Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 390148, 14 pages
http://dx.doi.org/10.1155/2014/390148

Research Article

VIM-Based Dynamic Sparse Grid Approach to Partial Differential Equations

Shu-Li Mei

College of Information and Electrical Engineering, China Agricultural University, P.O. Box 53, East Campus, 17 Qinghua Donglu Road, Haidian District, Beijing 100083, China

Correspondence should be addressed to Shu-Li Mei; meishuli@163.com

Received 9 December 2013; Accepted 14 January 2014; Published 27 February 2014

Academic Editors: L. Kong and E. Momoniat

Combining the variational iteration method (VIM) with the sparse grid theory, a dynamic sparse grid approach for nonlinear PDEs is proposed in this paper. In this method, a multilevel interpolation operator is constructed based on the sparse grids theory firstly. The operator is based on the linear combination of the basic functions and independent of them. Second, by means of the precise integration method (PIM), the VIM is developed to solve the nonlinear system of ODEs which is obtained from the discretization of the PDEs. In addition, a dynamic choice scheme on both of the inner and external grid points is proposed. It is different from the traditional interval wavelet collocation method in which the choice of both of the inner and external grid points is dynamic. The numerical experiments show that our method is better than the traditional wavelet collocation method, especially in solving the PDEs with the Nuemann boundary conditions.

1. Introduction

The sparse representation of functions via a linear combination of a small number of basic functions has recently received a lot of attention in several mathematical fields such as approximation theory as well as signal and image processing [1]. The advantage of the sparse grid approach is that it can be extended to nonsmooth solutions by adaptive refinement methods; that is, it can capture the steep waves appearing in the solution of the PDEs. In fact, the boundary conditions can also be taken as the nonsmooth parts appearing in the solution, especially to the Neumann boundary. Furthermore, it can be generalized from piecewise linear to high-order polynomials. Also, more sophisticated basis functions like interpolets, prewavelets, or wavelets can be used in a straightforward way [2]. In practice, the standard piecewise linear multiscale basis in one dimension, that is, the Faber-Schauder, can be viewed as a scaling function in the wavelet analysis. As an interpolation operator, the basis functions act as the Dirac delta function when operating on itself and its derivatives [3]. So, the interpolation wavelet such as the Shannon wavelet, Shannon-Gabor wavelet, Harr wavelet, and the autocorrelation function of the Daubechies scaling function can be taken as the basis function to construct the sparse grid approach directly.

Faber-Schauder and Haar scaling function do not have the smoothness property, so the function and its derivative to be approximated cannot be represented exactly by them. The autocorrelation function of Daubechies scaling function has been widely used in various numerical methods for PDEs such as the wavelet collocation method and the sparse grids method. The Daubechies scaling functions possess almost all the excellent numerical properties, such as orthogonality, smoothness, and compact support, which are helpful in improving numerical accuracy and efficiency. However, the autocorrelation function of the Daubechies scaling function loses the orthogonality. In addition, Daubechies scaling function has no exact analytical expression. This will bring error to the approximation solution from the Daubechies wavelet numerical method.

Cattani studied the properties of the Shannon wavelet function [4], which possesses many advantages such as orthogonality and is continuous and differentiable. It also has the advantage over the Hermite DAF in that it is an interpolating function, producing matrix equations that have the potential to be relatively sparse. In addition, the second-order approximation of a C^2-function, based on Shannon wavelet functions, is given [5]. The approximation is compared with the wavelet reconstruction formula and the error of approximation is explicitly computed [6]. The advantages of the Shannon wavelet have been illustrated in solving PDEs in engineering [7], which can avoid the shortcomings of Daubechies wavelet such as the interpolation property. Furthermore, Cattani studied the fractional calculus problems with the Shannon firstly. A perceived disadvantage of the Shannon scaling function is that it tends towards zero quite slowly. The direct consequence of this is that a large number of the nodal values will contribute significantly when calculating the derivatives of the function to be approximated. It is for that reason that Hoffman et al. constructed the Shannon-Gabor wavelet [8] using the Gaussian window function. In some ways it improves the approximation to a Dirac delta function compared with Shannon wavelet. However, the presence of the Gaussian window destroys the orthogonal properties possessed by the Shannon wavelet, effectively worsening the approximation to a Dirac delta function. In order to test the multilevel interpolation operator constructed in this paper, the Faber-Schauder, Shannon scaling functions and the autocorrelation function of the Daubechies scaling function are taken as the basis employed in the multilevel interpolator to discretize the PDEs in the experiments, respectively.

There are many ways to solve the system of nonlinear ODEs, which are obtained from the discretization of the nonlinear PDEs using the multilevel interpolator. Compared with the finite difference method, the retained grid points are sparse and the dimensionality of the system of ODEs is smaller. This is helpful to improve the efficiency, but the small change of the condition number and the smoothness of the function to be approximated can destroy the exactness of the numerical solution obtained by the traditional difference method. The variational iteration method proposed by Inokuti et al. in 1978 [9], which has been developed by He [10, 11] and widely used in various fields [12–14], is able to give the solution of the nonlinear problems in an infinite series usually converging to an accurate solution rapidly. By means of the precise integration method (PIM), VIM has been generated to solve the system of nonlinear ODEs by Mei and Zhang [15]. In fact, both of the PIM and VIM are the analytical method; so, the impact of the system of ODEs on the choice of the sparse grid points can be neglected.

The dynamic choice of the inner grid points relates with the smoothness and the gradient at each point of the solution to be approximated, and the external grid points relates with the boundary conditions. A better choice of the external grid points can restrict the boundary effect effectively. Most of the schemes are based on the extension of the solution function, such as the interval wavelet in the wavelet collocation method [16] and Lagrange multiplier in the sparse grid approaches [17]. In most cases, the smoothness and the gradient around the boundary are variational dynamically such as the Nuemann boundary conditions. The extension method by means of the Lagrange multiplier is not suited to change the external grid points dynamically.

In our approach we want to achieve several goals so that the solution should be sparse and should be a good approximation. First of all is to construct a multilevel interpolation operator with which the adaptive sparse grid approach can be simplified to a linear combination of the interpolation operators. The operator should be independent with the basic functions. So, we can take different basis functions in the interpolation operator to solve different problems. Second goal is to construct an adaptive sparse grid approach by combining the multilevel interpolation operator and the VIM. The last one is to construct a dynamic choice scheme on the external grid points, so that both of the inner and external grid points are dynamic with the development of the solution, especially to PDEs with the Neumann boundary conditions.

2. Multilevel Interpolator on Sparse Grids

2.1. Interpolating Multiresolution Analysis. Let us start with the interpolating multiresolution analysis [18] that is necessary for a detailed discussion of sparse grids for purposes of interpolation or approximation, respectively. Let $\phi(x)$ be any of the interpolating basis function such as the Shannon, Faber-Schauder, scaling functions or the autocorrelation function of the Daubechies scaling function. This mother of all basis functions can be used to generate an arbitrary $\phi_k^j(x)$ by dilation and translation; that is

$$\phi_k^j(x) = \phi\left(2^j x - k\right), \quad k = 0, 1, 2, \ldots, 2^j. \tag{1}$$

It is easy to check that introducing the spaces

$$V^j := \operatorname{span}\left\langle \phi_k^j, k = 0, 1, 2, \ldots, 2^j \right\rangle \subset L^2(\mathbb{R}). \tag{2}$$

For convenience of notation we use the superscript to denote the level of resolution and the subscript to denote the location in physical space. The sequence $\{V^j\}$ is a multiresolution analysis, which is an increasing sequence of closed subspaces of $L^2(\mathbb{R})$. We call such a structure an interpolating multiresolution analysis due to the fact that the function ϕ verifies what we call interpolation property, that is $\phi_k^j(n2^{-j}) = \delta_{n,k}$. We may then define an interpolation operator $I^j : C^0(0,1) \to V^j$

$$I^j f = \sum_{k=0}^{2^j} f\left(x_k^j\right)\phi_k^j, \quad x_k^j = k2^{-j}. \tag{3}$$

It is obvious that ϕ_k^j is just the nodal point basis of the finite-dimensional space V^j. Additionally, we introduce the hierarchical increments $W^j \subset V^{j+1}$

$$W^j = \operatorname{span}\left\langle \psi_k^j, k = 0, 1, 2, \ldots, 2^j - 1 \right\rangle, \tag{4}$$

where $\psi_k^j = \phi_{2k+1}^{j+1}$.

Let $y_k^j = x_{2k+1}^{j+1}$, we may remark that the function ψ_k^j verifies

$$\psi_k^j\left(y_n^j\right) = \delta_{kn}, \quad \psi_k^j\left(y_n^{j'}\right) = 0, \quad \forall j' < j. \tag{5}$$

It is obvious that

$$V^{j+1} = V^j \bigoplus W^j. \tag{6}$$

Such multiresolution analysis has been extensively investigated in [9]. According to this theory, any function $f \in C^0(0,1)$ can be represented approximately as:

$$f \approx f^j = \sum_{k=0}^{2^{j_0}} \beta_k^{j_0} \phi_k^{j_0} + \sum_{j \geq j_0} \sum_{k=0}^{2^j} \alpha_k^j \psi_k^j. \tag{7}$$

The coefficients $\beta_k^{j_0}$ and α_k^j are defined as:

$$\beta_k^{j_0} = f\left(x_k^{j_0}\right), \qquad \alpha_k^j = f\left(y_k^j\right) - I^j f\left(y_k^j\right), \tag{8}$$

respectively. This shows that the coefficient α_k^j measures the lack of approximation of f by $I^j f$ [19].

