^{-m} \frac{d^m}{d\omega^m} (H\psi)(\omega). \tag{4.96}$$

In the Fourier domain the Hilbert transform of the function g defined by

$$(H\widehat{g})(\omega) = -i\,\mathrm{sign}(\omega)\hat{g}(\omega), \tag{4.97}$$

may be viewed as a generalized function, derivatives of which act on test functions $f \in C_0^\infty(\mathbb{R})$ as

$$\begin{aligned}
< \frac{d^m}{d\omega^m} (-i\,\mathrm{sign}(\omega)\hat{g}(\omega)), f > &= \sum_{j=1}^{m} i \binom{m}{l} f^{(j-1)}(0)\hat{g}^{(m-j)}(0) + \\
&\quad \int_{-\infty}^{\infty} i\,\mathrm{sign}(\omega)\,\hat{g}^{(m)}(\omega) f(\omega) d\omega .
\end{aligned} \tag{4.98}$$

In order

where $\hat{\psi}$( ) is the Fourier transform of $\psi(x)$. Setting $\hat{g}$( ) = $\hat{\psi}$( ) in (4.98), the sum on the right hand side of (4.98) is zero. We also observe that the integrand on the right hand side of (4.98), i.e. sign( )$\hat{\psi}^{m)}$( )$\hat{f}$( ), is continuous at = 0, once again because $\psi(x)$ has M vanishing moments. We can then de ne functions $\hat{W}^{m)}$( ) for $m = 0, 1, \ldots, M$ 1, as

$$\hat{W}^{m)}(\ ) = \begin{cases} i\,\hat{\psi}^{m)}(\ ), & > 0; \\ 0, & = 0; \\ i\,\hat{\psi}^{m)}(\ ), & < 0, \end{cases} \quad (4.100)$$

such that $\hat{W}^{m)}$( ) coincides with the m-th derivative of the generalized function (4.97) on the test functions f 2 $C_0^\infty(\mathbb{R})$. Since $\hat{W}^{m)}$( ) are continuous functions for $m = 0, 1, \ldots, M$ 1, we obtain instead of (4.96)

$$\int_{-\infty}^{\infty} (\mathbf{H}\psi)(x)\, x^m e^{i}$$

11 Tf 64s4.167 10.909c..28877.9091 Tf 8.1606T 8.37817 0 Td (x)Tj 40.80 Td (of)T=8.0872 9 0 (that)Tj 26.6836 -th derivative of the generalized func-

tion

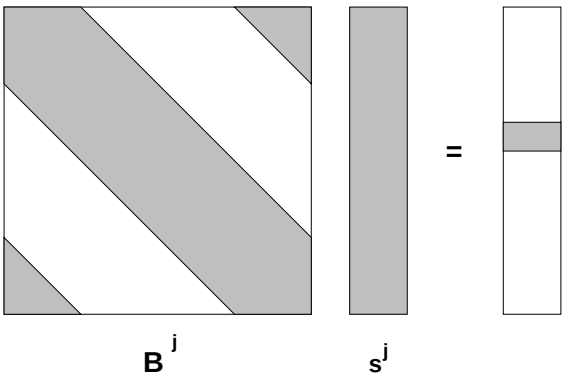


Figure 9: For the operators considered in Section 4.2 the vanishing-moment property of the rows of the B-block yields a sparse result (up to a given accuracy) when applied to a smooth and dense vector $\{s^j\}$.

for $j = 1, 2, \ldots, J - 1$ and $k \in \mathbb{F}_{2^{n-j}} = \{0, 1, 2, \ldots, 2^{n-J} - 1\}$ and on the the nal, coarse scale,

$$\hat{d}_k^J = \sum_l A_{k+l}^J d_{k+l}^J + \sum_l B_{k+l}^J s_{k+l}^J, \tag{4.104}$$

$$\hat{s}_k^J = \sum_l {}_{k+l}^J d_{k+l}^J + \sum_l T_{k+l}^J s_{k+l}^J, \tag{4.105}$$

for $k \in \mathbb{F}_{2^{n-j}}$. The di culty in adaptively applying the NS-form of an operator to such functions is the need to apply the B-blocks of the operator to the averages $\{s^j\}$ in (4.102). Since the averages are \smoothed" versions of the function itself, these vectors are not necessarily sparse and may consist of 2^{n-j} signi cant coe cients on scale j. Our algorithm uses the fact that for the operator functions considered in Section 4.1, the rows of the B-blocks have M vanishing moments. This means that when the row of a B-block is applied to the \smooth" averages $\{s^j\}$ the resulting vector is sparse (for a given accuracy), as is illustrated in Figure 9.

