East Asian Journal on Applied Mathematics
doi: 10.4208/eajam.2023-065.050423

Vol. **13**, No. 3, pp. 435-463
August 2023

The Random Feature Method for Time-Dependent Problems

Jing-Run Chen[1,2], Weinan E[3] and Yi-Xin Luo[1,2,*]

[1]*School of Mathematical Sciences, University of Science and Technology of China, Hefei 230026, China.*
[2]*Suzhou Institute for Advanced Research, University of Science and Technology of China, Suzhou 215123, China.*
[3]*AI for Science Institute, Beijing and Center for Machine Learning Research and School of Mathematical Sciences, Peking University.*

Received 21 February 2023; Accepted (in revised version) 5 April 2023.

Dedicated to Professor Tao Tang on the occasion of his 60th birthday.

Abstract. We present a framework for solving time-dependent partial differential equations (PDEs) in the spirit of the random feature method. The numerical solution is constructed using a space-time partition of unity and random feature functions. Two different ways of constructing the random feature functions are investigated: feature functions that treat the spatial and temporal variables (STC) on the same footing, or functions that are the product of two random feature functions depending on spatial and temporal variables separately (SoV). Boundary and initial conditions are enforced by penalty terms. We also study two ways of solving the resulting least-squares problem: the problem is solved as a whole or solved using the block time-marching strategy. The former is termed the space-time random feature method (ST-RFM). Numerical results for a series of problems show that the proposed method, i.e. ST-RFM with STC and ST-RFM with SoV, have spectral accuracy in both space and time. In addition, ST-RFM only requires collocation points, not a mesh. This is important for solving problems with complex geometry. We demonstrate this by using ST-RFM to solve a two-dimensional wave equation over a complex domain. The two strategies differ significantly in terms of the behavior in time. In the case when block time-marching is used, we prove a lower error bound that shows an exponentially growing factor with respect to the number of blocks in time. For ST-RFM, we prove an upper bound with a sublinearly growing factor with respect to the number of subdomains in time. These estimates are also confirmed by numerical results.

AMS subject classifications: 65M20, 65M55, 65M70

Key words: Time-dependent PDEs, partition of unity method, random feature method, collocation method, separation-of-variables random features.

*Corresponding author. *Email addresses:* `jingrunchen@ustc.edu.cn` (J.-R. Chen), `weinan@math.pku.edu.cn` (W.-N. E), `seeing@mail.ustc.edu.cn` (Y.-X. Luo)

1. Introduction

Time-dependent partial differential equations (PDEs), such as diffusion equation, wave equation, Maxwell equation, and Schrödinger equation, are widely used for modeling the dynamic evolution of physical systems. Numerical methods, including finite difference method [11], finite element methods [17], and spectral methods [15], have been proposed to solve these PDEs. Despite the great success in theory and application, these methods still face some challenges, to name a few, complex geometry, mesh generation, and possibly high dimensionality.

Along another line, the success of deep learning in computer vision and natural language processing [8] attracts great attention in the community of scientific computing. As a special class of functions, neural networks are proved to be universal approximators to continuous functions [3]. Many researchers seek for solving ordinary and partial differential equations with neural networks [5–7, 9, 14, 16, 19]. Since the PDE solution can be defined in the variational (if exists), strong, and weak forms, deep Ritz method [5], deep Galerkin method [16] and physics-informed neural networks [14], and weak adversarial network [19] are proposed using loss (objective) functions in the variational, strong, and weak forms, respectively. Deep learning-based algorithms have now made it fairly routine to solve a large class of PDEs in high dimensions without the need for mesh generation of any kind.

For low-dimensional problems, traditional methods are accurate, with reliable error control, stability analysis and affordable cost. However, in practice, coming up with a suitable mesh is often a highly non-trivial task, especially for complex geometry. On the contrary, machine-learning methods are mesh-free and only collocation points are needed. Even for low-dimensional problems, this point is still very attractive. What bothers a user is the absence of reliable error control in machine-learning methods. For example, without an exact solution, the numerical approximation given by a machine-learning method does not show a clear trend of convergence as the number of parameters increases.

There are some efforts to combine the merits of traditional methods and deep-learning based methods. The key ingredient is to replace deep neural networks by a special class of two-layer neural networks with the inner parameters fixed, known as random features [12, 13] or extreme learning machine [10]. Random feature functions are proved to be universal approximators as well, meanwhile only the parameters of the output layer need to be optimized, leading to a convex optimization problem. Extreme learning machines are employed to solve ordinary and partial differential equations in [18] and [1], respectively. Spectral accuracy is obtained for problems with analytic solutions, and the simplicity of network architectures reduces the training difficulty in terms of execution time and solution accuracy, compared to deep neural networks. In [4], a special kind of partition of unity (PoU), termed as domain decomposition, is combined with extreme learning machines to approximate the PDE solution and the block time-marching strategy is proposed for long time simulations. Spectral accuracy is obtained in both space and time for analytic solutions, but the error grows exponentially fast in most cases as the simulation time increases. In [2], combining PoU and random feature functions, the random feature method (RFM) is