MULTIRESOLUTION-ENHANCED BLOCK-STRUCTURED ADAPTIVE MESH REFINEMENT

A PREPRINT

Brandon Gusto
Department of Scientific Computing
Florida State University
Tallahassee, FL 32306
bgusto@fsu.edu

Tomasz Plewa
Department of Scientific Computing
Florida State University
Tallahassee, FL 32306
tplewa@fsu.edu

May 7, 2019

ABSTRACT

We present a generalization of Harten's multiresolution scheme for solving multiphysics problems on Cartesian, block-structured adaptive mesh refinement (AMR) grids. The scheme addresses a shortcoming of tree-based AMR codes, which is the creation of blocks with a low filling factor; that is, many cells in such a block are resolved beyond the desired error tolerance. To overcome this issue, a multiresolution (MR) representation is introduced not only to adapt grid but also to select fluxes which may be interpolated from the MR basis. The flux interpolation error introduced by this approximation is of the same order as the local truncation error of the reconstruction scheme. Thus the rate of convergence of the underlying spatial reconstruction scheme is preserved. This procedure can be easily incorporated into existing parallel codes. Additionally, the MR transform is applied per AMR block, obviating any need for communication of MR information between blocks. The efficiency of the scheme is demonstrated using several one and two-dimensional problems.

Keywords Multiresolution · Adaptive Mesh Refinement · Conservation Laws

1 Introduction

The dynamics of reactive flows are typically characterized by disparate spatial and temporal length scales. Certain features of the flow such as shocks, material discontinuities, or burning fronts often require significantly higher spatial resolution than other regions of the flow. Naturally, intense effort has gone into the development of methods which employ a multi-scale or adaptive strategy to accurately simulate such flows without over-resolving large areas of the domain.

Methods which introduce a hierarchy of nested grid levels are generally described as adaptive mesh refinement (AMR) methods. Obtaining an estimate of the local truncation error (LTE) of the reconstruction scheme on a particular grid level enables the scheme to identify regions where refinement is necessary. AMR methods may employ several strategies to approximate the LTE. The earliest work utilized gradient information to flag cells for refinement [**?**]. Alternative methods include feature-based refinement, and evaluation of gradient information (references here).

In a seminal paper, Harten (cite) introduced a scheme based on multiresolution (MR) analysis for numerically solving hyperbolic partial differential equations. The MR scheme represents the discrete solution on a set of nested grids using the MR basis. This basis is constructed using average-interpolating wavelets

Other methods are based on multiresolution analysis, refining the grid based on the absolute value of wavelet coefficients (long list of papers, harten, chiavassa, domingues, bla bla).

2 Multiresolution Analysis

A multiresolution analysis (MRA) provides a hierarchy of nested approximation spaces to construct a basis in $L^2(\mathbb{R}^{\mathbf{d}})$. While a Fourier basis decomposes a signal into only frequency components, a wavelet basis captures both frequency and spatial localization. A MRA is defined by a sequence of nested subspaces

$$\cdots \subset V_{j-1} \subset V_j \subset V_{j+1} \subset \dots \quad j \in \mathbb{Z},$$

where each subspace V_j is associated with a set of points γ_j. In the finite dimensional setting, a given signal is split into odd and even components, and regularity information is obtained by the accuracy of the prediction of odd components by even components. Considering successively finer approximation spaces yields

$$V_j = V_0 \oplus W_0 \oplus W_1 \oplus \cdots \oplus W_{j-1},$$

thus the fine-scale information is represented by the coarsest scale plus a series of differences at higher levels.

3 Finite Volume Framework

In the present work we are interested in numerically solving conservation laws of the form

$$\begin{cases} u_t + f(u)_x = s(u) \\ u(x,0) = u_0(x), \end{cases} \tag{1}$$

where u represents a conserved quantity, $f(u)$ is the flux function, and $s(u)$ is a source term. In the finite volume formulation, the solution $u(x,t)$ is approximated by a volume average defined over each cell $I_i = \left[x_i - \frac{h}{2}, x_i + \frac{h}{2}\right]$ as

$$u_i(t) = \frac{1}{h} \int_{x_i - \frac{h}{2}}^{x_i + \frac{h}{2}} u(\xi, t) d\xi, \tag{2}$$

where h is the cell width. For convenience we use $\pm\frac{1}{2}$ to indicate the left and right interfaces of a cell. The governing equations are cast into the conservative form

$$\frac{du_i(t)}{dt} = -\frac{1}{h}\left(f(u(x_{i+\frac{1}{2}})) - f(u(x_{i-\frac{1}{2}})),\right), \tag{3}$$

where $\hat{f}_{i\pm\frac{1}{2}}$ are numerical fluxes.

