On the generalized wavelet-Galerkin method

Zhaochen Yang [c], Shijun Liao [a,b,c,*]

[a] State Key Laboratory of Ocean Engineering, Shanghai 200240, China
[b] Collaborative Innovation Center for Advanced Ship and Deep-Sea Exploration (CISSE), Shanghai 200240, China
[c] School of Naval Architecture, Ocean and Civil Engineering, Shanghai Jiaotong University, Shanghai 200240, China

HIGHLIGHTS

- A new Galerkin method based on the compactly supported wavelets is proposed.
- It is generalized valid even for differential equations with non-constant coefficients.
- Unlike the traditional wavelet-Galerkin method, it is computationally rather efficient.

ARTICLE INFO

ABSTRACT

In the frame of the traditional wavelet-Galerkin method based on the compactly supported wavelets, it is important to calculate the so-called connection coefficients that are some integrals whose integrands involve products of wavelets, their derivatives as well as some known coefficients in considered differential equations. However, even for linear differential equations with non-constant coefficient, the computation of connect coefficients becomes rather time-consuming and often even impossible. In this paper, we propose a generalized wavelet-Galerkin method based on the compactly supported wavelets, which is computationally very efficient even for differential equations with non-constant coefficients, no matter linear or nonlinear problems. Some related mathematical theorems are proved, based on which the basic ideas of the generalized wavelet-Galerkin method are described in details. In addition, some examples are used to illustrate its validity and high efficiency. A nonlinear example shows that the generalized wavelet-Galerkin method is not only valid to solve nonlinear problems, but also possesses the ability to find new solutions of multi-solution problems. This method can be widely applied to various types of both linear and nonlinear differential equations in science and engineering.

© 2017 Elsevier B.V. All rights reserved.

1. Introduction

Wavelet, as a powerful mathematical tool, has been widely applied to solve many problems in science and engineering. Since the breakthrough in 1988 when Daubechies [1,2] constructed an orthogonal, compactly supported wavelet, there has been an increasing interest in wavelet-based methods for ordinary and partial differential equations. Briefly speaking, there exist three kinds of wavelet-based methods: the wavelet finite element method [3–5], the wavelet collocation method [6–8] and the wavelet-Galerkin method [9–13], whereas the wavelet-Galerkin method has gained the widest attention due to its good convergence and stability characteristics [14–16].

---

* Corresponding author at: School of Naval Architecture, Ocean and Civil Engineering, Shanghai Jiaotong University, Shanghai 200240, China.
  E-mail address: sjliao@sjtu.edu.cn (S. Liao).

To obtain convergent, accurate and stable solutions efficiently by the wavelet-Galerkin method, it is of great importance to choose a proper type of wavelet. Generally, there are some main considerations for the choice of wavelet: orthogonality, compact support, interpolation properties as well as high algebraic accuracy and the ability to represent functions at different levels of resolution [17,18]. However, it is well known that compactly supported wavelets, such as Daubechies' wavelet [1,2], Coiflet wavelet [19,20] and the generalized Coiflet-type wavelet [18,21], have no analytical expressions. More seriously, the derivatives of these compactly supported wavelets are highly oscillatory.

Therefore, great contradictions and challenges arise. On the one hand, the wavelet-Galerkin method possesses great advantages in solving differential equations so that these compactly supported wavelets are expected to be used. On the other hand, due to the lack of analytical expressions of these compactly supported wavelets and high oscillation of their derivatives, there has been a great challenge to calculate the connection coefficients [15,17,22], which play an important role in the wavelet-Galerkin method for differential equations. Simply speaking, the connection coefficients are some integrals whose integrands involve products of the wavelet bases, their derivatives as well as other known functions. However, as mentioned above, for compactly supported wavelets, it is difficult and sometimes even unable to calculate these connection coefficients by the traditional numerical methods.

To tackle the problem, a lot of works have been done. Motivated by the successful construction of orthogonal interval wavelets [23,24], Chen et al. [17] developed a method to compute a series of connection coefficients. In 2007, Zhang et al. [25] corrected some errors of this method, and develop better procedures for the computation of the connection coefficients. Similar to Chen et al. [17], Wang [21] developed a high-precision approach to calculate the connection coefficients for the generalized Coiflet-type wavelets.