2.2. Multilevel Interpolation Operator on Sparse Grids. Equation (7) is the approximate representation of function f, which is not unique since the set of functions is not linearly independent. In this section, we will try to determine the sparsest representation, that is, a representation with a maximal number of vanishing coefficients among the coefficients $\{\alpha_k^j, k = 0, 1, \ldots, 2^j, j \in Z\}$. The conventional scheme in signal processing, acquiring the entire signal and then compressing it, was questioned by Donoho and Elad [20]. Indeed, this technique uses tremendous resources to acquire often very large signals, just to throw away information during compression. The popular solving scheme is the compressed sensing technique proposed by Donoho [21]. In contrast to it, we try to achieve the same goals by constructing a multilevel interpolation operator via combining the interpolating multiresolution analysis described above and the wavelet transform theory [22].

Let us start with the definition of the interpolation operator

$$u^J(x) = \sum_{i \in Z_\Omega^J} I_i(x) u_i^J, \quad Z_\Omega^J := 0, 1, 2, \ldots, 2^J, \tag{9}$$

$I_i(x)$ is the interpolation function. According to the wavelet transform theory, function $u(x)$ can be expressed approximately as:

$$u^J(x) = \sum_{k_0=0}^{2^{j_0}} u\left(x_{k_0}^{j_0}\right) \varphi_{k_0}^{j_0}(x) + \sum_{j=j_0}^{J-1} \sum_{k \in Z^j} \alpha_k^j \psi_k^j(x), \tag{10}$$

where $Z^j := 0, 1, 2, \ldots, 2^j$, and the interpolation wavelet transform coefficient can be denoted as:

$$\begin{aligned} \alpha_k^j &= u\left(x_{j+1}^{2k+1}\right) \\ &\quad - \left[\sum_{k_0=0}^{2^{j_0}} u\left(x_{j_0}^{k_0}\right) \varphi_{j_0}^{k_0}\left(x_{j+1}^{2k+1}\right) + \sum_{j_1=j_0}^{j-1} \sum_{k_1=0}^{2^{j_1-1}} \alpha_{j_1}^{k_1} \psi_{j_1}^{k_1}\left(x_{j+1}^{2k+1}\right) \right] \\ &= \sum_{n=0}^{2^J} \left[R_{j+1,J}^{2k+1,n} - \sum_{k_0=0}^{2^{j_0}} R_{j_0,J}^{k_0,n} \varphi_{j_0}^{k_0}\left(x_{j+1}^{2k+1}\right) \right] u\left(x_J^n\right) \\ &\quad - \sum_{n=0}^{2^J} \sum_{j_1=j_0}^{j-1} \sum_{k_1=0}^{2^{j_1-1}} \alpha_{j_1}^{k_1} \psi_{j_1}^{k_1}\left(x_{j+1}^{2k+1}\right), \end{aligned} \tag{11}$$

where, $0 \leq j \leq J-1$, $k \in Z^j$, $n \in Z^J$, and R is the restriction operator defined as:

$$R_{i,m}^{l,j} = \begin{cases} 1, & x_i^l = x_m^j \\ 0, & \text{others}. \end{cases} \tag{12}$$

Suppose

$$\alpha_k^j = \sum_{n=0}^{2^J} C_{k,n}^{j,J} u\left(x_n^J\right). \tag{13}$$

Substituting (13) into (11), we obtain

$$\begin{aligned} C_{k,n}^{j,J} &= R_{2k+1,n}^{j+1,J} - \sum_{k_0=0}^{2^{j_0}} R_{k_0,n}^{j_0,J} \varphi_{k_0}^{j_0}\left(x_{2k+1}^{j+1}\right) \\ &\quad - \sum_{j_1=j_0}^{j-1} \sum_{k_1=0}^{2^{j_1-1}} C_{k_1,n}^{j_1,J} \psi_{k_1}^{j_1}\left(x_{2k+1}^{j+1}\right). \end{aligned} \tag{14}$$

If $j = j_0$, then

$$C_{k,n}^{j,J} = R_{2k+1,n}^{j+1,J} - \sum_{k_0=0}^{2^{j_0}} R_{k_0,n}^{j_0,J} \varphi_{k_0}^{j_0}\left(x_{2k+1}^{j+1}\right). \tag{15}$$

Substituting the restriction operator (12) and the wavelet transform coefficient (13) into (10), the approximate expression of the solution $u(x)$ can be obtained as:

$$\begin{aligned} u^J(x) = \sum_{i \in Z^J} \Bigg(&\sum_{k_0=0}^{2^{j_0}} R_{k_0,n}^{j_0,J} \varphi_{k_0}^{j_0}\left(x_{2k+1}^{j+1}\right) \\ &- \sum_{j_1=j_0}^{j-1} \sum_{k_1=0}^{2^{j_1-1}} C_{k_1,n}^{j_1,J} \psi_{k_1}^{j_1}\left(x_{2k+1}^{j+1}\right) \Bigg) u\left(x_i^J\right). \end{aligned} \tag{16}$$

According to the definition of the interpolation operator (9), it's easy to obtain the expression of the interpolation operator as follows:

$$I_i(x) = \sum_{k_0=0}^{2^{j_0}} R_{k_0,i}^{j_0,J} \varphi_{k_0}^{j_0}(x) + \sum_{j=j_0}^{J-1} \sum_{k \in Z^j} C_{k,i}^{j,J} \psi_k^j(x). \tag{17}$$

The corresponding m-order derivative of the interpolation operator is

$$D_i^{(m)}(x) = \sum_{k_0=0}^{2^{j_0}} R_{k_0,i}^{j_0,J} \varphi_{k_0}^{j_0(m)}(x) + \sum_{j=j_0}^{J-1} \sum_{k \in Z^j} C_{k,i}^{j,J} \psi_k^{j(m)}(x). \tag{18}$$

Substitute (17) and (18) into the nonlinear PDEs, and it can be changed to a system of nonlinear ODEs, the approximate analytical solution of which can be obtained with the variational iteration method (VIM).

3. Coupling Technique of VIM and Sparse Grid Method for Nonlinear PDEs

As mentioned above, the multilevel interpolation operator is independent of the basic functions; that is, any basis function with the interpolation property can be employed in (17) directly. But the basis function without the mth order derivative cannot be employed in (18) directly. In this section, we just consider the parabolic PDEs with the second order derivative as follows:

$$-\frac{\partial}{\partial x}\left(p(x)\frac{\partial u}{\partial x}\right)+r(x)\frac{\partial u}{\partial x}+q(x)u=\frac{\partial u}{\partial t}+f(x),$$
$$x\in[a,b],\quad (x,t)\in D, \quad (19)$$
$$u(a,0)=\alpha,\qquad p(b)\frac{\partial u(b,0)}{\partial x}+g(b)u(b,0)=\beta,$$

where D is the definition domain in x-t plane.

Therefore, there are two cases that will be discussed in detail in the following. One is that the basic function to be employed in (7) has second-order derivative; the other aims at the Faber-Schauder scaling function.

3.1. Basis Function with C^2 Continuity. Substituting (16) into (19), it is easy to obtain the nonlinear matrix differential equations as follows:

$$L\left(\dot{\mathbf{V}},\mathbf{V},t\right)+N\left(\dot{\mathbf{V}},\mathbf{V},t\right)=\mathbf{G}(t), \quad (20)$$

where L is a linear operator, N a nonlinear operator, and $\mathbf{G}(t)$ is an inhomogeneous term, $\mathbf{V}$ is an n-dimensional unknown vector, and dot stands for the differential with respect to time variable t. For convenience, (20) can be rewritten as:

$$\dot{\mathbf{V}}-\mathbf{H}\mathbf{V}-\mathbf{F}\left(\dot{\mathbf{V}},\mathbf{V},t\right)=0 \quad (21)$$

$\mathbf{H}$ is a given $n\times n$ constant matrix, and $\mathbf{F}(\dot{\mathbf{V}},\mathbf{V},t)$ is an n-dimensional nonlinear external force vector.

According to VIM, we can write down a correction functional as follows:

$$\mathbf{V}_{n+1}(t)=\mathbf{V}_n(t)+\int_0^t\boldsymbol{\lambda}\Big[\dot{\mathbf{V}}_n(\tau)-\mathbf{H}\mathbf{V}_n(\tau)-\mathbf{F}\left(\dot{\widetilde{\mathbf{V}}}_n,\widetilde{\mathbf{V}}_n,\tau\right)\Big]d\tau, \quad (22)$$

where $\boldsymbol{\lambda}$ is a general Lagrange vector multiplier [23] which can be identified optimally via the variational theory. The subscript n denotes the nth approximation and $\widetilde{\mathbf{V}}_n$ is considered as a restricted variation [24–27]; that is, $\delta\widetilde{\mathbf{V}}_n=0$.

Using VIM, the stationary conditions of (22) can be obtained as follows:

$$\begin{gathered}\boldsymbol{\lambda}'+\boldsymbol{\lambda}\mathbf{H}=0,\\ 1+\boldsymbol{\lambda}(\tau)|_{\tau=t}=0.\end{gathered} \quad (23)$$

The Lagrange vector multiplier can therefore be readily identified,

$$\boldsymbol{\lambda}(\tau)=-e^{\mathbf{H}(t-\tau)}. \quad (24)$$

As a result, we obtain the following iteration formula:

$$\mathbf{V}_{n+1}(t)=\mathbf{V}_n(t)-\int_0^t e^{\mathbf{H}(t-\tau)}\Big[\dot{\mathbf{V}}_n(\tau)-\mathbf{H}\mathbf{V}_n(\tau)-\mathbf{F}\left(\dot{\widetilde{\mathbf{V}}}_n,\widetilde{\mathbf{V}}_n,\tau\right)\Big]d\tau. \quad (25)$$

According to VIM, we can start with an arbitrary initial approximation that satisfies the initial condition. So we take the exact analytic solution of $\dot{\mathbf{V}}-\mathbf{H}\mathbf{V}=0$ as the initial approximation; that is,

$$\mathbf{V}_0(t)=e^{\mathbf{H}t}\mathbf{A}, \quad (26)$$

where $\mathbf{A}$ is the given initial value vector.