Since each row of the B-block has the same number of vanishing moments as the lter G, we can use the $\{d^j\}$ coe cients of the wavelet expansion to predict signi cant contributions to (4.102). In this way we can replace the calculations with a dense vector $\{s\}$ in (4.102) by calculations with a sparse

vector $\{s^j_k\}$,

$$d^j_k = \sum_l A^j_{k+l} d^j_{k+l} + \sum_l B^j_{k+l} s^j_{k+l}, \tag{4.106}$$

for $j = 1, 2, \ldots, J-1$ and $k \in \mathbb{F}_{2^{n-j}}$. In what follows we describe a method for determining the indices of $\{s^j_k\}$ using the indices of the significant wavelet coefficients $\{d^j_k\}$.

The formal description of the procedure is as follows. For the functions under consideration the magnitude of many wavelet coefficients $\{d^j_k\}$ are below a given threshold of accuracy ε. The representation of f on V_0, (3.41), using only coefficients above the threshold is

$$(P_0 f)(x) = \sum_{j=1}^{J} \sum_{\{k : |d^j_k| > \varepsilon\}} d^j_k \psi_{j;k}(x) + \sum_{k \in \mathbb{F}_{2^{n-J}}} s^J_{k,\varepsilon} \phi_{J;k}(x), \tag{4.107}$$

whereas for the error we have

$$\|(P_0 f)(x) - (P_0 f)(x)\|_2 = \left(\sum_{j=1}^{J} \sum_{\{k : |d^j_k| \le \varepsilon\}} |d^j_k|^2 \right)^{1/2} < N_r^{1/2} \varepsilon, \tag{4.108}$$

where N_r is the number of coefficients below the threshold. The number of significant wavelet coefficients is defined as $N_s = N - N_r$, where N is the dimension of the space V_0.

We define the ε-accurate subspace for f, denoted $D_f \subset V_0$, as the subspace spanned by only those basis functions present in (4.107),

$$D_f = V_J \bigcup \{\text{span } \{\psi_{j;k}(x)\} : |d^j_k| > \varepsilon\}, \tag{4.109}$$

for $1 \le j \le J$ and $k \in \mathbb{F}_{2^{n-j}}$. Associated with D_f are subspaces $S_{f;j}$

In this way we can use D_f to `mask' V_0 forming $S_{f;j}$; in practice all we do is manipulate indices. The subset of coefficients $\{s^j\}$ that contribute to the sum (4.106) may now be identified by indices of the coefficients corresponding to basis functions in $S_{f;j}$.

We now show that significant wavelet coefficients d^{j+1} and contributions of $B^j s^j$ to (4.102) both originate from the same coefficients s^j. In this way we can use the indices of d^{j+1} to identify the coefficients s^j that contribute to the sum (4.106). We begin by expanding $f(x + 2^j l)$ into its Taylor series,

Using the vanishing moments of the filter $G = \{g_l\}$, we obtain

$$d_{k'}^{j+1} = \frac{2^{-j/2}}{M!} \sum_l g_l \int_{-\infty}^{\infty} \phi(x) f^{(M)}(z)(z - 2^j(x + 2k'))^M dx, \qquad (4.117)$$

for $k' \in \mathbb{Z}_{2^{J-(j+1)}}$.

To show that $|d_{k'}^{j+1}| < \varepsilon$ implies $|\bar{d}_k^j| < C\varepsilon$, we consider two cases. First, if $|d_{k'}^{j+1}| < \varepsilon$ and k is even, i.e. $k = 2n$ for $n \in \mathbb{Z}_{2^{J-(j+1)}}$, then we see that $\bar{d}_{2n}^j$ and $d_{k'}^{j+1}$ given by (4.117) only differ in the coefficients g_l and β_{2n+l}^j. Since g_l and β_{2n+l}^j are of the same order, the differences satisfy $|\bar{d}_{2n}^j| < C\varepsilon$ for some constant C. On the other hand, if $k = 2n + 1$ for $n = 2 \in \mathbb{Z}_{2^{J-(j+1)}}$, we find

$$\bar{d}_{2n+1}^j = \frac{2^{-j/2}}{M!} \sum_{l=-L}^{L} \beta_{2n+1+l}^j \int_{-\infty}^{\infty} \phi(x+1) f^{(M)}(z)(z - 2^j(x + 2n))^M dx, \qquad (4.118)$$

which again is of the same order as $d_{k'}^{j+1}$. Therefore, if $|d_{k'}^{j+1}| < \varepsilon$ for $k' \in \mathbb{Z}_{2^{J-(j+1)}}$, then for some constant C, $|\bar{d}_k^j| < C\varepsilon$, for $k \in \mathbb{Z}_{2^{J-j}}$.