However, the problem is not completely solved, because these works only provide some algorithms to calculate connection coefficients for special cases. If a differential equation contains some non-constant coefficients, some new problems arise. For example, let us consider a differential equation on the bounded interval [0, 1], which contains a term $\sin(x)\frac{d^n u(x)}{dx^n}$, where $\sin(x)$ is a non-constant coefficient and $u(x)$ is the unknown function to be solved. Suppose that the wavelet base functions are constructed as $\{\varphi_k(x)\}_{k=0}^N$, and an arbitrary function $u(x) \in L^2[0, 1]$ can be approximated by

$$u(x) \approx \sum_{k=0}^N \mu_k \varphi_k(x), \tag{1}$$

where $\mu_k, k = 0, 1, 2, \ldots, N$ are unknown coefficients to be determined. To solve the differential equation, we substitute this approximation into the original equation. As a result, it contains the term

$$\sin(x)\frac{d^n u(x)}{dx^n} \approx \sum_{k=0}^N \mu_k \sin(x)\frac{d^n \varphi_k(x)}{dx^n}. \tag{2}$$

Therefore, when the wavelet-Galerkin method is adopted, it is necessary to calculate the following connection coefficients

$$\int_0^1 \sin(x)\frac{d^n \varphi_k(x)}{dx^n}\varphi_l(x)dx, \tag{3}$$

where $k, l = 0, 1, 2, \ldots, N$. Obviously, it is extremely difficult and even impossible to calculate such kinds of integrals for compactly supported wavelets by known methods yet, especially when $\sin(x)$ is replaced by an arbitrary function $f(x)$. Note that such kind of connection coefficients are dependent upon the function $f(x)$ and thus one had to calculate them for different equations. In theory, it is convenient if all connection coefficients are only dependent upon the wavelet base functions, i.e. independent of differential equations under consideration.

Of course, we can further approximate the coefficient function $\sin(x)$ by

$$\sin(x) \approx \sum_{k=0}^N \alpha_k \varphi_k(x), \tag{4}$$

where $\alpha_k = \int_0^1 \sin(x)\varphi_k(x)dx, k = 0, 1, 2, \ldots, N$, so that the term $\sin(x)\frac{d^n u(x)}{dx^n}$ becomes

$$\sin(x)\frac{d^n u(x)}{dx^n} \approx \sum_{k=0}^N \sum_{l=0}^N \mu_k \alpha_l \frac{d^n \varphi_k(x)}{dx^n}\varphi_l(x). \tag{5}$$

In this way, the connection coefficients are indeed independent of differential equations under consideration. However, it is time-consuming to calculate the connection coefficients

$$\Gamma_{k,l,m}^n = \int_0^1 \frac{d^n \varphi_k(x)}{dx^n}\varphi_l(x)\varphi_m(x)dx, \tag{6}$$

where $k, l, m = 0, 1, 2, \ldots, N$. In addition, it may also cause serious increase of the number of necessary terms, which is another reason that makes the algorithm seriously inefficient. Thus, connection coefficients expressed in the forms of (3) and (6) are called the complicated connection coefficients in this paper.

The problem of the above-mentioned complicated connection coefficients seriously limits the wide applications of the wavelet-Galerkin method for differential equations with non-constant coefficients. Therefore, we are motivated to develop a generalized wavelet-Galerkin method which possesses the following advantages:

(1) The above-mentioned complicated connection coefficients is completely avoided;
(2) All connection coefficients are independent of differential equations under consideration, so that the algorithm is very efficient and adaptive for different boundary value problems no matter what kinds of non-constant coefficients are contained;
(3) It is valid not only for linear differential equations with non-constant coefficients, but also for nonlinear ones;
(4) It can be conveniently extended to high-dimensional boundary value problems.

In this paper, a generalized approach is developed to efficiently handle both linear and nonlinear differential equations with non-constant coefficients in the frame of the wavelet-Galerkin method. The paper is arranged as follows: In Section 2, some properties of the generalized Coiflet-type wavelet are introduced. In Section 3, a few of related mathematical theorems are given and proved. In Section 4, some linear cases with exact solutions are given to illustrate the validity, accuracy and computational efficiency of the generalized wavelet-Galerkin method. In Section 5, a comparatively complicated nonlinear application is presented to show the validity as well as the potential of the generalized wavelet-Galerkin method. In Section 6, some concluding remarks are given.

