Journal of Computational Physics 274 (2014) 370–392

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

Asynchronous finite-difference schemes for partial differential equations

Diego A. Donzis *, Konduri Aditya

Department of Aerospace Engineering, Texas A&M University, College Station, TX 77843, United States

ARTICLE INFO

Article history:
Received 25 March 2014
Received in revised form 5 June 2014
Accepted 11 June 2014
Available online 17 June 2014

Keywords:
Asynchronous schemes
Partial differential equations
Massive computations

ABSTRACT

Current trends in massively parallel computing systems suggest that the number of processing elements (PEs) used in simulations will continue to grow over time. A known problem in this context is the overhead associated with communication and/or synchronization between PEs as well as idling due to load imbalances. Simulation at extreme levels of parallelism will then require an elimination, or at least a tight control of these overheads. In this work, we present an analysis of common finite difference schemes for partial differential equations (PDEs) when no synchronization between PEs is enforced. PEs are allowed to continue computations regardless of messages status and are thus asynchronous. We show that while stability is conserved when these schemes are used asynchronously, accuracy is greatly degraded. Since message arrivals at PEs are essentially random processes, so is the behavior of the error. Within a statistical framework we show that average errors drop always to first-order regardless of the original scheme. The value of the error is found to depend on both grid spacing as well as characteristics of the computing system including number of processors and statistics of the delays. We propose new schemes that are robust to asynchrony. The analytical results are compared against numerical simulations.

1. Introduction

Many natural and engineered systems and processes can be accurately described by partial differential equations (PDEs). This includes fluid mechanics, electromagnetism, quantum mechanics as well as common simple processes in continuum media such as diffusion and wave propagation. In a number of real applications, however, due to the complexity of the equations themselves as well as the geometrical aspects of the problem, analytical solutions are not known and numerical simulations provide invaluable information to understand these systems.

Even with numerical simulations, the complexity of systems at realistic conditions typically requires massive computational resources. In the last few decades, this computational power has been realized through increasing levels of parallelism. When a problem is decomposed into a number of processing elements (PEs), solving the PDE typically requires communication between PEs to compute spatial derivatives. As the number of PEs increases, this communication becomes more challenging (e.g. [1,2]). In fact, this may well be a major bottleneck at the next generation of computing systems [3] which may comprise an extremely large number of PEs. At those extreme levels of parallelism, even small imbalances due to noise [4] in otherwise perfectly balanced codes can represent enormous penalties as PEs idle waiting to receive data from other

* Corresponding author.
E-mail addresses: donzis@tamu.edu (D.A. Donzis), akonduri@tamu.edu (K. Aditya).

http://dx.doi.org/10.1016/j.jcp.2014.06.017
0021-9991/

PEs. This is especially critical when, as commonly done, a global synchronization is imposed at each time step to finalize all communications as well as to obtain information to determine the time-step size in unsteady calculations subjected to a so-called Courant–Friedrichs–Lewy condition.

Thus, in order to take advantage of computational systems at extreme levels of parallelism, relaxing all (especially global) synchronizations is of prime necessity [3]. This is the main thrust of this work. In particular, our objective here is to study widely employed finite difference schemes but used in such a way to avoid all synchronizations as well as potential idle time. While it is currently possible to implement asynchronous communications using modern hardware and software, current numerical schemes still require synchronizations at the mathematical level. In this work we propose schemes which are asynchronous at a mathematical level and thus relax this strong constraint. These new schemes do not require updated values from other PEs and can proceed with calculations without interruption. Studying the numerical properties of these schemes including stability, consistency and accuracy as well as suggesting a framework for devising new formulations resilient to asynchrony are the main objectives of this work.

Substantial efforts have been devoted to asynchronous algorithms in different contexts such as linear systems of equations or more general fixed-point formulations [5,6]. Asynchronous linear solvers have been used in the context of PDEs to solve linear systems required in the computations but global synchronizations are typically still required. Due to the importance of increasing parallelism, other approaches have also been investigated such as explicit–implicit methods (e.g. [7]) though, again, some type of synchronization at some point during the calculations is typically unavoidable or so-called discrete event-driven simulations (e.g. [8]). Work exploiting asynchrony to solve directly problems governed by time-dependent PDEs, however, has been more limited. For example, some studies that focused on PDEs [9,10] were limited to a particular class of PDE (heat equation) and the order of accuracy of the resulting schemes remained low. Further extensions to higher orders has also been limited (e.g. [11] for second order schemes). We study the behavior of general finite differences when used in an asynchronous manner. We provide general results on stability and accuracy that can be applied to a broader class of problems. Unlike previous studies, our analysis incorporates stochastic characteristics of the performance of the parallel computing system which allows us to determine accuracy based on network performance. We show, for example, that accuracy depends on how the number of PEs is increased as the problem size increases as well as how the time step size is determined based on common stability conditions or physical considerations (some preliminary related results have been presented in Ref. [12]). We also provide a framework to devise new asynchronous schemes.

In Section 2 we introduce the concept behind asynchronous schemes using finite differences. The stability of such schemes is analyzed in Section 3 and the resulting accuracy is studied in Section 4. Numerical simulations to support the theoretical developments in previous sections are shown in Section 5. Due to the loss in accuracy when asynchrony is present, we develop new schemes that can preserve some desired order of accuracy which is presented in Section 6. Conclusions and further discussion are included in Section 7.

2. Concept

Our interest is in the general linear PDE:

$$\frac{\partial u}{\partial t} = \sum_{d=1,D} \beta_d \frac{\partial^d u}{\partial x^d} \tag{1}$$

with D being the highest derivative in the PDE and the constants β_d's determine the characteristic of the different physical processes represented by the different terms. Particular cases of interest are the wave equation ($D = 1$ with $\beta_1 \neq 0$), the heat equation ($D = 2$ with $\beta_1 = 0$ and $\beta_2 \neq 0$), and the advection–diffusion equation ($D = 2$ with $\beta_1 \neq 0$ and $\beta_2 \neq 0$).

For illustration purposes consider the unsteady one-dimensional heat (or diffusion) equation

$$\frac{\partial u}{\partial t} = \alpha \frac{\partial^2 u}{\partial x^2}, \tag{2}$$

where $u(x,t)$ is the temperature at a spatial location $x \in [0,l]$ and time t and α is the thermal diffusivity of the medium. With N uniformly distributed grid points, Eq. (2) can be discretized using a second-order central difference in space and first-order forward difference in time to obtain a numerical scheme with well-know characteristics [13]:

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \alpha \frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{\Delta x^2} + \mathcal{O}(\Delta t, \Delta x^2) \tag{3}$$

where u_i^n is the temperature at a point $x = x_i$ and time level n. Here $x_i = i\Delta x$ with $\Delta x = l/N$ being the grid spacing and $i = 1, \dots, N$. We will assume, unless explicitly mentioned, periodic boundary conditions. The time step size is Δt. The last term represents the order of the truncation error in time and space for this approximation.

The scheme in Eq. (3) can be rewritten in the following form

$$u_i^{n+1} = u_i^n + \frac{\alpha \Delta t}{\Delta x^2}\left(u_{i+1}^n - 2u_i^n + u_{i-1}^n\right) \tag{4}$$