Evaluating Functions in Wavelet Bases

In this Section we describe our adaptive algorithm for evaluating the pointwise product of functions represented in wavelet bases. More generally, our results may be applied to computing functions $f(u)$, where f is an analytic function and u is expanded in a wavelet basis. We note that since pointwise multiplication is a diagonal operator in the `physical' domain, computing the pointwise product in any other domain appears [illegible] coefficients. [illegible]

For example, if $u(x)$ is expanded in its Fourier series, clearly the Fourier coe cients of the function $f(u)$ do not correspond to the function of the Fourier coe cients. This has led to the development of pseudo-spectral algorithms for numerically solving partial di erential equations, see e.g. [23, 24].

In order to explain the algorithm for computing $f(u)$ in the wav

A n u^2

Since the product of two functions can be expressed as a difference of squares, it is sufficient to explain an algorithm for evaluating u^2. The algorithm we describe is an improvement over that found in [6, 7].

In order to compute u^2 in a wavelet basis, we first recall that the projections of u on subspaces V_j and W_j are given by $P_j u \in V_j$ and $Q_j u \in W_j$ for $j = 0, 1, 2, \ldots, J - n$, respectively (see the discussion in Section 3). Let j_f, $1 \le j_f \le J$ (see, e.g., Figure 10 where $j_f = 5$ and $J = 8$), be the finest scale having significant wavelet coefficients that contribute to the ε-accurate approximation of u, i.e. the projection of u can be expressed as

$$(P_0 u)(x) = \sum_{j=j_f}^{J} \sum_{\{k:|d_k^j|>\varepsilon\}} d_k^j \psi_{j,k}(x) + \sum_{k \in \mathbb{F}_{n-J}} s_{k,J}^J \phi_{J,k}(x). \qquad (5.125)$$

Let us first consider the case where u and $u^2 \in V_0$, so that we can expand $(P_0 u)^2$ in a `telescopic'

($P_j u$)($Q_j u$) do not necessarily belong to the same subspace as the multiplicands. However, since

$$V_j \overset{M}{} W_j = V_{j-1} \quad V_{j-2} \quad \dots \quad V_{j-j} \quad \dots, \tag{5.129}$$

we may think of both $P_j u$ 2 V_j and $Q_j u$ 2 W_j as elements of a ner subspace, that we denote V_{j-j} , for some j_0 1. We compute the coe cients of $P_j u$ and