Substituting (26) into (25) and after simplification, we have

$$\mathbf{V}_{n+1}(t)=\mathbf{V}_n(t)+\int_0^t e^{\mathbf{H}(t-\tau)}\mathbf{F}\left(\dot{\widetilde{\mathbf{V}}}_n,\widetilde{\mathbf{V}}_n,\tau\right)d\tau. \quad (27)$$

According to the theory of matrices, the analytical expression of the external force $\mathbf{F}(\dot{\widetilde{\mathbf{V}}}_n,\widetilde{\mathbf{V}}_n,\tau)$ is required now, but it is not always available except $\mathbf{F}(\dot{\widetilde{\mathbf{V}}}_n,\widetilde{\mathbf{V}}_n,\tau)$ is a constant vector $\mathbf{f}$; that is,

$$\mathbf{F}\left(\dot{\widetilde{\mathbf{V}}}_n,\widetilde{\mathbf{V}}_n,\tau\right)=\mathbf{f}. \quad (28)$$

The integration term of (15) is

$$\int_0^t e^{\mathbf{H}(t-\tau)}\mathbf{f}\,d\tau=\left(e^{\mathbf{H}t}-\mathbf{I}\right)\mathbf{H}^{-1}\mathbf{f}, \quad (29)$$

where the exponential matrix $e^{\mathbf{H}t}$ can be calculated accurately in PIM, and $\mathbf{I}$ is a unit matrix. Substituting (29) into (27), we obtain the variational iteration formula of the matrix differential equation as follows:

$$\mathbf{V}_{n+1}(t)=\mathbf{V}_n(t)+\left(e^{\mathbf{H}t}-\mathbf{I}\right)\mathbf{H}^{-1}\mathbf{f}. \quad (30)$$

$e^{\mathbf{H}t}$ can be solved exactly by means of the precise integration method (PIM) [28].

3.2. Basis Function with C^1 Continuity. Faber-Schauder scaling function is the typical basis with C^1 continuity. For convenience to construct the variational equation, the parameter t should be discretized as $t_0,t_1,t_2,\ldots,t_m,t_m+1,\ldots$, where $t_0=0$, $t_m=m\Delta t$. Then, $\partial u/\partial t$ can be approximated as:

$$\frac{\partial u}{\partial t}\approx\frac{1}{\Delta t}\left[u(x,m\Delta t)-u(x,(m-1)\Delta t)\right]. \quad (31)$$

Substituting above equation into (19), we obtain

$$-\frac{d}{dx}\left(p(x)\frac{du}{dx}\right)+r(x)\frac{du}{dx}+q(x)u=F(x),$$
$$x\in[a,b],\quad (x,t)\in D,$$
$$u(a,0)=\alpha,\qquad p(b)\frac{du(b,0)}{dx}+g(b)u(b,0)=\beta, \quad (32)$$
$$F(x)=f(x)+\frac{1}{\Delta t}\left[u(x,m\Delta t)-u(x,(m-1)\Delta t)\right].$$

Obviously, it is the initial-boundary elliptic PDEs. Using the virtual displacement theory, the variation equation can be obtained as:

$$a(u,v)=G(v), \quad u\in H_E^1(a,b),\ \forall v\in H_{0E}^1(a,b), \tag{33}$$

where

$$\begin{aligned} H_E^1(a,b) &:= \left\{u\in H^1(a,b) \mid u(a)=\alpha\right\}, \\ H_{0E}^1(a,b) &:= \left\{v\in H^1(a,b) \mid v(a)=0\right\}, \\ a(u,v) &:= \int_a^b \left[p(x)\frac{du}{dx}\frac{dv}{dx}+r(x)\frac{du}{dx}v+q(x)uv\right]dx \\ &\quad + g(b)u(b)v(b), \\ G(v) &:= \int_a^b F(x)v(x)\,dx+\beta v(b), \end{aligned} \tag{34}$$

$H^1(a,b)$ is the Sobolev space.

According to the interpolation wavelet transform theory, the variables u and v can be approximated as:

$$\begin{aligned} u(x,t) &= \sum_{k=0}^{2^{j_0}} u\left(x_{j_0,k}\right)w_k^{j_0}(x)+\sum_{j=j_0}^{J-1}\sum_{k=0}^{2^j-1}\alpha_{j,k}(t)\,w_{2k+1}^{j+1}(x) \\ v(x,t) &= \sum_{k=0}^{2^{j_0}} v\left(x_{j_0,k}\right)w_k^{j_0}(x)+\sum_{jv=j_0}^{J-1}\sum_{kv=0}^{2^{jv}-1}\alpha_{jv,kv}(t)\,w_{2kv+1}^{jv+1}(x). \end{aligned} \tag{35}$$

The first-order derivatives are

$$\begin{aligned} \frac{d}{dx}u(x,t) &= \sum_{k=0}^{2^{j_0}} u\left(x_{j_0,k}\right)\left(w_k^{j_0}(x)\right)' \\ &\quad + \sum_{j=j_0}^{J-1}\sum_{k=0}^{2^j-1}\alpha_{j,k}(t)\left(w_{2k+1}^{j+1}(x)\right)', \\ \frac{d}{dx}v(x,t) &= \sum_{k=0}^{2^{j_0}} v\left(x_{j_0,k}\right)\left(w_k^{j_0}(x)\right)' \\ &\quad + \sum_{jv=j_0}^{J-1}\sum_{kv=0}^{2^{jv}-1}\alpha_{jv,kv}(t)\left(w_{2kv+1}^{jv+1}(x)\right)', \end{aligned} \tag{36}$$

respectively. Substituting (35)-(36) into (32), the sparse method for the parabolic PDEs based on the Faber-Schauder scaling function will be obtained. The system of ODEs can be solved exactly by means of the precise integration method (PIM).

4. Dynamic Choice Scheme on the External Grid Points

Combining the multilevel interpolation operator with the threshold scheme, it is easy to obtain the sparse inner grid points dynamically. Any adaptive method can capture the steep gradient appearing in the solution; that is, the inner grid points can concentrate around the larger gradient points adaptively. The PDEs in engineering are always defined in the finite domain, so the boundary condition can usually change the smoothness of solution around the boundary. This results in that more grid points around the boundary contribute to the solution and increase the calculation amount. The reasonable choice of the external grid points can decrease the boundary effect and improve the precision of the solution. In this section, we try to give a dynamic choice scheme of the external grid points, which is deduced from concept of the interval interpolation wavelet and is different from it.

4.1. Construction of the Interval Interpolation Wavelet. In general, the interpolation basis functions defined in interval can be represented as:

$$\omega_{jk} = \begin{cases} \omega\left(2^jx-k\right)+\sum\limits_{n=-L+1}^{-1} a_{nk}\omega\left(2^jx-n\right), & k=0,\dots,L \\ \omega\left(2^jx-k\right), & k=L+1,\dots,2^j-L-1 \\ \omega\left(2^jx-k\right)+\sum\limits_{n=2^j+1}^{2^j+L-1} b_{nk}\omega\left(2^jx-n\right), & k=2^j-L,\dots,2^j, \end{cases} \tag{37}$$

where

$$\begin{gathered} a_{nk}=\prod_{\substack{i=L-1\\ i\neq k}}^{-1}\frac{x_{j,n}-x_{j,i}}{x_{j,k}-x_{j,i}}, \qquad b_{nk}=\prod_{\substack{i=2^j+1\\ i\neq k}}^{2^j+1+L}\frac{x_{j,n}-x_{j,i}}{x_{j,k}-x_{j,i}}, \\ x_{j,k}=k\frac{x_{\max}-x_{\min}}{2^j}, \quad k\in\mathbb{Z}, \end{gathered} \tag{38}$$

where L is the amount of the external collocation points; the amount of discrete points in the definition domain is 2^j+1 $(j\in Z)$; $[x_{\min},x_{\max}]$ is the definition domain of the approximated function.

Equations (37) and (38) show that the interval wavelet is derived from the domain extension. The supplementary discrete points in the extended domain are called external points. The value of the approximated function at the external points can be obtained by Lagrange extrapolation method. Using the interval wavelet to approximate a function, the boundary effect can be left in the supplementary domain; that is, the boundary effect is eliminated in the definition domain.

