Journal of Computational Physics 272 (2014) 814–841

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

WAMR: An adaptive wavelet method for the simulation of compressible reacting flow. Part I. Accuracy and efficiency of algorithm

Samuel Paolucci *, Zachary J. Zikoski, Damrongsak Wirasaet

University of Notre Dame, Notre Dame, IN 46556, United States

ARTICLE INFO

Article history:
Received 20 August 2012
Received in revised form 8 October 2013
Accepted 16 January 2014
Available online 6 February 2014

Keywords:
Wavelet
Adaptive
Compressible reacting flow
Newtonian fluid
Detailed kinetics
Multicomponent diffusion

ABSTRACT

The Wavelet Adaptive Multiresolution Representation (WAMR) method provides a robust method for controlling spatial grid adaptation — fine grid spacing in regions where a solution varies greatly (*i.e.*, near steep gradients, or near-singularities) and a much coarser grid where the solution varies slowly. Subsequently, a wide range of spatial scales, often demanded in challenging continuum physics problems, can be efficiently captured. Furthermore, the wavelet transform provides a direct measure of local error at each collocation point, effectively producing automatically verified solutions.

The method is applied to the solution of unsteady, compressible, reactive flow equations, and includes detailed diffusive transport and chemical kinetics models. Accuracy and performance of the method are examined on several test problems. The sparse grids produced by the WAMR method exhibit an impressive compression of the solution, reducing the number of collocation points used by factors of many orders of magnitude when compared to uniform grids of equivalent resolution.

1. Introduction

For numerical simulations to be used as truly predictive tools, solutions of computational physics models must adequately capture all relevant scales embodied in them in an accurate and consistent manner. However, this becomes particularly challenging for problems having a strong multiscale character, where spatial and temporal scales may span several orders of magnitude. Such multiscale problems are encountered in a number of fields of practical interest such as astrophysics, material sciences, meteorology, and combustion.

According to a 2006 Department of Energy Office of Basic Energy Sciences' workshop [1], combustion science has a single grand challenge — the development of a "validated, predictive, multiscale combustion modeling capability to optimize the design and operation of evolving fuels in advanced engines used for transportation." Simulation of combustion is particularly challenging due to its typically multidimensional nature and the wide range of spatial and temporal scales present. In addition to the challenges associated with chemical reactions, this work is primarily concerned with the simulation of *high-speed* reactive flows, where compressibility effects and the presence of phenomena such as shock waves is important. The intricate coupling between the fluid mechanics and chemistry results in detonations being particularly difficult to simulate accurately. For detailed chemistry models in a hydrogen/oxygen mixture, the smallest spatial scales have been estimated to

* Corresponding author.
E-mail addresses: paolucci@nd.edu (S. Paolucci), zzikoski@alumni.nd.edu (Z.J. Zikoski), dwirasae@nd.edu (D. Wirasaet).

http://dx.doi.org/10.1016/j.jcp.2014.01.025
0021-9991/

be of the order of 0.1 μm [2]. This estimate is based purely on the resolution required to capture reaction zone structures, and is of the same order as the mean free path for the mixture. Diffusive effects, like chemical reactions, occur on molecular scales, and the resolution required to capture viscous effects would be of a similar size. Thus, spatial scales span seven or more orders of magnitude. Using a standard finite difference approximation on a uniform grid would require $O(10^{7d})$ grid points for a simulation in d spatial dimensions. Similarly, time scales can range from those associated with chemical reactions (nano-seconds) up to macro time scales (of the order of seconds). Such physical/chemical demands require the use of advanced computational approaches along with modern computational resources.

These multiscale problems are generally impractical to solve (in terms of computer time and memory required) on a fixed grid. However, in many problems of practical interest, small scales only occur in limited regions of the computational domain. Adaptive methods take advantage of this general knowledge by providing fine resolution only in areas where it is needed and coarser resolution elsewhere. There are a number of adaptive methods which have been developed and applied to the solution of partial differential equations (PDEs), including adaptive mesh refinement (AMR), finite element methods (FEM), multigrid, and wavelet methods.