2. Some properties of the generalized Coiflet-type wavelet

The generalized Coiflet-type wavelet is defined by its scaling function $\varphi(x)$ and wavelet function $\psi(x)$ which possess the following properties [21,26]:

$$(a)\, \varphi(x) = \sum_{k\in\mathbb{Z}} p_k \varphi(2x-k); \tag{7}$$

$$(b)\, \psi(x) = \sum_{k\in\mathbb{Z}} (-1)^k p_{1-k} \psi(2x-k); \tag{8}$$

$$(c)\, M_n = M_1^n, \quad for\ 0 \le n < N; \tag{9}$$

$$(d)\, \int_{-\infty}^{+\infty} x^n \psi(x)dx = 0, \quad for\ 0 \le n < N; \tag{10}$$

$$(e)\, \sum_{k\in\mathbb{Z}} p_k \varphi(x-k) = 1; \tag{11}$$

where p_k is the low-pass filter coefficients [18,27], N is the number of vanishing moment, $M_n = \int_{-\infty}^{+\infty} x^n \varphi(x)dx$ is the nth-order moment of the scaling function and the corresponding supported interval is $supp\,[\varphi(x)] = [0,\ 3N-1]$. Without loss of generality, the generalized Coiflet-type wavelet with $N = 6$, $M_1 = 7$ is used in this paper.

The generalized Coiflet-type wavelet possesses a lot of advantages: orthogonality, compact support, quasi-interpolation (not exactly interpolation but with high accuracy) properties as well as high algebraic accuracy and the ability to represent functions at different levels of resolution. Based on these advantages, wavelet basis for L^2-function space is constructed by Wang et al. [21,28], and an arbitrary function $f(x) \in L^2[0, 1]$ can be approximated by

$$f(x) \approx P_x^j f(x) = \sum_{k=0}^{2^j} f\left(\frac{k}{2^j}\right) \varphi_{j,k}(x), \tag{12}$$

in which P_x^j is an projective operator with respect to x, j is the resolution level and $\varphi_{j,k}(x)$ is the wavelet basis defined by

$$
\varphi_{j,k}(x) =
\begin{cases}
\displaystyle\sum_{i=2-3N+M_1}^{-1} T_{0,k}\left(\frac{i}{2^j}\right) \varphi(2^j x - i + M_1) + \varphi(2^j x - k + M_1), \\
\qquad\qquad\qquad\qquad\qquad\qquad k \in [0, 3], \\
\varphi(2^j x - k + M_1), \\
\qquad\qquad\qquad\qquad\qquad\qquad k \in [4, 2^j - 4], \\
\displaystyle\sum_{i=2^j+1}^{2^j-1+M_1} T_{1,2^j-k}\left(\frac{i}{2^j}\right) \varphi(2^j x - i + M_1) + \varphi(2^j x - k + M_1), \\
\qquad\qquad\qquad\qquad\qquad\qquad k \in [2^j - 3, 2^j],
\end{cases}
\tag{13}
$$

where

$$T_{0,k}(x) = \sum_{i=0}^{3} \frac{p_{0,i,k}}{i!} x^i, \quad T_{1,k}(x) = \sum_{i=0}^{3} \frac{p_{1,i,k}}{i!} (x-1)^i, \tag{14}$$

and coefficients $p_{0,i,k}$ and $p_{1,i,k}$ are assigned as

$$\mathbf{P_0} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -11/6 & 32 & -3/2 & 1/3 \\ 22 & -5 & 42 & -1 \\ -1 & 32 & -3 & 1 \end{bmatrix}, \tag{15}$$

$$\mathbf{P_1} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 11/62 & -3 & 3/22 & -1/3 \\ 22 & -5 & 42 & -1 \\ 12 & -3 & 32 & -1 \end{bmatrix} \tag{16}$$

through the relations $\mathbf{P_0} = \{2^{-ij} p_{0,i,k}\}$ and $\mathbf{P_1} = \{2^{-ij} p_{1,i,k}\}$ with $i, k = 0, 1, 2, 3$ and j being the resolution level.