According to (37) and (38), the interval wavelet approximant of the function $f(x)x\in[x_{\min},x_{\max}]$ can be expressed as

$$\begin{gathered} f_j(x)=\sum f_j(x_n)\,W_j\left(2^jx-n\right), \\ x_n=x_{\min}+n\frac{x_{\max}-x_{\min}}{2^j}, \end{gathered} \tag{39}$$

$f_j(x_n)$ is the given value at the discrete point x_n. At the external points, $f_j(x_n)$ can be calculated by extrapolation method; that is,

$$f_j(x_n) = \begin{cases} \sum_{k=0}^{L-1}\left(f_j(x_k) \prod_{\substack{i=0\\ i\neq k}}^{L-1} \dfrac{x_n - x_i}{x_k - x_i}\right), & n = -1, \ldots, -L \\ \sum_{k=2^j-L+1}^{2^j}\left(f_j(x_k) \prod_{\substack{i=2^j-L+1\\ k\neq i}}^{2^j} \dfrac{x_n - x_i}{x_k - x_i}\right), & n = 2^j+1, \ldots, 2^j+L. \end{cases} \tag{40}$$

So the interval wavelet approximant of $f(x)$ can be rewritten as:

$$\begin{aligned} f_j(x) = {} & \sum_{n=-L}^{-1}\left(\sum_{k=0}^{L-1} f_j(x_k) \prod_{i=0}^{L-1}\frac{x_n - x_i}{x_k - x_i}\right)\omega\left(2^j x - n\right) \\ & + \sum_{n=0}^{2^j} f_j(x_k)\,\omega\left(2^j x - n\right) \\ & + \sum_{n=2^j+1}^{2^j+L}\left(\sum_{k=2^j-L}^{2^j} f_j(x_k) \prod_{i=2^j-L}^{2^j}\frac{x_n - x_i}{x_k - x_i}\right)\omega\left(2^j x - n\right). \end{aligned} \tag{41}$$

Let

$$\begin{aligned} \mathrm{LS}_L(x_n) &= \sum_{k=0}^{L-1} f_j(x_k) \prod_{i=0}^{L-1}\frac{x_n - x_i}{x_k - x_i}, \\ \mathrm{LE}_L(x_n) &= \sum_{k=2^j-L}^{2^j} f_j(x_k) \prod_{i=2^j-L}^{2^j}\frac{x_n - x_i}{x_k - x_i} \end{aligned} \tag{42}$$

then

$$\begin{aligned} f_j(x) = {} & \sum_{n=-L}^{-1} \mathrm{LS}_L(x_n)\,\omega\left(2^j x - n\right) + \sum_{n=0}^{2^j} f_j(x_k)\,\omega\left(2^j x - n\right) \\ & + \sum_{n=2^j+1}^{2^j+L} \mathrm{LE}_L(x_n)\,\omega\left(2^j x - n\right). \end{aligned} \tag{43}$$

$\mathrm{LS}_L(x_n)$ and $\mathrm{LE}_L(x_n)$ correspond to the left and the right external points, respectively. They are obtained by Lagrange extrapolation using the internal collocation points near the boundary. So, the interval wavelet's influence on the boundary effect can be attributed to Lagrange extrapolation. It should be pointed out that we did not care about the reliability of the extrapolation. The only function of the extrapolation is enlarging the definition domain of the given function which can avoid the boundary effect occurring in the domain. Therefore, we can discuss the choice of L by means of Lagrange inner- and extrapolation error polynomial as follows:

$$R_L(x) = \frac{f^{(L+1)}(\xi)}{(L+1)!}\prod_{i=0}^{L}(x - x_i), \quad \text{for some } \xi \text{ between } x, x_0, \ldots, x_L. \tag{44}$$

Equation (44) indicates that the approximation error is both related to the smoothness and the gradient of the original function near the boundary. Setting different L can satisfy the different error tolerance requirement.

4.2. Dynamic Choice Scheme of External Points in Sparse Grids Approach. This scheme is made up with 2 steps. First, the Newton interpolation operator is employed instead of the traditional Lagrange interpolation. Second, both of the error tolerance and condition number are taken as the termination procedure of dynamic choice of external grid points. We will discuss it in detail in this section.

In order to construct the dynamic choice scheme of external grid points, the Newton interpolation theory should be introduced instead of the traditional Lagrange interpolation theory. It is well known that the Newton interpolation is equivalent with Lagrange interpolation, but the Lagrange interpolation algorithm has no inheritance which is the key feature of Newton interpolation. So, the advantage of Newton interpolation method is that the basis function does not need to be recalculated as one point is added except only one more term is needed to be added, which reduces the number of compute operation, especially the multiplication.

The expression of Newton interpolation can be written as :

$$\begin{aligned} N_n(x) = {} & f(x_0) + (x - x_0) f(x_0, x_1) \\ & + (x - x_0)(x - x_1) f(x_0, x_1, x_2) \\ & + \cdots + (x - x_0)(x - x_1), \ldots, (x - x_{n-1}) \\ & \times f(x_0, x_1, \ldots, x_n), \end{aligned} \tag{45}$$

substituting the Newton interpolation instead of the Lagrange interpolation into (43), which can be rewritten as:

$$\begin{aligned} f_j(x) = {} & \sum_{n=-L}^{-1}\left(\mathrm{NS}_L(x_n)\right)\omega\left(2^j x - n\right) \\ & + \sum_{n=0}^{2^j} f_j(x_n)\,\omega\left(2^j x - n\right) \\ & + \sum_{n=2^j+1}^{2^j+L}\left(\mathrm{NE}_L(x_n)\right)\omega\left(2^j x - n\right), \end{aligned} \tag{46}$$

where

$$\begin{aligned}\mathrm{NS}_L(x_n) &= f(x_0) + (x_n - x_0) f(x_0, x_1) \\ &\quad + (x_n - x_0)(x_n - x_1) f(x_0, x_1, x_2) \\ &\quad + \cdots + (x_n - x_0)(x_n - x_1), \ldots, (x_n - x_{L-1}) \\ &\quad \times f(x_0, x_1, \ldots, x_L), \\ \mathrm{NS}_R(x_n) &= f(x_{2^j}) + (x_n - x_{2^j}) f(x_{2^j}, x_{2^j-1}) \\ &\quad + (x_n - x_{2^j})(x_n - x_{2^j-1}) f(x_{2^j}, x_{2^j-1}, x_{2^j-2}) \\ &\quad + \cdots + (x_n - x_{2^j})(x_n - x_{2^j-1}), \ldots, (x_n - x_{2^j-L}) \\ &\quad \times f(x_{2^j}, x_{2^j-1}, \ldots, x_{2^j-L}).\end{aligned} \tag{47}$$

It is well known that the Newton interpolation is equivalent to the Lagrange interpolation. The corresponding error estimation can be expressed as:

$$R_n(x) = (x - x_0)(x - x_1), \ldots, (x - x_n) f(x, x_0, \ldots, x_n). \tag{48}$$

And the simplest criterion to terminate the dynamic choice on L is $|R_n(x)| \leq \mathrm{Ta}$ (Ta is the absolute error tolerance). Obviously, it is difficult to define Ta which should meet with the precision requirement of all approximated curves. In fact, the difference coefficient $f(x, x_0, \ldots, x_n)$ can be used directly as the criterion; that is,

$$|f(x, x_0, \ldots, x_n)| < \varepsilon. \tag{49}$$

As mentioned above, once the curves with lower-order smoothness are approximated by higher-order polynomial expression, the errors will become bigger on the contrary. In fact, even if the L is infinite, the computational precision cannot be satisfied except increasing computational complexity. To avoid this, we design the termination procedure of dynamic choice about L as follows :

```
If $f(x_0, x_1) < \mathrm{Ta}$, then $L = 1$
elseif $f(x_0, x_1, x_2) < \mathrm{Ta}$
or $f(x_0, x_1, x_2) < f(x_0, x_1)$, then $L = 2$
elseif $f(x_0, x_1, x_2, x_3) < \mathrm{Ta}$
or $f(x_0, x_1, x_2, x_3) < f(x_0, x_1, x_2)$, then $L = 3$
⋮
```

In the field of numerical analysis, the condition number of a function with respect to an argument measures how much the output value of the function can change for a small change in the input argument. This is used to measure how sensitive a function is to changes or errors in the input, and how much error in the output results from an error in the input. There is no doubt that the choice of L can change the condition number of the system of algebraic equations discretized by the wavelet interpolation operator or the finite difference method. Therefore, the choice of L should take the condition number into account. In fact, if the condition number $\mathrm{cond}(A) = 10^k$, then you may lose up to k digits of accuracy on top of what would be lost to the numerical method due to loss of precision from arithmetic methods. According to the general rule of thumb, the choice should follow the rule as follows:

$$\frac{\mathrm{Cond}(A_{L+1})}{\mathrm{Cond}(A_L)} < 10. \tag{50}$$

The computational complexity of interpolation calculation is not proportional to the increasing points. The former is mainly up to the computation amount of $(x - x_0)(x - x_1), \ldots, (x - x_n)$ and the derivative operations. Obviously, according to (9), the increase in computational complexity is $O(L^3)$ when the number of extension points L increases by 1. But the computational complexity of adaptively increasing collocation points is related to the different basis functions. For the basis with compact support property such as Daubechies wavelet and Shannon wavelet, the value of L is impossible to be infinite. For Haar scaling function which has no smoothness property, L can be taken as 0 at most since it does not need to be extended. For Faber-Schauder wavelet, L can be taken as 1 at most. For Daubechies wavelet, L can be taken as different values according to the order of vanishing moments, but it must be finite. For the wavelets without compact support property, L can take value dynamically, such as Shannon wavelet. The computational complexity of increasing points is mainly dependent on the basis function of itself.

5. Numerical Experiments

5.1. Dynamic Choice of the Sparse Grid Points. In order to test the adaptability of the sparse grid approach proposed in this paper, the Faber-Schauder and Shannon scaling function, the autocorrelation function of Daubechies scaling functions, are taken as the basis, respectively. Faber-Schauder scaling function has first-order derivative and Daubechies scaling function has second derivative, so both of the dynamic choice schemes will be tested.

Example 1. Burger equation with Dirichlet boundary conditions.