3.1 Multiresolution Scheme

The multiresolution basis used in the following scheme is based on average- interpolating wavelets (reference here). The forward wavelet transform consists of two steps:

Split: The cells at level l are split into even and odd components.

Predict: The cell averages at level $l+1$ are predicted based on an average-interpolating stencil.

3.2 Grid Hierarchy

In Harten's method, the domain is discretized into a hierarchy of uniformly-spaced, nested grids. In Cartesian coordinates, the grid is defined by

$$\mathcal{G}^l = \left\{x_i^l\right\}_{i=0}^{N_l}, \quad x_i^l = i \cdot h^l, \;\; h^l = 2^l \cdot h^0, \;\; N_l = N_0/2^l, \tag{4}$$

where on level l, h_l is the cell width and N_l is the number of cells.

3.3 Forward Transform

In Table (1) the coefficients are shown for several average-interpolating stencils

$$\tilde{u}_{2i+1}^{l-1} \approx u_i^l - \sum_{p=1}^{s} \gamma_p \left(u_{i-p}^l - u_{i+p}^l\right). \tag{5}$$

	γ_{m-2}	γ_{m-1}	γ_m	γ_{m+1}	γ_{m+2}
$n=3$	0	$-\frac{1}{8}$	1	$\frac{1}{8}$	0
$n=5$	$-\frac{3}{128}$	$\frac{22}{128}$	1	$-\frac{22}{128}$	$\frac{3}{128}$

Table 1: Coefficients for cell-average interpolation in the prediction step.

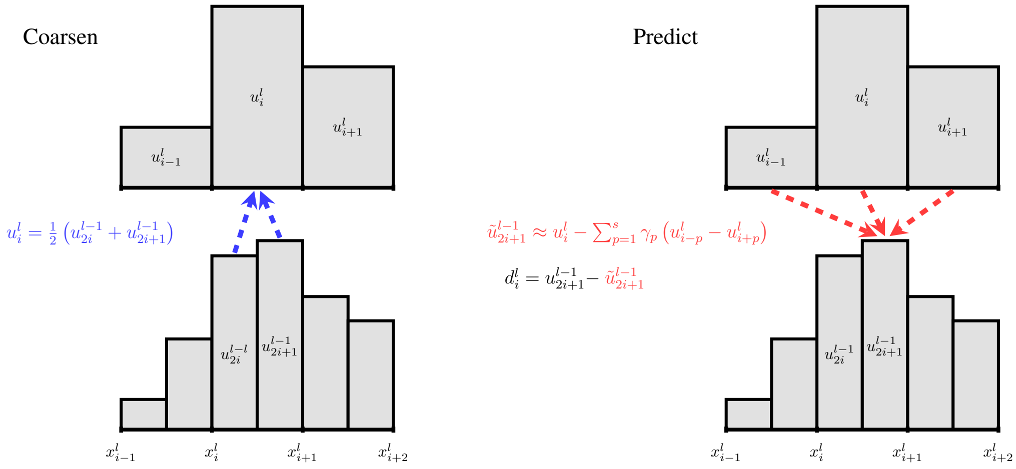

Figure 1: Left: quadratic prediction from coarse-scale j to fine-scale $j+1$, given cell-averages $\mathbf{c}^j$. Right: Fine-scale cell averages are coarsened.

The detail coefficients are then computed as

$$d_i^l = u_{2i+1}^{l-1} - \tilde{u}_{2i+1}^{l-1}. \tag{6}$$

These steps are illustrated in Figure (2). **Once the detail coefficients have been obtained, the MR scheme proceeds by setting a threshold ϵ and truncating coefficients which have an absolute value below the threshold.** Lastly, the inverse transform then starts from grid $l = L$ and at each interface either computes fluxes using the fine-grid scheme, or interpolates them using the MR basis. The fluxes are interpolated by

$$\tilde{f}_{2i+1}^{l-1} \approx \sum_{p=1}^{s+1} \beta_p \left(\hat{f}_{i-p+1}^l + \hat{f}_{i+p}^l\right), \tag{7}$$

where the interpolants are of degree $2s+1$. The process repeats until all fluxes are either computed or interpolated on the fine grid $l = 0$.