One of the earliest and most popular method is AMR, originally proposed by Berger and Oliger [3]. In AMR, the domain is composed of cells in an initially coarse grid. Individual cells are then flagged for adaption, according to criteria depending on the local solution. Subsequently, a great deal of computational savings can be realized from two sources: the first is the reduction of the total number of grid points or cells required in the solution, and second, all grid points are advanced according to the time step based on the level of refinement.

AMR schemes require a criterion for controlling grid refinement. Originally, Richardson extrapolation was suggested as a means to estimate the error between grid resolutions. In this way, an error tolerance could be set, and where the local error estimate exceeded the tolerance, those cells would be refined. However, Richardson extrapolation is only accurate in regions where the solution is sufficiently smooth; the error estimate is not accurate across steep gradients or near-discontinuities, which is where refinement would be expected. Heuristic methods are also commonly applied, using gradients of the solution to indicate where refinement should occur. Because the actual error introduced by AMR is not well understood, the effectiveness of an adaption criteria requires *a posteriori* error estimates [4].

In FEM the solution in each element is represented by combinations of basis functions, which may take a variety of forms, though polynomials are common. The representation of the solution inside elements provides additional options for adaptivity, beyond mesh refinement. Adaptive FEM may contain any combination of three main types of adaptivity: h-, p-, and r-adaptation [5]. In h-adaptation, individual elements may be subdivided into several smaller elements, improving mesh resolution locally. An initially coarse mesh is recursively refined according a refinement criterion or local error estimate [6]. In p-adaptivity, solution accuracy is improved by increasing the order of the basis functions used in an element. Using p-adaptation proceeds from a fixed mesh and using low-order basis functions. Gui and Babuska develop much of the theory behind h- and p-adaptation, as well as the combination of the two hp-adaptation schemes in [7] and [8]. It has been found that the combination of geometric refinement and higher-order basis functions in hp-adaptation can result in spectral convergence rates, at the cost of a more complex stiffness matrix and subsequent implementation [9]. The third type of adaptivity found in FEM is r-adaptation, the movement or relocation of elements. This type of adaptation iteratively relocates elements to minimize discretization errors while maintaining a fixed number of degrees of freedom. Such adaptivity is also often coupled with h-adaptation [10], though perhaps not explicitly identified in general h- and hp-adaptation methods.

Criteria used to control the various methods of adaptation in FEM are, like in the case of AMR, limited generally to *a posteriori* or heuristic methods [11–13]. For hyperbolic or parabolic problems, iterative creation of the mesh would be quite costly, as features may evolve and move throughout the domain [14].

Multigrid methods use the residual of the solution to determine where refinement is needed. In this manner, a solution to a discretized PDE can be found to a prescribed error tolerance, while reducing the total number of computations needed [15,16]. Being a relaxation method, adaptive multigrid methods are typically used to solve elliptic problems. However, an adaptive multigrid method has been applied to the solution of the Euler equations by using a space–time discretization [17].

Wavelets are mathematical functions with compact support in both location and scale. Wavelet representation of signals have been noted for their compression properties, and have found significant use in signal processing and data analysis. In the adaptive solution of PDEs, wavelets have been applied in a number of methods such as Galerkin formulations, collocation, and finite-difference methods [18].

Wavelets were first applied as basis functions in Galerkin formulations, using thresholding of wavelet amplitudes to substantially reduce the number of degrees of freedom [19]. However, treatment of non-linear terms and evaluations of integrals of wavelet values arising in the method offset potential reductions in execution time. To improve computational efficiency, techniques borrowed from pseudo-spectral methods are used, where nonlinear terms are computed in real space and transformed into wavelet space. Liandrat et al. [20] develop a dynamically adaptive method for unsteady problems using a Petrov–Galerkin formulation. Many approaches have been developed for handling irregular domains and non-periodic boundary conditions [21,22].

Wavelet collocation methods solve for function values at collocation points rather than for wavelet amplitudes as in wavelet-Galerkin methods. This difference circumvents many of the difficulties associated with non-linear terms. Many of the collocation methods also require far fewer operations per collocation point for both linear and non-linear terms, as well as allow the use of general boundary conditions on finite domains [23,24]. Dynamically adaptive algorithms for initial value