for $m = 1, 2, \ldots, M$ and arrive at

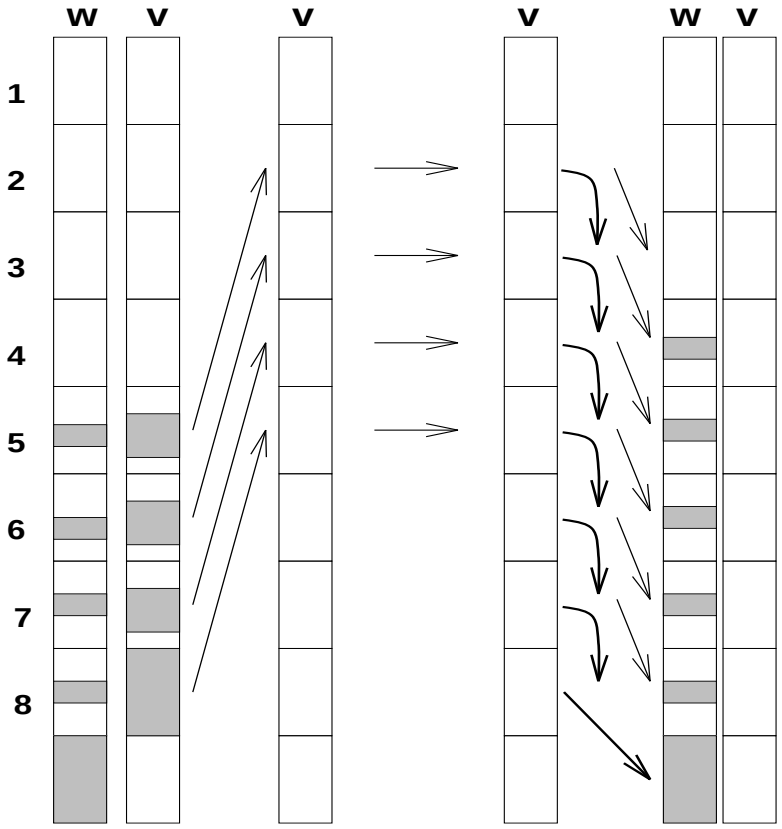
$$(\quad i)^{-m} \qquad ^{m}$$

where $P_j f(u)$ is the contribution to $f(u)$ on subspace V_j (see (5.127). On the final coarse scale J, we compute

$$P_{J-j}(u^2) = (R_j^j(P_J u))^2 + 2(R_j^j(P_J u))(R_j^j(Q_J u)) + (R_j^j(Q_J u))^2. \quad (5.141)$$

We then project the representation on subspaces V_{j-j}, for $j = j_f, \ldots J$ into the wavelet basis. This procedure is completely equivalent to the decomposition one has to perform after applying the NS-form. The algorithm for computing the projection of u^2 in a wavelet basis is illustrated in Figure 10. In analogy with \pseudo-spectral" schemes, as in e.g. [23, 24], we refer to this as an *adaptive pseudo-wavelet algorithm*.

To demonstrate that the algorithm is adaptive, we recall that u has a sparse representation in the wavelet basis. Thus, evaluating $(Q_j u)^2$ for $j = 1, 2,$


w
v
v
v
w
v
1
2
3
4
5
6
7
8

5.5 Evaluation of f(u)

This Section consists of a number of observations regarding the evaluation of functions other than $f(u) = u^2$ in wavelet bases. For analytic $f(u)$ we can apply the same approach as in Section 5.1, wherein we assume $f(P_0 u) \in V_0$ and expand the projection $f(P_0 u)$ in the `telescopic' series

$$f(P_0 u) \quad f(P_J u) = \sum_{j=1}^{J} f(P_{j-1} u) \quad f(P_j u). \tag{5.142}$$

Using $P_{j-1} = Q_j + P_j$ to decouple scale interactions in (5.142) and assuming $f(\,)$ to be analytic, we substitute the Taylor series

$$f(Q_j u + P_j u) = \sum_{n=0}^{N} \frac{f^{(n)}(P_j u)}{n!} (Q_j u)^n + E_{j;N}(f, u), \tag{5.143}$$

to arrive at

$$f(P_0 u) = f(P_J u) + \sum_{j=1}^{J} \sum_{n=1}^{N} \frac{f^{(n)}(P_j u)}{n!} (Q_j u)^n + E_{j;N}(f, u). \tag{5.144}$$

For $f(u) = u^2$, $j_f = 1$ and $N = 2$ we note that (5.144) and (5.127) are identical.

This approach can be used for functions $f(u)$ that have rapidly converging Taylor series expansions, e.g. $f(u) = \sin(u)$, for $|u|$ sufficiently small. In this case, for a given accuracy we fix an N so that $|E_{j;N}(f, u)| < \varepsilon$. We for

In

begin by setting

$$U_0(t_{j+1}) = E(U(t_j)) + I(U(t_j), U(t_j)), \tag{6.151}$$

and repeatedly evaluate

$$U_{k+1}(t_{j+1}) = E(U(t_j)) + I(U(t_j), U_k(t_{j+1})), \tag{6.152}$$

for $k = 0, 1, 2 \ldots$. We terminate the iteration when

$$\| U_{k+1}(t_{j+1}) - U_k(t_{j+1}) \| < \varepsilon, \tag{6.153}$$

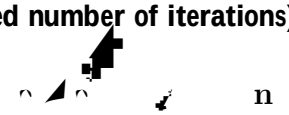
where

$$\| U_{k+1}(t_{j+1}) - U_k(t_{j+1}) \| = 2^{-n} \sum_{i=1}^{2^n} (U_{k+1}(x_i, t_{j+1}) - U_k(x_i, t_{j+1}))^2 \Bigg)^{1/2}. \tag{6.154}$$

Once (6.153) is satisfied, we update the solution and set

$$U(t_{j+1}) = U_{k+1}(t_{j+1}). \tag{6.155}$$

Again we note that one can use a more sophisticated iterative scheme and different stopping criteria for evaluating (6.150) (e.g. simply compute (6.152) for a fixed number of iterations).



