Numer. Math. Theor. Meth. Appl. Vol. **10**, No. 4, pp. 872-894
doi: 10.4208/nmtma.2017.m1624 November 2017

Adaptive Dimensionality-Reduction for Time-Stepping in Differential and Partial Differential Equations

Xing Fu and J. Nathan Kutz*

Department of Applied Mathematics, University of Washington, Seattle, WA 98195, USA

Received 6 July 2016; Accepted (in revised version) 13 April 2017

Abstract. A numerical time-stepping algorithm for differential or partial differential equations is proposed that adaptively modifies the dimensionality of the underlying modal basis expansion. Specifically, the method takes advantage of any underlying low-dimensional manifolds or subspaces in the system by using dimensionality-reduction techniques, such as the proper orthogonal decomposition, in order to adaptively represent the solution in the optimal basis modes. The method can provide significant computational savings for systems where low-dimensional manifolds are present since the reduction can lower the dimensionality of the underlying high-dimensional system by orders of magnitude. A comparison of the computational efficiency and error for this method are given showing the algorithm to be potentially of great value for high-dimensional dynamical systems simulations, especially where slow-manifold dynamics are known to arise. The method is envisioned to automatically take advantage of any potential computational saving associated with dimensionality-reduction, much as adaptive time-steppers automatically take advantage of large step sizes whenever possible.

AMS subject classifications: 74S25, 65M70, 37L65

Key words: Time-stepping, dimensionality reduction, proper orthogonal decomposition.

1. Introduction

Computation is ubiquitous across the physical, biological and engineering sciences, revolutionizing many fields of study by allowing one to solve complex problems through algorithmic/numerical methods. And with the continued and significant increase in computing power for a fixed cost, even problems thought once to be intractable are now routinely solved with high-performance computing architectures or even desktop/laptop computing. The role of improved processor performance is unquestioned in helping to revolutionize the impact of computational science. However, equally as important are the algorithmic developments over the past few decades that have exploited any mathematical structure

*Corresponding author. *Email address:* `kutz@uw.edu` (J. N. Kutz)

in the equations of interest to its full advantage. As an example, one needs only to consider adaptive time-stepping algorithms that are applied to the solutions of differential and partial differential equations. Such algorithms aim to take as large a time-step as possible while being constrained to some absolute or relative error measure. So prevalent are these adaptive time-steppers that they are the standard, and expected, routines in most high-level language packages such as MATLAB, Octave or Scipy. In a similar fashion, we propose a simple and efficient *adaptive* time-stepping scheme where the *dimensionality* of the underlying system is adapted in order to more efficiently solve a given differential or partial differential equation system. Although dimensionality reduction techniques for ODEs and PDEs are well known (See [1, 2] and references therein), the method advocated here provides a new, natural framework for a fully adaptive, robust and general time-stepping scheme which exploits optimal proper orthogonal decomposition (POD) basis modes whenever possible.

Dimensionality-reduction techniques, although around for more than a century [3–6], have recently grown in importance for solving a wide range of physical problems. The rise of such techniques often is related to the numerical discretization of PDE systems, for instance, that can often yield a system of equations with millions or billions of degrees of freedom (or more). Thus dimensionality reduction methods can form the underpinnings of developing reduced-order models through projections such as the proper orthogonal decomposition [1–9]. This manuscript aims to bring together well-known dimensionality reduction and adaptive time-stepping schemes with the goal of producing an intuitively appealing and natural, perhaps even obvious, time-stepping algorithm for differential and partial differential equations that automatically takes advantage of any low-dimensional, slow-manifold dynamics that exists in the system. Thus an adaptive time-stepping algorithm is demonstrated where dimensionality reduction is exploited whenever possible in an automated way. The basis of such a strategy is rooted in the vast literature on coherent structures (low-dimensionality) that are observed to be prevalent in, for instance, PDE systems across a broad range of the physical, biological and engineering sciences [16]. Indeed, some underlying low-dimensional and slow-manifold seems to be often exhibited in the dynamics of high-dimensional dynamical systems.

Many other techniques have also been recently advocated with the aim of exploiting some form of dimensionality reduction. In Schaeffer et al. [17] and Jokar et al. [18], *sparsity* is exploited for minimizing the number of basis functions for evaluations of PDE.

By reducing or compressing the information needed to represent the solution at every step, only the essential dynamics are represented. Thus the basis modes are not modified, they are simply turned on and off as necessary at each time-step. Alternatively, one can capitalize on the newly developed dynamic mode decomposition (DMD) technique which projects onto a lower modal basis and evolves the future state according to a linear set of differential equations [19]. Although dimensionality reduction is achieved, the time-stepping equations are constrained to be linear. In yet another technique, an empirical interpolation method (DEIM) [20, 21] provides a modification of the POD method that reduces the complexity of evaluating the nonlinear terms of the reduced model to a cost proportional to the number of reduced variables obtained by POD. However, no