Actually, $p_{0,i,k}$ and $p_{1,i,k}$ represent the difference coefficients of $f^{(i)}(0)$ and $f^{(i)}(1)$, respectively. Let $h = \frac{1}{2^j}$, then according to Taylor's theorem, $f(kh)$ and $f(1-kh)$ can be expanded into

$$f(kh) = f(0) + f'(0)(kh) + \frac{f''(0)}{2!}(kh)^2 + \frac{f'''(0)}{3!}(kh)^3 + o(h^3), \tag{17}$$

and

$$f(1-kh) = f(1) + f'(1)(-kh) + \frac{f''(1)}{2!}(-kh)^2 + \frac{f'''(1)}{3!}(-kh)^3 + o(h^3), \tag{18}$$

respectively. Therefore, the coefficients $p_{0,i,k}$ and $p_{1,i,k}$ can be obtained by the following equations

$$\mathbf{d_0} = \mathbf{P_0}\mathbf{f_0}, \quad \mathbf{d_1} = \mathbf{P_1}\mathbf{f_1}, \tag{19}$$

where $\mathbf{d_0} = \{f(0), hf'(0), h^2 f''(0), h^3 f'''(0)\}^T$, $\mathbf{f_0} = \{f(0), f(h), f(2h), f(3h)\}^T$, $\mathbf{d_1} = \{f(1), hf'(1), h^2 f''(1), h^3 f'''(1)\}^T$, and $\mathbf{f_1} = \{f(1), f(1-h), f(1-2h), f(1-3h)\}^T$, $f(kh)$ and $f(1-kh)$ are approximated by Eqs. (17) and (18) without the high-order term $o(h^3)$.

Obviously, the derivative of the function can be approximated by

$$\frac{d^n f(x)}{dx^n} \approx \frac{d^n P_x^j f(x)}{dx^n} = \sum_{k=0}^{2^j} f\left(\frac{k}{2^j}\right) \frac{d^n \varphi_{j,k}(x)}{dx^n}, \tag{20}$$

where the values of $\frac{d^n \varphi_{j,k}(x)}{dx^n}$ on equinoxes can be calculated through the method suggested by Wang [21].

Of course, for boundary value problems, there are always some boundary conditions to be satisfied. So some minor modifications need to be made on the approximation (12) according to different boundary conditions. For example,

(a) When the value of $f(0)$ or $f(1)$ is given, the boundary value condition can be satisfied by directly substituting these two given values into wavelet approximation (12).
(b) When $f'(0) = 0$, the wavelet basis should be modified into

$$\tilde{\varphi}_{j,k}(x) = \varphi_{j,k}(x)|_{p_{0,1,k} \to 0}. \tag{21}$$

(c) When $f'(0) = 0$ and $f''(0) = 0$, the wavelet basis should be modified into

$$\tilde{\tilde{\varphi}}_{j,k}(x) = \varphi_{j,k}(x)|_{p_{0,1,k} \to 0, \; p_{0,2,k} \to 0}. \tag{22}$$

For more details on the modifications, please refer to [18].

According to [18,29], the error estimation of the wavelet approximation (12) is

$$\left\| \frac{d^n f(x)}{dx^n} - \frac{d^n P_x^j f(x)}{dx^n} \right\|_{L^2} \leq C 2^{-j(N-n)}, \tag{23}$$

where C is a positive constant that depends only on the function $f(x)$ and low-pass filter coefficients p_k, N is the number of vanishing moment, and $0 \leq n < N$.

According to the theory of multiresolution analysis [30], two-dimensional wavelet basis can be directly extended by the tensor products of one-dimensional wavelet basis. Therefore, for an arbitrary function $f(x, y) \in L^2[0, 1]^2$, we have the following approximations

$$f(x, y) \approx P_x^j P_y^j f(x, y) = \sum_{k=0}^{2^j} \sum_{l=0}^{2^j} f\left(\frac{k}{2^j}, \frac{l}{2^j}\right) \varphi_{j,k}(x)\varphi_{j,l}(y) \tag{24}$$

and

$$\frac{\partial^{m+n}f(x, y)}{\partial x^m \partial y^n} \approx \frac{\partial^{m+n}P_x^j P_y^j f(x, y)}{\partial x^m \partial y^n} = \sum_{k=0}^{2^j} \sum_{l=0}^{2^j} f\left(\frac{k}{2^j}, \frac{l}{2^j}\right) \frac{d^m \varphi_{j,k}(x)}{dx^m} \frac{d^n \varphi_{j,l}(y)}{dy^n}. \tag{25}$$

The error estimation can be derived from Eq. (23) as

$$\left\| \frac{\partial^{m+n}f(x, y)}{\partial x^m \partial y^n} - \frac{\partial^{m+n}P_x^j P_y^j f(x, y)}{\partial x^m \partial y^n} \right\|_{L^2} \le C2^{-\min\{j(N-m), j(N-n)\}}. \tag{26}$$

3. Some related theorems

In this section, some fundamental theorems are proposed and proved. Based on these theorems, the generalized wavelet-Galerkin method for linear boundary value problems governed by differential equations with non-constant coefficients is deduced. For simplicity, we mainly describe the basic ideas as well as the theorems with linear cases, while all of them are similar for nonlinear ones.