We begin with this simple linear example in order to illustrate several points and provide a bridge to the nonlinear problems discussed below. In particular we show that in the wavelet system of coordinates, higher order schemes do not necessarily require more operations than lower order schemes. We consider the heat equation on the unit interval,

$$u_t = \nu u_{xx}, \qquad 0 \le x \le 1, \quad 0 \le t \le 1, \tag{6.156}$$

for $\nu > 0$, with the initial condition

$$u(x, 0) = u_0(x), \quad 0 \le x \le 1, \tag{6.157}$$

and the periodic boundary condition $u(0, t) = u(1, t)$. There are several well-known approaches for solving (6.156) and more general equations of this type having variable coefficients. Equation (6.156) can be viewed as a

simple representative of this class of equations and we emphasize that the following remarks are applicable to the variable coe cient case, $\nu = \nu(x)$ (see also [32]).

For di usion-type equations, explicit nite di erence schemes are conditionally stable with the stability condition $\nu \ t/(\ x)^2 < 1$ (see e.g. [19]) where $t = 1/N_t$, $x = 1/N$, and N_t is the number of time steps. This condition tends to require prohibitively small time steps. An alternate, implicit approach is the Crank-Nicolson scheme, [19], which is unconditionally stable and accurate to $O((\ t)^2 + (\ x)^2)$. At each time step, the Crank-Nicolson scheme requires solving a system of equations,

$$AU(t_{j+1}) = BU(t_j), \tag{6.158}$$

for $j = 0, 1, 2, \ldots, N_t \ 1$, where we have suppressed the dependence of $U(x,t)$ on x. The matrices A and B are given by $A = \text{diag}(\ _{\overline{2}}, 1 + \ , \ _{\overline{2}})$ and $B = \text{diag}(_{\overline{2}}, 1 \ \ , _{\overline{2}})$, where $= \nu \frac{\ t}{\ x)}$.

Alternatively, we can write the solution of (6.156) as

$$u(x,t) = e^{t\mathcal{L}} u_0(x), \tag{6.159}$$

where $L = \nu \ _{xx}$, and compute (6.159) by discretizing the time interval $[0,1]$ into N_t subintervals of length $t = 1/N_t$, and by repeatedly applying the NS-form of the operator $e^{\ t\mathcal{L}}$ via

$$U(t_{j+1}) = e^{\ t\mathcal{L}} U(t_j), \tag{6.160}$$

for $j = 0, 1, 2, \ldots, N_t \ 1$, where $U(t_0) = U(0)$. The numerical method described by (6.160) is explicit and unconditionally stable since the eigenvalues of $e^{\ t@_x}$ are less than one.

The fact that the Crank-Nicolson scheme is unconditionally stable allows one to choose t independently of x; in particular one can choose t to be proportional to x. In order to emphasize our point we set $x = t$ and $\nu = 1$. Although the Crank-Nicolson scheme is second order accurate and such choices of the parameters x, t, and ν appear to be reasonable, by analyzing the scheme in the Fourier domain, we nd that high frequency components in an initial condition decay very slowly. By diagonalizing matrices A and B in (6.158), it is easy to nd the largest eigenvalue of $A^{-1}B$, $_N = \frac{1-2}{1+2}$. For the choice of parameters $\nu = 1$ and $t = x$, we see that as becomes large, the eigenvalue $_N$ tends to 1. We note that there are various ad hoc remedies (e.g. smoothing) used in conjunction with

the Crank-Nicolson scheme to remove these slowly decaying high frequency components.

For example, let us consider the following initial condition

u on

$e^{-t\partial_{xx}}$, the adaptive algorithm developed in Section 4.3 and the sparsity of the solution in the wavelet basis. Finally, we note that if we were to consider (6.156) with variable coefficients, e.g.

$$u_t = \nu(x)u_{xx}, \tag{6.162}$$

the exponential operator $e^{-t\nu(x)\mathcal{L}}$ can be computed in $O(N)$ operations using the scaling and squaring method outlined in e.g. [9] (see also [12]).