As a test problem for the numerical algorithm described in the previous section, we will consider Burgers equation as follows:

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = \frac{1}{\mathrm{Re}}\frac{\partial^2 u}{\partial x^2}, \quad x \in [0, 2] \tag{51}$$

with initial and boundary conditions

$$\begin{aligned} u(x, 0) &= \sin(\pi x), \\ u(0, t) &= u(2, t) = 0, \end{aligned} \tag{52}$$

where t represents the time and Re denotes the Reynolds number. With the increasing of the value of Re, the solution develops into a saw-tooth wave at the origin point.

Table 1: Condition number of the Fokker-Planck equation.

j	Re	$\tau = 0.0001$			$\tau = 0.00001$		
		$L = 0$	$L = 1$	$L = 2$	$L = 0$	$L = 1$	$L = 2$
4	10	8.9742	6.3786	12.1637	1.56319	1.9723	2.3335
	100	91.8021	49.1826	186.9260	9.5434	11.9168	19.2971
	1000	1891.2	512.7634	3856.3	403.3223	148.8976	812.9812
7	10	40.3968	32.2256	82.2844	41.2311	4.3834	7.2188
	100	813.5472	199.1953	1245.7	81.7719	49.5538	173.4967
	1000	21987.0000	2426.8	39004	3688.30	699.4512	6917.4
10	10	298.5375	145.7761	663.4654	20.6612	18.4523	40.1116
	100	7821.7000	1217.8	14346	698.5623	212.9856	1274.6
	1000	194670	13887	523830	38421	3974.4	79248

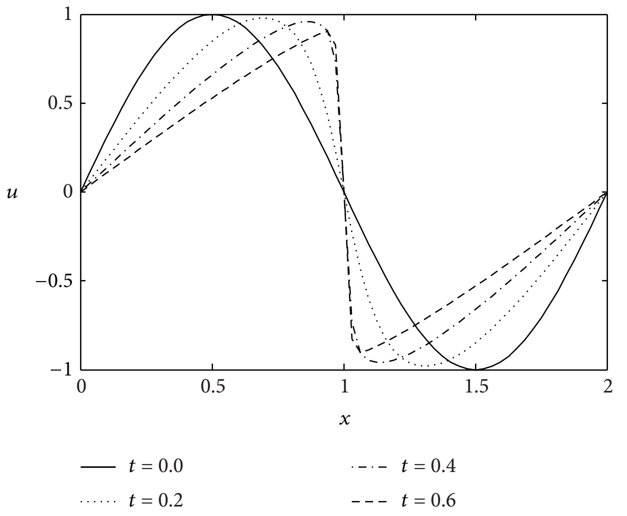


Figure 1: Analytical solutions of the Burgers equation at different times ($t = 0, 0.4, 0.6$).

The gradient at the origin reaches its maximum value. Therefore, the performance of a numerical method is often judged by its ability to resolve the large gradient region that develops in the solution, which is shown in Figure 1.

Using the difference coefficient to approximate the partial differential operator $\partial u/\partial t$, the Burgers equation becomes

$$-\frac{1}{\mathrm{Re}}\frac{d^2u(x,m.\Delta t)}{dx^2}+u(x,(m-1)\Delta t)\cdot\frac{du(x,m\Delta t)}{dx}$$
$$+\frac{1}{\Delta t}[u(x,m\Delta t)-u(x,(m-1)\Delta t)]=0, \qquad (53)$$
$$u(0,m\Delta t)=u(2,m\Delta t)=0, \quad m=1,2,\ldots.$$

According to the virtual displacement theory, the variational form of the Burgers equation can be represented as:

$$\int_0^2 \varepsilon\frac{du(x,m\cdot\Delta t)}{dx}\cdot\frac{dv(x)}{dx}dx$$
$$+\int_0^2\left[u(x,(m-1)\Delta t)\frac{du(x,m\cdot\Delta t)}{dt}\right.$$
$$\left.+\frac{1}{\Delta t}u(x,m\cdot\Delta t)\right]v(x)d(x)$$
$$=\int_0^2\frac{1}{\Delta t}u(x,(m-1)\Delta t)vdx. \qquad (54)$$

This can be solved by means of (35)-(36).

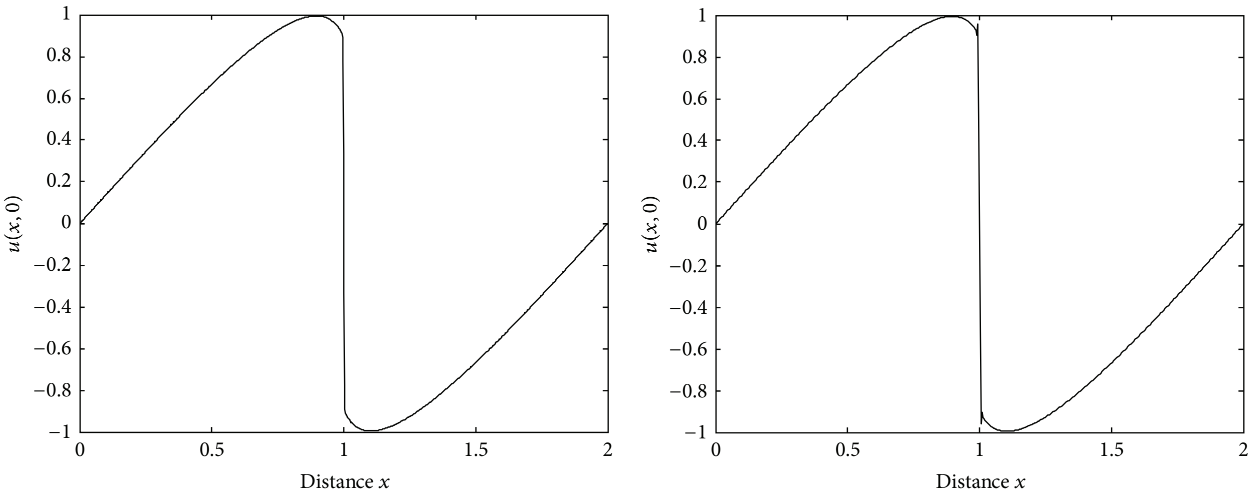
In the experiments, the Reynolds number Re = 1000, and the time step $\tau = 0.001$.

The numerical results showed in Figure 2 are obtained by the finite difference method. As the amount of the even discrete points is taken as 512, the Gibbs phenomena appeared at $x = 1$ where exists a steep slope in the solution (Figure 2(b)). Increasing the discrete points can restrict the Gibbs phenomena (Figure 2(a)).

Figure 3 illustrates the performance of the sparse solution method on this example by comparison of the sparse solution and the true solution produced using a standard fully resolved method (finite difference method).

With the increasing of the parameter t, the gradient of the solution at the point $x = 1$ becomes more and more large. Any of the Faber-Schauder scaling function and the auto-correlation function of the Daubechies scaling function is taken as the basis functions, the sparse method can capture the steep slope appeared in the solution effectively. This shows that more and more grid points concentrate around the point $x = 1$. However, the maximum of the gradient at $x = 1$ appeared as $t = 0.4$, the 1024 coefficients used in the true solution, only 64 with Faber-Schauder basis and 152 with Daubecies basis, are retained in the sparse solution (about 6.25% and 14.84%), respectively. Begin with $t = 0.4$, the gradient value of the solution at $x = 1$ becomes smaller and smaller with the increasing of t. The amount of the sparse grid points also decreases with the increasing of t accordingly. This is illustrated in Figure 3. The adaptability of the proposed sparse method is helpful to improve the efficiency and the calculation precision of the algorithm.

Besides, we also noticed that the condition number of Burger equation from Table 1 varies with the change of j, Re, and the time step τ. In fact, the condition number relates closely with the sparse grid points. As Re and j are smaller, the steep gradient will not appear in the solution and the grid



(a) $t = 0.4$ 1024 grid points

(b) $t = 0.4$ 512 grid points

FIGURE 2: Numerical solution obtained by the finite difference method.

TABLE 2: Dynamic L and the iteration times at the same L value ($j = 5$, $T = 0.2$, $\tau = 0.01$).

L	3	2	4	2	3	2	4	2	3	2	4	2	3
Iteration times	2	1	2	1	1	1	3	1	1	1	3	1	1

TABLE 3: Dynamic L and the iteration times at the same L value ($j = 5$, $T = 0.2$, $\tau = 0.00001$).

L	6	5	4	5	4
Iteration times	15	14	1	1	19968

points are sparse. In this case, the condition number is smaller and will not destroy the numerical precision apparently. On the contrary, if the Re, j, and the time step are larger, the steep slope appearing in the solution and the error brought from the larger time step will bring more grid points adaptively into the algorithm. This will deduce that the condition number becomes larger. It has been mentioned in Section 4.2 that the larger condition number can decrease the calculation precision greatly. Table 1 shows that the condition number ($L = 2$) increases more rapidly than $L = 1$ with the increase of j and Re. This also can be illustrated in Figure 4.

Figure 4 illustrates that the external grid points change with the development of t dynamically. As $t \le 0.04$ (Figure 4(a)), the solution function is smooth and the condition number is smaller. The approach can take more external grid points dynamically to improve the precision. As $t > 0.04$ (Figure 4(b)), the steep slope is appearing in the solution and the condition number is increasing. In this case, the increase of the external grid points cannot improve the precision anymore. In fact, this explained the reason why we construct the dynamic choice scheme of external grid points to some extent.