3.1. Some basic theorems and definitions

Theorem 1. *Assuming that I is a bounded open interval, n is a positive integer, $f(x) \in C^n(I)$ and $u(x) \in C^n(I)$, then in the interval I it holds*

$$f(x)\frac{d^n u(x)}{dx^n} = \sum_{k=0}^{n}(-1)^k \binom{n}{k} \frac{d^{n-k}}{dx^{n-k}} \left[\frac{d^k f(x)}{dx^k} u(x) \right], \tag{27}$$

where the 0th-order derivative of arbitrary function $g(x) \in C(I)$ is defined as itself, i.e.

$$\frac{d^0 g(x)}{dx^0} \triangleq g(x). \tag{28}$$

Proof. Considering $n \in \mathbb{Z}^+$, it is equivalent to dealing with the following open statement:

$$S(n) : f(x)\frac{d^n u(x)}{dx^n} = \sum_{k=0}^{n}(-1)^k \binom{n}{k} \frac{d^{n-k}}{dx^{n-k}} \left[\frac{d^k f(x)}{dx^k} u(x) \right],$$

for arbitrary function $f(x), u(x) \in C^n(I)$.

Here, we prove this theorem by the method of mathematical induction.

Basic step: we start with the statement $S(1)$ and find that

$$S(1) : f(x)\frac{du(x)}{dx} = \frac{d}{dx}[f(x)u(x)] - \frac{df(x)}{dx}u(x),$$

$$= \sum_{k=0}^{1}(-1)^k \binom{1}{k} \frac{d^{1-k}}{dx^{1-k}} \left[\frac{d^k f(x)}{dx^k} u(x) \right],$$

holds for arbitrary $f(x), u(x) \in C^1(I)$, say, $S(1)$ is true.

Inductive step: Assuming the truth of $S(k)$ for some particular $k \in \mathbb{Z}^+$, we have

$$S(k) : f(x)\frac{d^k u(x)}{dx^k} = \sum_{i=0}^{k}(-1)^i \binom{k}{i} \frac{d^{k-i}}{dx^{k-i}} \left[\frac{d^i f(x)}{dx^i} u(x) \right],$$

for arbitrary $f(x), u(x) \in C^k(I)$.

Then, for arbitrary $f(x), u(x) \in C^{k+1}(I)$, it is true that $\frac{df(x)}{dx} \in C^k(I)$ and $\frac{d^k f(x)}{dx^k} \in C^1(I)$. Therefore,

$$f(x)\frac{d^{k+1}u(x)}{dx^{k+1}} = f(x)\frac{d}{dx}\left[\frac{d^k u(x)}{dx^k} \right]$$

$$= \frac{d}{dx}\left[f(x)\frac{d^k u(x)}{dx^k} \right] - \frac{df(x)}{dx}\frac{d^k u(x)}{dx^k}$$

$$= \frac{d}{dx}\left\{ \sum_{i=0}^{k}(-1)^i \binom{k}{i} \frac{d^{k-i}}{dx^{k-i}} \left[\frac{d^i f(x)}{dx^i} u(x) \right] \right\}$$

$$- \sum_{i=0}^{k}(-1)^i \binom{k}{i} \frac{d^{k-i}}{dx^{k-i}} \left[\frac{d^i}{dx^i} \left(\frac{df(x)}{dx} \right) u(x) \right].$$