4 Numerical Tests

	$\epsilon = 0.0$				$\epsilon = 10^{-12}$			
grid cells	L_1 error	order	L_∞ error	order	L_1 error	order	L_∞ error	order
16								
32								
64								
128								
256								
	$\epsilon = 10^{-6}$				$\epsilon = 10^{-4}$			
grid cells	L_1 error	order	L_∞ error	order	L_1 error	order	L_∞ error	order
16								
32								
64								
128								
256								

4.1 Example

Using the inviscid flow assumption, the dynamics of compressible fluids are modeled using the reactive Euler equations add domain notation

$$u_t + f(u)_x + g(u)_y = s(u), \tag{8}$$

where $u = (\rho, \rho u, \rho v, \rho w, E)^T$ is the state vector, the flux vectors are given by

$$f = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uw \\ u(E+p) \end{pmatrix}, \quad g = \begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ \rho vw \\ v(E+p) \end{pmatrix}, \quad h = \begin{pmatrix} \rho w \\ \rho uw \\ \rho vw \\ \rho w^2 + p \\ w(E+p) \end{pmatrix}, \tag{9}$$

and $s(u)$ represents sources. The total energy per unit volume is given by

$$E = \rho \left(\frac{1}{2}\mathbf{V}^2 + e \right),$$

where e is the internal energy and the kinetic energy contribution is

$$\frac{1}{2}\mathbf{V}^2 = \frac{1}{2}\mathbf{V} \cdot \mathbf{V} = \frac{1}{2}\left(u^2 + v^2 + w^2\right).$$

The system of nonlinear equations is closed by an equation of state which is in general not derived from that of an ideal gas.

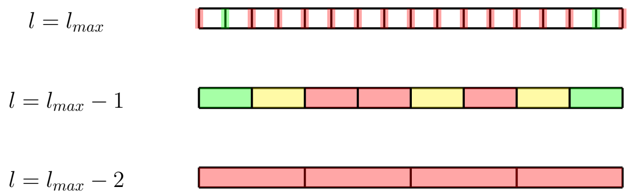

Figure 2: An illustrative case of the multiresolution decomposition applied to a block of data 16 cells wide, showing two coarser grid levels, with green and red cells indicating insignificant and significant detail coefficients, respectively. Yellow cells represent the buffer region surrounding a cell with a large detail coefficient.

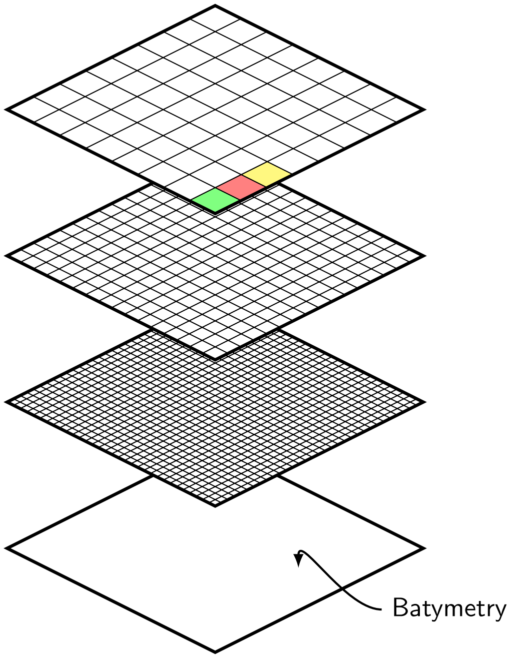

Figure 3: Wavelet decomposition of a 32×32 block of cells with three grid levels. Coloring indicates cells with a significant detail coefficient (red), cells in the buffer region (yellow), and cells with a small detail coefficient that are not in the buffer region (green).

5 Acknowledgements

A Literature Review

B Derivation of Prediction Operator in One-Dimension

We are interested in obtaining the difference between approximation spaces at varying levels of resolution. We are given cell-averaged values as input data to our wavelet transform. This data is fed to the scheme at some arbitrary maximum resolution level J, and the wavelet transform produces details coefficients at each lower level until the coarsest level, $j = 0$, is reached. The coefficients in this case are interchangeable with the cell-averages and are denoted by c_k^j, where the level of resolution is denoted by j, and the spatial index is denoted by k. We consider an interpolating polynomial $p(x)$ such that

$$c_{k-1}^j = \int_{x_{k-1}^j}^{x_k^j} p(x)dx \tag{10}$$

$$c_k^j = \int_{x_k^j}^{x_{k+1}^j} p(x)dx \tag{11}$$

$$c_{k+1}^j = \int_{x_{k+1}^j}^{x_{k+2}^j} p(x)dx. \tag{12}$$

The polynomial $p(x)$ should then predict the finer cell-averages of cell c_k^j as