5.2. Comparison between the Dynamic Choice Scheme and the Wavelet Collocation Method

Example 2. Consider the Heat equation

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial^2 x} + e^{x+2t}, \quad x \in [0,1], \; 0 \le t \le T, \tag{55}$$

with the initial and boundary conditions

$$u(x,0) = e^{x}, \qquad \frac{\partial u(0,t)}{\partial x} = e^{2t}, \qquad \frac{\partial u(1,t)}{\partial x} = e^{1+2t}, \tag{56}$$

where t denotes the time parameter.

Let $I_i(x)$ and $D_i(x)$ denote the interpolation operator and the corresponding derivative; according to the classical collocation approach, the approximating formulation $u_J(x)$ of a function $u(x)$ can be written as:

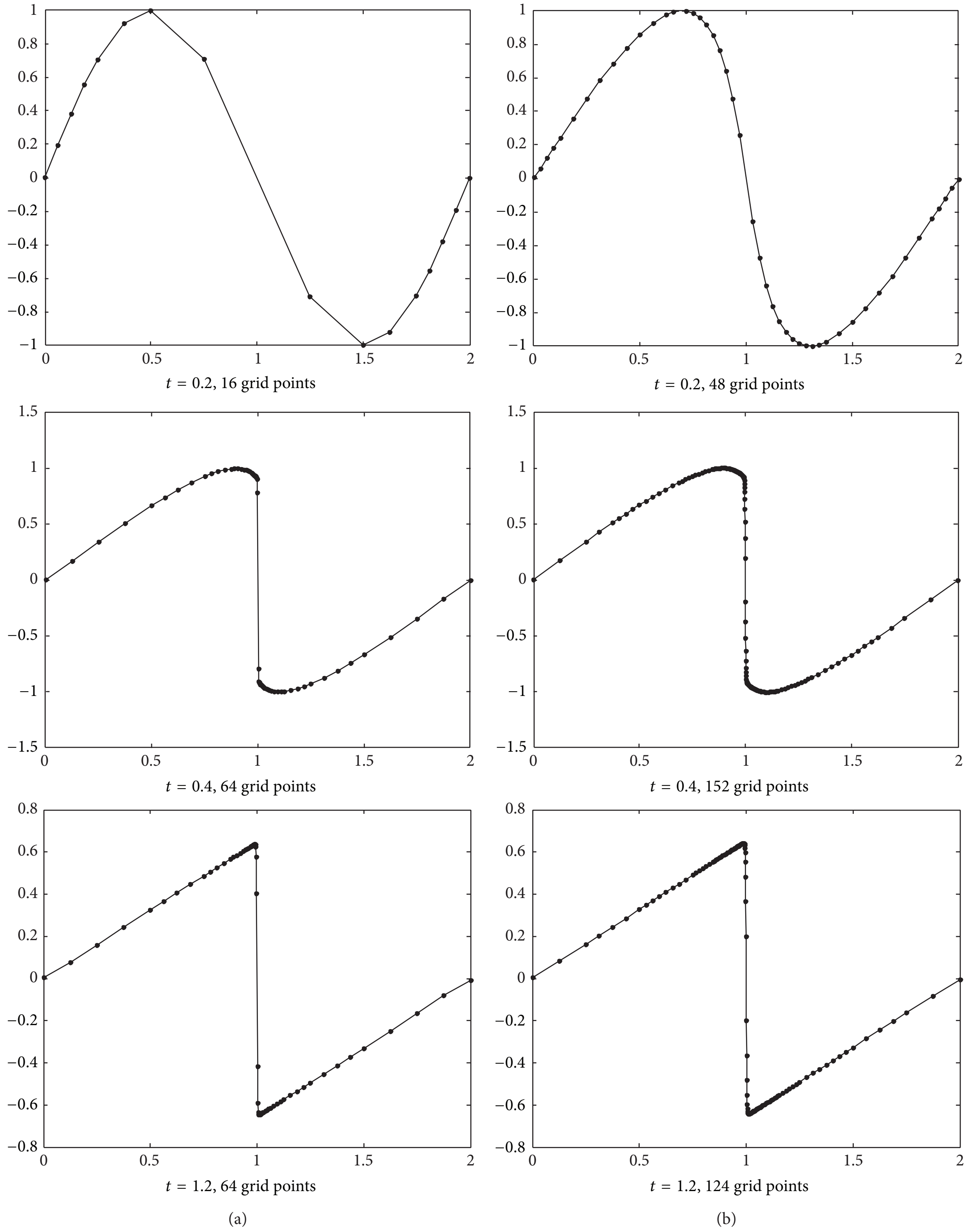
$$u_J(x) = \sum_{i\in Z^c} I_i(x)\, u_{Ji}. \tag{57}$$

Substituting (57) into (55) leads to a system of nonlinear ordinary differential equations as follows:

$$\sum_{n\in Z^c} u_J(x_n,t)\, D_n''(x_k) + \exp(x_k + 2t) = \frac{\partial u_j(x_k,t)}{\partial t}, \tag{58}$$

where $k \in Z_c$. The corresponding vector expression is

$$\frac{\partial}{\partial t} V_J = M_0 V_J + F(t). \tag{59}$$



t = 0.2, 16 grid points

t = 0.2, 48 grid points

t = 0.4, 64 grid points

t = 0.4, 152 grid points

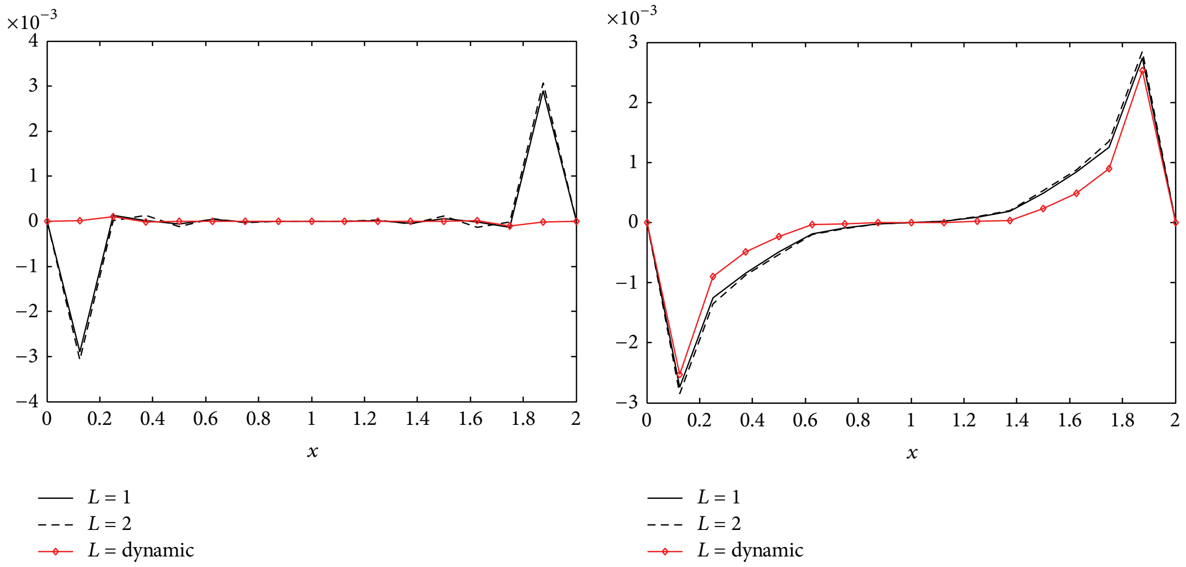
t = 1.2, 64 grid points

t = 1.2, 124 grid points

(a)

(b)

Figure 3: Solution evolution of Burgers equation with Re = 1000.



(a) Dynamic external grid points L ($t = 0$-0.01, $L = 5$; $t = 0.01$-0.02, $L = 3$; $t = 0.02$-0.03, $L = 5$; $t = 0.03$-0.04, $L = 3$)

(b) Dynamic L ($t = 0.04$–0.06, $L = 3$; $t = 0.06$–0.2, $L = 1$)

FIGURE 4: The influence of the condition number to the error (Re = 1000, $j = 7$).

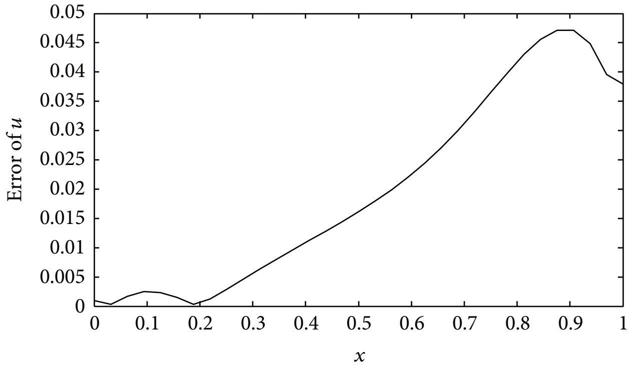


FIGURE 5: Error of the solution with the dynamic grid approach ($T = 0.2$, $\tau = 0.01$, $j = 5$).

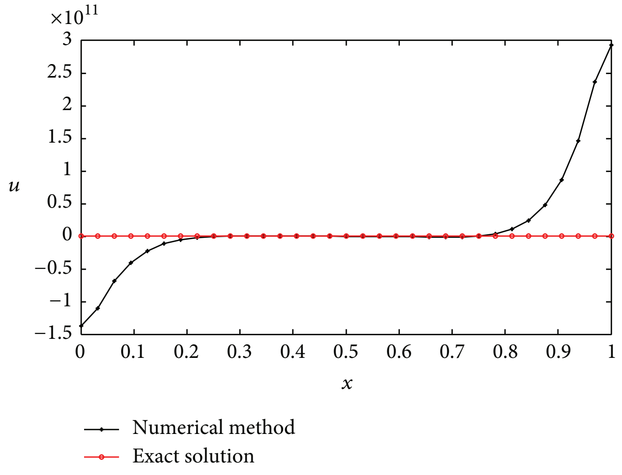


FIGURE 6: Numerical solution with interval wavelet method ($L = 4$, $T = 0.2$, $\tau = 0.01$, $j = 5$).