Using the induction hypothesis, we have

$$f(x)\frac{d^{k+1}u(x)}{dx^{k+1}} = \sum_{i=0}^{k}(-1)^i\binom{k}{i}\frac{d^{k+1-i}}{dx^{k+1-i}}\left[\frac{d^if(x)}{dx^i}u(x)\right]$$

$$-\sum_{i=0}^{k}(-1)^i\binom{k}{i}\frac{d^{k-i}}{dx^{k-i}}\left[\frac{d^{i+1}f(x)}{dx^{i+1}}u(x)\right]$$

$$= \sum_{i=0}^{k}(-1)^i\binom{k}{i}\frac{d^{k+1-i}}{dx^{k+1-i}}\left[\frac{d^if(x)}{dx^i}u(x)\right]$$

$$+\sum_{j=1}^{k+1}(-1)^j\binom{k}{j-1}\frac{d^{k+1-j}}{dx^{k+1-j}}\left[\frac{d^jf(x)}{dx^j}u(x)\right]$$

$$= (-1)^0\binom{k}{0}\frac{d^{k+1}}{dx^{k+1}}\left[\frac{d^0f(x)}{dx^0}u(x)\right]+(-1)^{k+1}\binom{k}{k}\frac{d^0}{dx^0}\left[\frac{d^{k+1}f(x)}{dx^{k+1}}u(x)\right]$$

$$+\sum_{i=1}^{k}(-1)^i\left[\binom{k}{i}+\binom{k}{i-1}\right]\frac{d^{k+1-i}}{dx^{k+1-i}}\left[\frac{d^if(x)}{dx^i}u(x)\right].$$

Since $\binom{k+1}{0} = \binom{k}{0} = \binom{k+1}{k+1} = \binom{k}{k} = 1$ and $\binom{k}{i}+\binom{k}{i-1} = \binom{k+1}{i}$, we have

$$f(x)\frac{d^{k+1}u(x)}{dx^{k+1}} = \sum_{i=0}^{k+1}(-1)^i\binom{k+1}{i}\frac{d^{k+1-i}}{dx^{k+1-i}}\left[\frac{d^if(x)}{dx^i}u(x)\right],$$

which indicates that the statement $S(k + 1)$ is true.

Hence, for any particular $k \in \mathbb{Z}^+$, it follows that $S(k) \Longrightarrow S(k+1)$. Considering the truth of $S(1)$, it leads to the conclusion that the statement $S(n)$ is true for all $n \in \mathbb{Z}^+$.

This ends the proof.

Theorem 2. *Let I denote a bounded open interval. The linear ordinary differential equation*

$$\sum_{n=0}^{N}a_n(x)\frac{d^nu(x)}{dx^n} = r(x), \quad x \in I \tag{29}$$

where $a_n(x) \in C^n(I)$ and $r(x)$ are known functions can be rewritten in the form

$$\sum_{n=0}^{N}\frac{d^n\,(b_n(x)u(x))}{dx^n} = r(x), \quad x \in I, \tag{30}$$

where

$$b_n(x) = \sum_{r=n}^{N}(-1)^{r-n}\binom{r}{n}\frac{d^{r-n}a_r(x)}{dx^{r-n}}. \tag{31}$$

Proof. According to Theorem 1, for each $a_n(x) \in C^n(I)$, $n = 0, 1, 2, \ldots, N$, we have

$$a_n(x)\frac{d^nu(x)}{dx^n} = \sum_{k=0}^{n}(-1)^k\binom{n}{k}\frac{d^{n-k}}{dx^{n-k}}\left[\frac{d^ka_n(x)}{dx^k}u(x)\right]$$

$$= \sum_{k=0}^{n}(-1)^{n-k}\binom{n}{k}\frac{d^k}{dx^k}\left[\frac{d^{n-k}a_n(x)}{dx^{n-k}}u(x)\right].$$

Therefore,

$$\sum_{n=0}^{N}a_n(x)\frac{d^nu(x)}{dx^n} = \sum_{n=0}^{N}\sum_{k=0}^{n}(-1)^{n-k}\binom{n}{k}\frac{d^k}{dx^k}\left[\frac{d^{n-k}a_n(x)}{dx^{n-k}}u(x)\right]$$

$$= \sum_{k=0}^{N}\sum_{n=k}^{N}(-1)^{n-k}\binom{n}{k}\frac{d^k}{dx^k}\left[\frac{d^{n-k}a_n(x)}{dx^{n-k}}u(x)\right].$$

Rewriting the summation index n as r, and k as n on the right side, we have

$$
\sum_{n=0}^{N} a_n(x)\frac{d^n u(x)}{dx^n} = \sum_{n=0}^{N}\sum_{r=n}^{N}(-1)^{r-n}\binom{r}{n}\frac{d^n}{dx^n}\left[\frac{d^{r-n}a_r(x)}{dx^{r-n}}u(x)\right]
$$

$$
= \sum_{n=0}^{N}\frac{d^n}{dx^n}\left\{\left[\sum_{r=n}^{N}(-1)^{r-n}\binom{r}{n}\frac{d^{r-n}a_r(x)}{dx^{r-n}}\right]u(x)\right\}.
$$

This ends the proof.