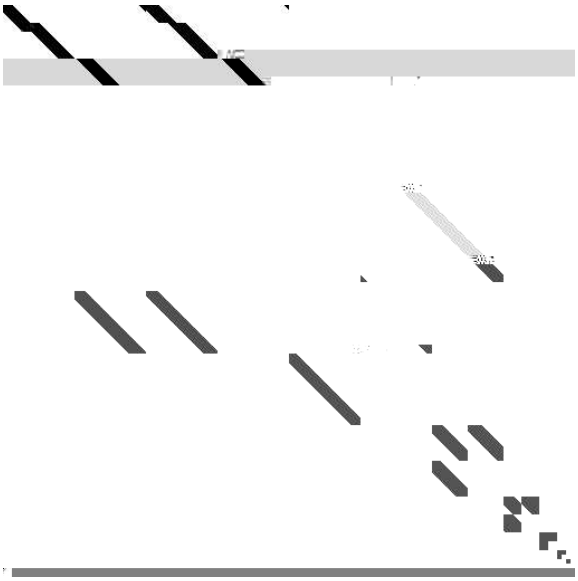


Figure 13: NS**-form representation of the operator** $A^{-1}B$ **used in the Crank-Nicolson scheme (6.158). Entries of absolute value greater than** 10^{-8} **are shown in black. The wavelet basis is Daubechies with** M **= 6 vanishing moments (**$L_{\mathbf{f}}$ **= 18), the number of scales is** n **= 9 and** J **= 7. We have set** ν **= 1.0 and** $t =$ $x = 2^{-9}$**. Note that the top left portion of the Figure contains non-zero entries which indicate high frequency components present in the operator** $A^{-1}B$**.**

Figure 14: NS**-form representation of the operator** $e^{\overline{\mathbf{t}\mathcal{L}}}$ **used in (6.160). Entries of absolute value greater than** 10^{-8} **are shown in black. The wavelet basis is Daubechies with** M **= 6 vanishing moments (**$L_{\mathbf{f}}$ **= 18), the number of scales is** n **= 9 and** J **=**

Our next example is the numerical calculation of solutions of Burgers' equation

$$u_t + uu_x = \nu u_{xx}, \qquad 0 \quad x \quad 1, \quad t \quad 0, \tag{6.163}$$

for $\nu > 0$, together with an initial condition,

$$u(x, 0) = u_0(x), \qquad 0 \quad x \quad 1, \tag{6.164}$$

and periodic boundary conditions $u(0, t) = u(1, t)$. Burgers' equation is the simplest example of a nonlinear partial differential equation incorporating both linear diffusion and nonlinear advection. Solutions of Burgers' equation consist of stationary or moving shocks and capturing such behavior is an important simple test of a new numerical method, see e.g. [34, 29, 4].

Burgers' equation may be solved analytically by the Cole-Hopf transformation [27, 17], wherein it is observed that a solution of (6.163) may be expressed as

$$u(x, t) = 2\nu \frac{\phi_x}{\phi}, \tag{6.165}$$

where $\phi = \phi(x, t)$ is a solution of the heat equation with initial condition

$$\phi(x, 0) = e^{-\frac{1}{4\nu}\int_0^x u(x;0)dx}. \tag{6.166}$$

Remark: We note that if

in an essentially exact way. Thus, we may attribute all numerical artifacts in the solution to the nonlinear advection term in (6.163).

For each of the following examples, we illustrate the accuracy of our approach by comparing the approximate solution U_w with the exact solution U_e using

$$\|U_w - U_e\|_2 = 2^{-n} \sum_{i=0}^{2^n - 1} (U_w(x_i, t) - U_e(x_i, t))^2 \Bigg)^{1/2}. \tag{6.168}$$

For comparison purposes, we compute the exact solution U_e via

$$U_e(x, t) = \frac{\int_{-\infty}^{\infty} \frac{x-\eta}{t} e^{-G(\eta;x,t)/2} \, d\eta}{\int_{-\infty}^{\infty} e^{-G(\eta;x,t)/2} \, d\eta}, \tag{6.169}$$