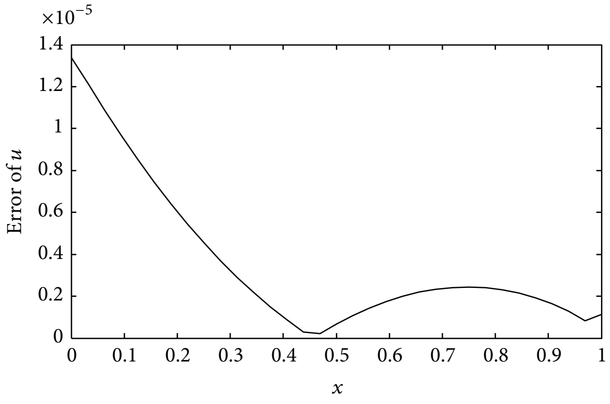


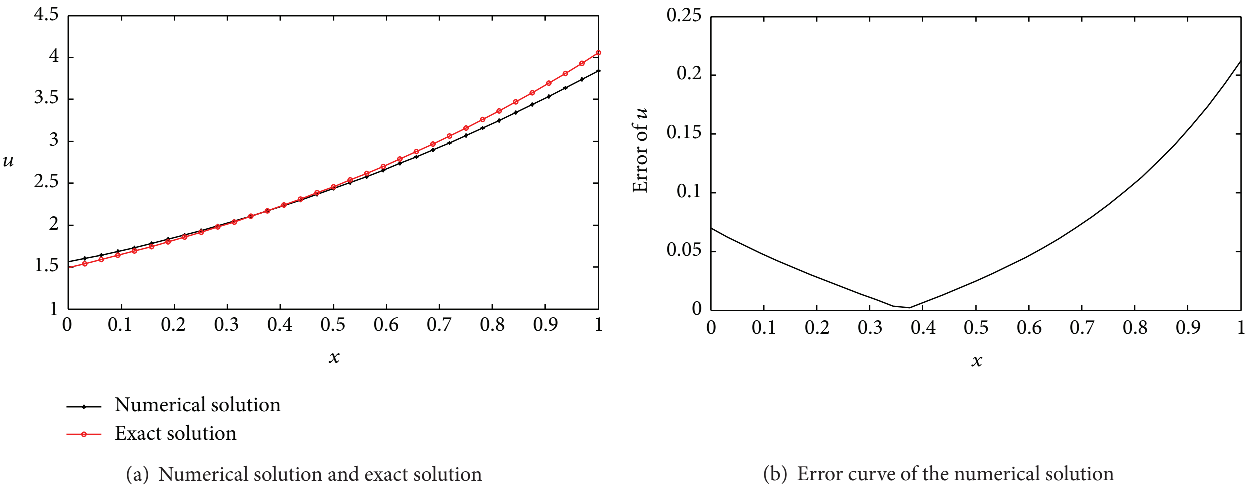
FIGURE 7: Error of the solution with the dynamic sparse grid approach ($j = 5$, $T = 0.2$, and $\tau = 0.00001$).

The corresponding Neumann boundary condition can be expressed as:

$$\begin{gathered} M_1(1,1)\,V_J(1) + \sum_{i=2}^{2^J} M_1(1,i)\,V_J(i) = e^{2t}, \\ M_1\left(2^J,2^J\right)V\left(2^J\right) + \sum_{i=2}^{2^J} M_1\left(2^J,i\right)V_J(i) = e^{1+2t}, \end{gathered} \tag{60}$$

where

$$\begin{gathered} V_J = \left(u_J(x_0,t), u_J(x_1,t), \ldots, u_J(x_{2^J},t)\right)^T, \\ F(t) = \left(\exp(x_0+2t), \exp(x_1+t), \ldots, \exp(x_{2^J}+2t)\right)^T, \end{gathered}$$



(a) Numerical solution and exact solution

(b) Error curve of the numerical solution

FIGURE 8: Numerical solution and error curves with interval wavelet Runge-Kutta method ($L = 4$, $T = 0.2$, $\tau = 0.00001$, $j = 5$).

$$\begin{aligned} M_0(k,n) &= m^0_{k,n} = D''_n(x_k), \quad k,n \in Z^c, \\ M_1(k,n) &= m^1_{k,n} = D'_n(x_k), \quad k,n \in Z^c. \end{aligned} \tag{61}$$

Equations (59)-(60) can be solved by the VIM and PIM. In the following, we will take heat equation as examples to illustrate the effectiveness of the algorithm proposed in this paper. The Shannon scaling function is employed to be the basis function. The exact analytical solution of (55) is $u(x,t) = e^{x+2t}$. Obviously, the solution function's infinite order derivative exists for all x in the definition domain.

(1) Comparison between the Dynamic Choice Scheme and the Static Interval Wavelet. Let $T = 0.2$ and let $\tau = 0.01$. The computational error curve of the dynamic sparse grid approach is shown in Figure 5. The maximum of the absolute error is 0.0471, which occurs near the right boundary. This shows that the bigger gradient of the solution can cause bigger error. The dynamic L and the iteration times at the same L value are shown in Table 2. The value of L varies between 2, 3, and 4, and the iteration times at $L = 4$ is much more than L equaling 2 or 3. So, we take $L = 4$ in the static interval wavelet PIM to solve the heat equation in the same parameters with the dynamic scheme. The numerical solution is shown in Figure 6. Obviously, the error is too big that the algorithm is invalid. There are many reasons that can lead to this result such as the smoothness of the solution and the nonlinear term in the PDEs. As the time step $\tau = 0.00001$, the error curve was shown in Figure 7. The dynamic L and the iteration times at the same L value are shown in Table 3. With the decreasing of the time step, the influence of the nonlinear term on PDEs becomes smaller and smaller. The biggest errors of both dynamic and static interval wavelet PIM are 1.3388×10^{-5}. This shows that the construction of dynamic grid approach is necessary for nonlinear PDEs with Neumann boundary conditions.

(2) Comparison between the VIM and PIM and Runge-Kutta Method for Time-Domain Integration. The numerical solution and error curves with the VIM and PIM and Runge-Kutta method are shown in Figure 8. It is obvious that the calculation precision of VIM and PIM (Figure 5) is better than Runge-Kutta method. It should be pointed out that Runge-Kutta is not sensitive to the time step τ (Table 4) compared with VIM and PIM. One of the most important reasons is that the nonlinear term in PDEs was integrated with explicit format in VIM and PIM, and implicit format was employed in Runge-Kutta method.

6. Conclusions

The multilevel interpolation operator constructed in this paper is independent of the basis. Although Faber-Schauder scaling function has no second-order derivative, it still can be the basis employed in the multiscale interpolation operator to solve the Burgers equation, while only retaining important nodes. The reduced dynamics created by the sparse projection property can dynamically capture the true phenomena exhibited by the solution. This sparse projection amounts to a shrinkage of the coefficients of the updated solution at every time step. Compared with the finite difference method, the retained coefficients are less than 10% in the sparse solution of the Burgers equation.

The dynamic sparse grid approach, which is constructed by combining the multiscale interpolation operator and the variational iteration method, is able to choose both of the internal and external grid points based on the gradient and the smoothness of the solution, the condition number of the PDEs, and the error tolerance dynamically. This property is good suit to the PDEs with Neumann boundary conditions. It can eliminate the boundary effect efficiently. With regard to the accuracy and time complexity of the solution in comparison with those obtained with other algorithms, the dynamic sparse grid approach constructed in this paper is more reasonable. The numerical experiments illustrate that

Table 4: Error comparison between VIMand PIM and Runge-Kutta Method.

τ	0.1	0.01	0.001	0.0001	0.00001
VIM and PIM	0.4176	0.0471	0.0045	4.1368×10^{-4}	1.7610×10^{-5}
Runge-Kutta	0.1366	0.2193	0.2110	0.2122	0.2124

it is necessary to construct the dynamic sparse grid approach for the nonlinear PDEs with Neumann boundary conditions and Dirichlet boundary conditions.

Conflict of Interests

The author declares that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (no. 41171337) and the National High Technology Research and Development Program of China (no. 2012BAD35B02).

References

[1] H. Schaeffer, R. Caflisch, C. Hauck C, and S. Osher, “Sparse dynamics for partial differential equations,” *Proceedings of the National Academy of Sciences of the United States of America*, vol. 110, no. 17, pp. 6634–6639, 2013.

[2] H.-J. Bungartz and M. Griebel, “Sparse grids,” *Acta Numerica*, vol. 13, pp. 147–269, 2004.

[3] S. McWilliam, D. J. Knappett, and C. H. J. Fox, “Numerical solution of the stationary FPK equation using Shannon wavelets,” *Journal of Sound and Vibration*, vol. 232, no. 2, pp. 405–430, 2000.

[4] C. Cattani, “Second order Shannon wavelet approximation of C2-functions,” *UPB Scientific Bulletin A*, vol. 73, no. 3, pp. 73–84, 2011.