Theorem 3. *Assuming that Ω is a bounded open region, $f(x,y) \in C^{m+n}(\Omega)$ and $u(x,y) \in C^{m+n}(\Omega)$, then in the region Ω it holds*

$$
f(x,y)\frac{\partial^{m+n}u(x,y)}{\partial x^m \partial y^n} = \sum_{i=0}^{m}\sum_{j=0}^{n}(-1)^{i+j}\binom{m}{i}\binom{n}{j}\frac{\partial^{(m-i)+(n-j)}}{\partial x^{m-i}\partial y^{n-j}}\left[\frac{\partial^{i+j}f(x,y)}{\partial x^i \partial y^j}u(x,y)\right], \tag{32}
$$

where the derivation operators are defined by

$$
\begin{cases}
\dfrac{\partial^{m+0}g(x,y)}{\partial x^m \partial y^0} \triangleq \dfrac{\partial^m g(x,y)}{\partial x^m} \\[2mm]
\dfrac{\partial^{0+n}g(x,y)}{\partial x^0 \partial y^n} \triangleq \dfrac{\partial^n g(x,y)}{\partial y^n} \\[2mm]
\dfrac{\partial^{0+0}g(x,y)}{\partial x^0 \partial y^0} \triangleq g(x,y).
\end{cases} \tag{33}
$$

Theorem 4. *Let Ω denote a bounded open region. The linear partial differential equation*

$$
\sum_{m=0}^{M}\sum_{n=0}^{N} a_{m,n}(x,y)\frac{\partial^{m+n}u(x,y)}{\partial x^m \partial y^n} = r(x,y), \quad (x,y) \in \Omega \tag{34}
$$

where $a_{m,n}(x,y) \in C^{m+n}(\Omega)$ and $r(x,y)$ are known functions can be rewritten in the form

$$
\sum_{m=0}^{M}\sum_{n=0}^{N}\frac{\partial^{m+n}\big(b_{m,n}(x,y)u(x,y)\big)}{\partial x^m \partial y^n} = r(x,y), \quad (x,y) \in \Omega, \tag{35}
$$

where

$$
b_{m,n}(x,y) = \sum_{r=m}^{M}\sum_{s=n}^{N}(-1)^{(r-m)+(s-n)}\binom{r}{m}\binom{s}{n}\frac{\partial^{(r-m)+(s-n)}a_{r,s}(x,y)}{\partial x^{r-m}\partial y^{s-n}}. \tag{36}
$$

Then, let us define some operators that will be used later.

Definition 1 (*Hadamard/Schur Product*)**.** Let $\mathbf{A} = \{a_{kl}\}_{m\times n}$, $\mathbf{B} = \{b_{kl}\}_{m\times n} \in \mathbb{R}^{m\times n}$ (vectors when $n = 1$), then the Hadamard/Schur product of matrixes or vectors is defined by

$$
\mathbf{A} \circ \mathbf{B} = \{c_{kl} = a_{kl}b_{kl}\}_{m\times n}. \tag{37}
$$

Definition 2 (*Column-dot Product*)**.** Let $\mathbf{A} = \{a_{kl}\}_{m\times n} \in \mathbb{R}^{m\times n}$, $\mathbf{v} = \{v_l\}_{n\times 1} \in \mathbb{R}^{n\times 1}$, then the column-dot product is defined by

$$
\mathbf{A} \odot \mathbf{v} = \{c_{kl} = a_{kl}v_l\}_{m\times n}. \tag{38}
$$

3.2. The generalized wavelet-Galerkin method

Using the above-mentioned theorems, we propose here the generalized wavelet-Galerkin method. Without loss of generality, let us consider the linear ordinary and partial differential equations (29) and (34), which are difficult to solve by means of the traditional wavelet-Galerkin method if $a_n(x)$ and $a_{m,n}(x,y)$ are non-constants. However, according to the basic theorems, they can be rewritten as Eqs. (30) and (35), which are more convenient to solve by means of the wavelet-Galerkin method, as shown below. In this way, the connection coefficients in the frame of the wavelet-Galerkin method are independent of differential equations. This is the basic idea of the generalized wavelet-Galerkin method, which is valid for both linear and nonlinear differential equations.