$$\hat{c}_{2k}^{j+1} = 2\int_{x_k^j}^{x_{k+1/2}^j} p(x)dx \tag{13}$$

$$\hat{c}_{2k+1}^{j+1} = 2\int_{x_{k+1/2}^j}^{x_{k+1}^j} p(x)dx \tag{14}$$

At present, it may not be clear how to implement such a scheme on a computer. However this interpolation procedure can be cast in a more suitable form by introducing another polynomial, the integral of $p(x)$:

$$P(x) = \int_0^x p(y)dy. \tag{15}$$

Now the problem is to interpolate the following data

$$0 = P(x^j_{k-1}) \tag{16}$$
$$c^j_{k-1} = P(x^j_k) \tag{17}$$
$$c^j_{k-1} + c^j_k = P(x^j_{k+1}) \tag{18}$$
$$c^j_{k-1} + c^j_k + c^j_{k+1} = P(x^j_{k+2}). \tag{19}$$

This can easily be done using Lagrange polynomials. Then the predictions are given in terms of $P(x)$ by

$$\hat{c}^{j+1}_{2k} = 2\left(P(x^j_{k+1/2}) - P(x^j_k)\right) \tag{20}$$
$$\hat{c}^{j+1}_{2k+1} = 2\left(P(x^j_{k+1}) - P(x^j_{k+1/2})\right). \tag{21}$$

This interpolating polynomial is cast in the Lagrange form,

$$P(x) = \sum_{i=0}^{n} y_i l_i(x), \tag{22}$$

where y_i are the functional data, and $l_i(x)$ are the Lagrange polynomials. For $n = 3$ these are given by

$$l_0(x) = \frac{x - x_1}{x_0 - x_1}\frac{x - x_2}{x_0 - x_2}\frac{x - x_3}{x_0 - x_3} \tag{23}$$
$$l_1(x) = \frac{x - x_0}{x_1 - x_0}\frac{x - x_2}{x_1 - x_2}\frac{x - x_3}{x_1 - x_3} \tag{24}$$
$$l_2(x) = \frac{x - x_0}{x_2 - x_0}\frac{x - x_1}{x_2 - x_1}\frac{x - x_3}{x_2 - x_3} \tag{25}$$
$$l_3(x) = \frac{x - x_0}{x_3 - x_0}\frac{x - x_1}{x_3 - x_1}\frac{x - x_2}{x_3 - x_2}, \tag{26}$$

and the final interpolating polynomial is

$$P(x) = (0)l_0(x) + (c^j_{k-1})l_1(x) + (c^j_{k-1} + c^j_k)l_2(x) + (c^j_{k-1} + c^j_k + c^j_{k+1})l_3(x). \tag{27}$$

Several evaluations are necessary in order to obtain the predictions. Using intervals of equal length, these values are

$$P(x^j_k) = c^j_{k-1} \tag{28}$$
$$P(x^j_{k+1/2}) = \frac{17}{16}c^j_{k-1} + \frac{1}{2}c^j_k - \frac{1}{16}c^j_{k+1} \tag{29}$$
$$P(x^j_{k+1}) = c^j_{k-1} + c^j_k. \tag{30}$$

Then the predictions of the cell-averages at the higher level of resolution are finally given by

$$\hat{c}^{j+1}_{2k} = c^j_k + \frac{1}{8}\left(c^j_{k-1} - c^j_{k+1}\right) \tag{31}$$
$$\hat{c}^{j+1}_{2k+1} = c^j_k - \frac{1}{8}\left(c^j_{k-1} - c^j_{k+1}\right). \tag{32}$$

This procedure could easily be extended to non-uniformly spaced intervals, giving different weights. Note that only the odd indices are counted because in the multiresolution scheme the data is initially split into even and odd signals. All data at level j are just considered to be a copy of the even-index data at level $j + 1$, whereas the odd-indexed data at level $j + 1$ is what is predicted by even-indexed data at level $j + 1$. Also important are the interpolants at the ends of the domain. Given below are the left and right predictions, respectively:

$$\hat{c}^{j+1}_{2k+1} = \frac{5}{8}c^j_k + \frac{1}{2}c^j_{k+1} - \frac{1}{8}c^j_{k+2} \tag{33}$$
$$\hat{c}^{j+1}_{2k+1} = \frac{1}{8}c^j_{k-2} - \frac{1}{2}c^j_{k-1} + \frac{11}{8}c^j_k. \tag{34}$$