where

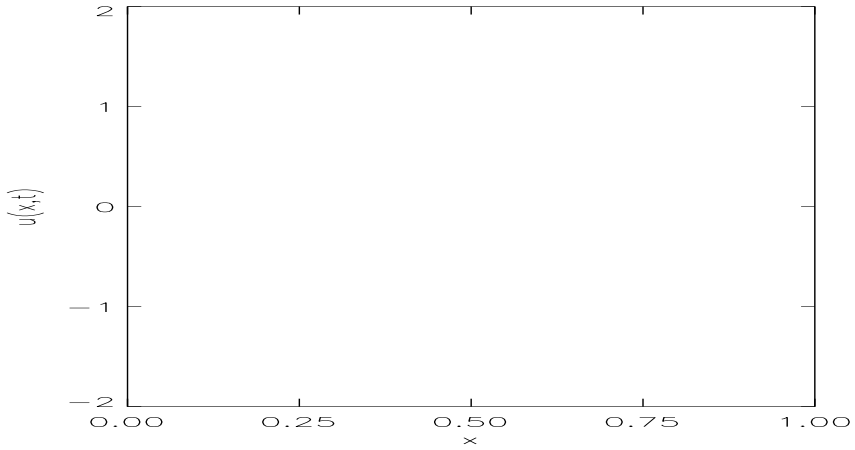
$$G(\eta; x, t) = \int_0^{\eta} F(\eta') d\eta' + \frac{(x - \eta)^2}{2t}, \tag{6.170}$$

and $F(\eta) = u - u$

0.0

ν = 0.001, and = 10^{-6}, and we refer to Figures 17 and 18. Using n = 10 scales to represent the solution in the wavelet basis is insu cient to represent the high frequency components present in the solution. Figure 17 illustrates the projection of the solution on V_0 beyond the poin

O.


2
1
0
−1
−2
u(x,t)
0.00
0.25
0.50
0.75
1.00
x

In this Section we consider the numerical solution of the generalized Burgers' equation

$$u_t + u\ u_x + \quad u \quad = \nu u_{xx}, \qquad 0 \quad x$$

growth of the solution, depending on the size of the coe cient ν. We have increased the di usion coe cient to ν = 0.005, and Figure 23 illustrates the evolution of the projection of the solution and Figure 24 illustrates the number of signi cant wavelet coe cients. We point out that the number of operations required to update the solution is proportional to the number of signi cant coe cients.

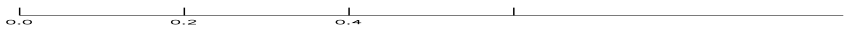
Example 5. As a nal example, we compute approximations to the solution of the so-called cubic Burgers' equation

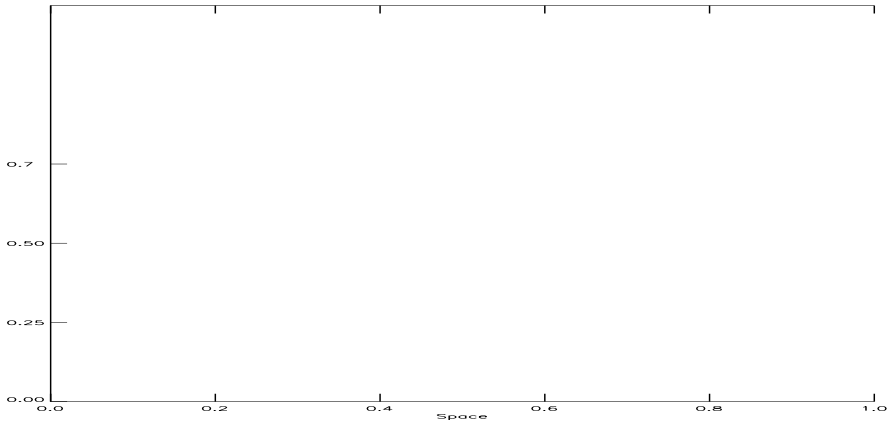
$$u_t + u^2 u_x = \nu u_{xx}, \qquad 0 \quad x \quad 1, \quad t \quad 0, \tag{6.176}$$

via

$$U(t_{i+1}) = e^{-t\,@_x} U(t_i) \quad \frac{1}{2} O_{@_x;1} \left[U^2(t_i)\ {}_x U(t_{i+1}) + U^2(t_{i+1})\ {}_x U(t_i) \right], \tag{6.177}$$

where $O_{@_x;1}$ is given by (2.19). The only di erence in (6.177), as compared with the approximation to Burgers' equations, (6.167), is the presence of the cubic nonlinearity. We have computed approximations to the solution using our algorithms with n = 13, J = 6, t = 0.001, ν = 0.001, and = 10^{-6}. Figures 25 and 26 illustrate the evolution of the solution for a gaussian initial condition, and Figures 27 and 28 illustrate the evolution of the solution for a sinusoidal initial condition. The gaussian initial condition evolves to a moving shock, and the sinusoidal initial condition evolves into two right-moving shocks. We note that although the number of grid points in a uniform discretization of such an initial value problem is, in this case, $N = 2^{13}$, we are using only a few hundred signi cant wavelet coe cients to update the solution.