[5] C. Cattani, “Harmonic wavelets towards the solution of nonlinear PDE,” *Computers and Mathematics with Applications*, vol. 50, no. 8-9, pp. 1191–1210, 2005.

[6] C. Cattani, “Shannon wavelets for the solution of Integro-differential equations,” *Mathematical Problems in Engineering*, vol. 2010, Article ID 408418, 22 pages, 2010.

[7] C. Cattani, “Fractional calculus and Shannon wavelet,” *Mathematical Problems in Engineering*, vol. 2012, Article ID 502812, 26 pages, 2012.

[8] D. K. Hoffman, G. W. Wei, D. S. Zhang, and D. J. Kouri, “Shannon-Gabor wavelet distributed approximating functional,” *Chemical Physics Letters*, vol. 287, no. 1-2, pp. 119–124, 1998.

[9] M. Inokuti, H. Sekine, and T. Mura, “General use of the Lagrange multiplier in non-linear mathematical physics,” in *Variational Method in Mechanics of Solids*, pp. 156–162, Pergamon Press, Oxford, UK, 1978.

[10] J.-H. He, “Variational iteration method: a kind of non-linear analytical technique: some examples,” *International Journal of Non-Linear Mechanics*, vol. 34, no. 4, pp. 699–708, 1999.

[11] J.-H. He, “Asymptotic methods for solitary solutions and compactons,” *Abstract and Applied Analysis*, vol. 2012, Article ID 916793, 130 pages, 2012.

[12] M. A. Noor and S. T. Mohyud-Din, “Variational iteration methods for solving initial and boundary value problems of Bratu-type,” *Applications and Applied Mathematics*, vol. 3, no. 1, pp. 89–99, 2008.

[13] M. Aslam Noor and S. T. Mohyud-Din, “Variational iteration technique for solving higher order boundary value problems,” *Applied Mathematics and Computation*, vol. 189, no. 2, pp. 1929–1942, 2007.

[14] H. H. Yan, “Adaptive wavelet precise integration method for nonlinear black-scholes model based on variational iteration method,” *Abstract and Applied Analysis*, vol. 2013, Article ID 735919, 6 pages, 2013.

[15] S.-L. Mei and S.-W. Zhang, “Coupling technique of variational iteration and homotopy perturbation methods for nonlinear matrix differential equations,” *Computers and Mathematics with Applications*, vol. 54, no. 7-8, pp. 1092–1100, 2007.

[16] O. V. Vasilyev and S. Paolucci, “A dynamically adaptive multilevel wavelet collocation method for solving partial differential equations in a finite domain,” *Journal of Computational Physics*, vol. 125, no. 2, pp. 498–512, 1996.

[17] S.-L. Mei, H.-L. Lv, and Q. Ma, “Construction of interval wavelet based on restricted variational principle and its application for solving differential equations,” *Mathematical Problems in Engineering*, vol. 2008, Article ID 629253, 14 pages, 2008.

[18] D. L. Donoho, “Interpolating wavelet transforms,” Preprint, Department of Statistics, Stanford University 2. 3, 1992.

[19] S. Bertoluzza, “Adaptive wavelet collocation method for the solution of Burgers equation,” *Transport Theory and Statistical Physics*, vol. 25, no. 3–5, pp. 339–352, 1996.

[20] D. L. Donoho and M. Elad, “Optimally sparse representation in general (nonorthogonal) dictionaries via $\ell 1$ minimization,” *Proceedings of the National Academy of Sciences of the United States of America*, vol. 100, no. 5, pp. 2197–2202, 2003.

[21] D. L. Donoho, “Compressed sensing,” *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[22] O. V. Vasilyev, S. Paolucci, and M. Sen, “A multilevel wavelet collocation method for solving partial differential equations in a finite domain,” *Journal of Computational Physics*, vol. 120, no. 1, pp. 33–47, 1995.

[23] J.-H. He and X.-H. Wu, “Variational iteration method: new development and applications,” *Computers and Mathematics with Applications*, vol. 54, no. 7-8, pp. 881–894, 2007.

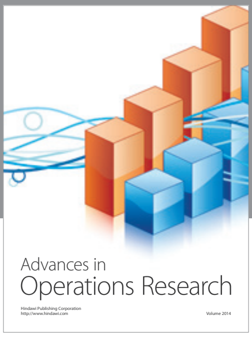
[24] G. C. Wu, “Variational iteration method for q-diference equations of second order,” *Journal of Applied Mathematics*, vol. 2012, Article ID 102850, 5 pages, 2012.

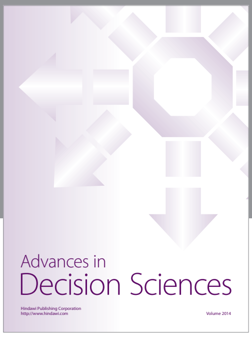
[25] H. Kong and L. L. Huang, “Lagrange multipliers of q-diference equations of third order,” *Communications in Fractional Calculus*, vol. 3, no. 1, pp. 30–33, 2012.

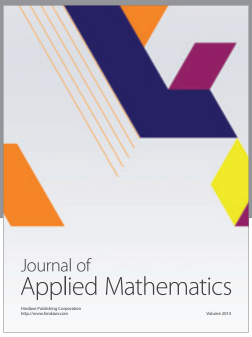
[26] G. C. Wu and D. Baleanu, “Variational iteration method for the Burgers’ flow with fractional derivatives: new Lagrange multipliers,” *Applied Mathematical Modelling*, vol. 37, no. 9, pp. 6183–6190, 2013.

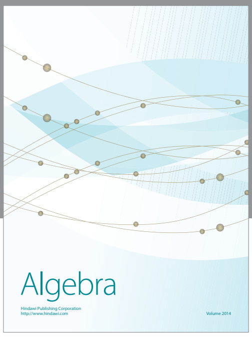
[27] G. C. Wu, "Challenge in the variational iteration method: a new approach to identification of the Lagrange multipliers," *Journal of King Saud University*, vol. 25, pp. 175–178, 2013.

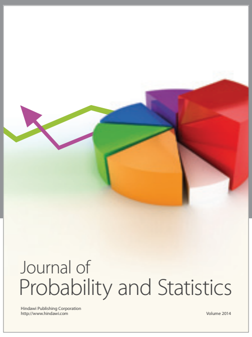
[28] S.-L. Mei, Q.-S. Lu, S.-W. Zhang, and L. Jin, "Adaptive interval wavelet precise integration method for partial differential equations," *Applied Mathematics and Mechanics*, vol. 26, no. 3, pp. 364–371, 2005.


Advances in
Operations Research

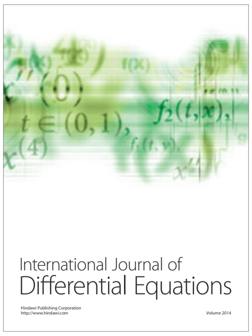

Advances in
Decision Sciences


Journal of
Applied Mathematics

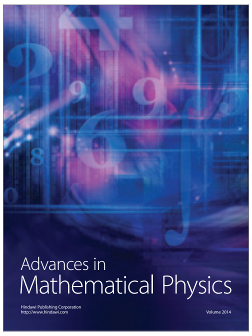

Algebra

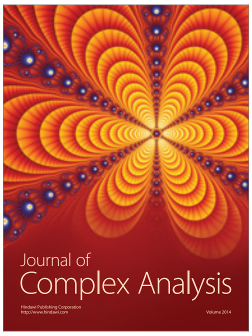

Journal of
Probability and Statistics

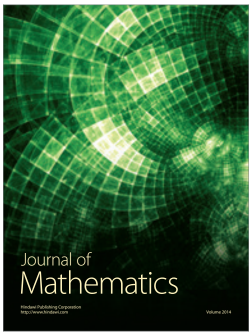
The Scientific
World Journal

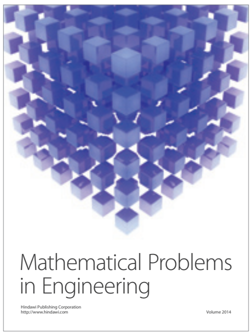

International Journal of
Differential Equations

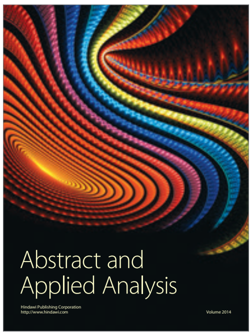
International Journal of
Combinatorics

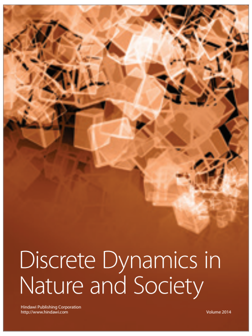

Advances in
Mathematical Physics

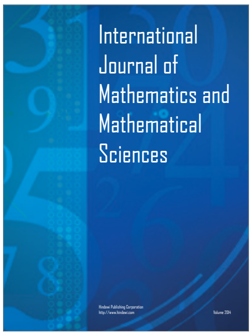

Journal of
Complex Analysis


Journal of
Mathematics


Mathematical Problems
in Engineering

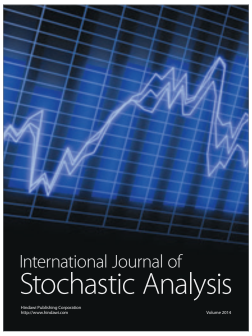

Abstract and
Applied Analysis


Discrete Dynamics in
Nature and Society


International
Journal of
Mathematics and
Mathematical
Sciences

Journal of
Discrete Mathematics

Journal of
Function Spaces


International Journal of
Stochastic Analysis

Journal of
Optimization