0.0
0.2
0.4


0.7
0.50
0.25
0.00
0.0
0.2
0.4
0.6
0.8
1.0
Space


0.


0.

Conclusions

In this Chapter we have synthesized the elements of

using a wavelet (or multi-wavelet) basis on an interval rather than a periodized wavelet basis. Also, we note that variable coe cients in

[13] **Bony, J.M., Calcul symbolique et propagation des singularites pour les equations aux derivees partielles non-lineaires.** *Ann c en E N* **14 (1981), 209-246.**

[14] **Bramble, J., Pasciak, J. and Xu, J., Parallel multilevel preconditioning,** *M Co p* **55 (1990), 1-22.**

[15] **Burgers, J.M., A mathematical model illustrating the theory of turbulence,** *Ad App Mec* **1 (1948), 171.**

[16] **Chui, C. K.,** *An n rod c on o e e* **, Academic Press, Boston, MA, 1992.**

[17] **Cole, J.D., On a quasilinear parabolic equation occurring in aerodynamics,** *r App M* **9 (1951), 225.**

[18] **Constantin, P., Lax, P.D., Majda, A., A Simple One-dimensional Model for the Three-dimensional Vorticity Equation,** *Co P re nd App M* **38 (1985), 715.**

[19] **Dahlquist, G. and Bjerck, A.,** *N er c Me od* **, Prentice-Hall, Englewood Cli s, NJ, 1974.**

[20] **Dahmen, W. and Kunoth, A., Multilevel Preconditioning.** *N er M* **63 (1992), 315.**

[21] **Daubechies, I., Orthonormal bases of compactly supported wavelets,** *Co P re nd App M* **41 (1988), 909-996.**

[22] **Daubechies,I.,** *en Lec re on e e* **, CBMS-NSF Series in Applied Mathematics, SIAM Philadelphia, Penn. 1992.**

[23] **Fornberg, B., On a Fourier Method for the Integration of Hyperbolic Equations.** *AM N er An* **12 (1975), 509.**

[24] **Fornberg, B. and Whitham, G.B., A numerical and theoretical study of certain nonlinear wave phenomena,** *P r n oc Lond* **289 (1978), 373.**

[25] **Gagnon, L. and Lina, J.M., Wavelets and Numerical Split-Step Method: A Global Adaptive Scheme. Preprint To appear in** *Jour. Opt. Soc. Amer. B*.

[26] **Haar, A., Zur Theorie der orthogonalen Funktionensysteme.** *M e c e Ann en* **(1910), 331-371.**

[27] **Hopf, E., The Partial Di erential Equation** $u_t + uu_x = \mu u_{xx}$**.** *Co on P re nd App M* **3 (1950), 201.**

[28] **Ja ard, S., Wavelet methods for fast resolution of elliptic problems.** *AM N An* **29 (1992), 965-986.**

[29] **Liandrat, J., Perrier, V. and Tchamitchian, P., Numerical Resolution of Nonlinear Partial Di erential Equations using the Wavelet Approach, in** *e e nd e r App c on* **, Eds: M.B. Ruskai, G. Beylkin, R. Coifman, I. Daubechies, S. Mallat, Y. Meyer and L. Raphael. Jones and Bartlett Publishing, Inc., 1992, 227-238.**

[30] **Meyer, Y.,** *e e nd oper or* **. Cambridge Studies in Advanced Mathematics, Cambridge University Press 1992.**

[31] **Meyer, Y., Le Calcul Scienti que, les Ondelettes et les Filtres Miroirs en Quadrature.** *Cen re de ec erc e de M e q e de Dec on epor*

[32] **Engquist, B., Osher, S. and Zhong, S., Fast Wavelet Based Algorithms for Linear Evolution Equations. Preprint, 1991.**

[33] **Sachdev, P.L.,** *Non ne r D e e* **, Cambridge University Press (1987).**

[34] **Schult, R.L. and Wyld, H.W., Using Wavelets to Solve the Burgers' Equation: A Comparative Study.** *P y c e e A* **46 (1992), 12.**

[35] **Whitham, G.B.,** *L ne r nd Non ne r e* **, John Wiley and Sons, NY, 1974.**

[36] **Wickerhauser, M.V.,** *Ad p ed e e An y fro eory o of re*, **A. K. Peters, Ltd. Wellesley, Massachusetts, 1994.**

[37] **Yosida, K.,** *nc on An y* **, Springer-Verlag, 1980.**