3.2.1. Approach for one-dimensional BVPs

Let us consider the generalized linear boundary value problem (BVP) on the interval [0, 1] governed by the ordinary differential equation

$$\sum_{n=0}^{N} \frac{d^n (b_n(x)u(x))}{dx^n} = r(x), \ \ 0 < x < 1 \tag{39}$$

with some boundary conditions, where $r(x)$ and $b_n(x)$ are known functions.

Instead of directly approximating the unknown function $u(x)$ by wavelet basis, in the frame of the generalized wavelet-Galerkin method we take $b_n(x)u(x)$ as a whole function. Using wavelet approximation (12) and considering the boundary conditions, $b_n(x)u(x)$ can be approximated by

$$f_n(x)u(x) \approx P_x^j [f_n(x)u(x)] = \sum_{k=0}^{2^j} f_n \left(\frac{k}{2^j} \right) u \left(\frac{k}{2^j} \right) \varphi_{j,k}(x) \tag{40}$$

for $n = 1, 2, \ldots, N$, and $r(x)$ can be approximated by

$$r(x) \approx P_x^j [r(x)] = \sum_{k=0}^{2^j} r \left(\frac{k}{2^j} \right) \varphi_{j,k}(x). \tag{41}$$

Substituting Eqs. (40) and (41) into Eq. (39), we have

$$\sum_{n=0}^{N} \sum_{k=0}^{2^j} b_n \left(\frac{k}{2^j} \right) u \left(\frac{k}{2^j} \right) \frac{d^n \varphi_{j,k}(x)}{dx^n} = \sum_{k=0}^{2^j} r \left(\frac{k}{2^j} \right) \varphi_{j,k}(x). \tag{42}$$

Multiplying both sides of Eq. (42) by $\varphi_{j,l}(x)$ where $l = 0, 1, 2, \ldots, 2^j$ (or from 1 to $2^j - 1$ if values of $u(x)$ at $x = 0$ and $x = 1$ are given), respectively, and integrating over the interval [0, 1], it yields

$$\sum_{n=0}^{N} \mathbf{A_n}^T (\mathbf{b_n} \circ \mathbf{u}) = \mathbf{A_0}^T \mathbf{r}, \tag{43}$$

where

$$\begin{cases} \mathbf{A_n} = \left\{ a_{kl} = \Gamma_{k,l}^{j,n} = \int_0^1 \frac{d^n \varphi_{j,k}(\xi)}{d\xi^n} \varphi_{j,l}(\xi)d\xi \right\}_{(2^j+1) \times (2^j+1)} \\ \mathbf{b_n} = \left\{ b_k = b_n \left(\frac{k}{2^j} \right) \right\}_{(2^j+1) \times 1} \\ \mathbf{u} = \left\{ u_k = u \left(\frac{k}{2^j} \right) \right\}_{(2^j+1) \times 1} \\ \mathbf{r} = \left\{ r_k = r \left(\frac{k}{2^j} \right) \right\}_{(2^j+1) \times 1} . \end{cases} \tag{44}$$

Eq. (43) can be rewritten as

$$\left[\sum_{n=0}^{N} \mathbf{A_n}^T \odot \mathbf{b_n} \right] \mathbf{u} = \mathbf{A_0}^T \mathbf{r}. \tag{45}$$

In practice, there might be some minor differences due to the different boundary conditions, but the basic form of the wavelet-Galerkin equation does not change. Eq. (45) indicates that the connection coefficients are independent of $b_n(x)$ at all so that we only need to calculate the connection coefficients in the simplest form

$$\Gamma_{k,l}^{j,n} = \int_0^1 \frac{d^n \varphi_{j,k}(\xi)}{d\xi^n} \varphi_{j,l}(\xi)d\xi, \tag{46}$$

which is easy to gain in the frame of the wavelet-Galerkin method. Especially, since the connection coefficients are independent of differential equations, it is easy to build a database of the connection coefficients for data sharing. This helps standardize the calculation procedure and can greatly increase the computational efficiency, as shown later by examples. Note that, according to Theorem 2, any a linear differential equation (29) can be rewritten in the form of (39). So, this approach is valid in general.

More importantly, in the frame of the generalized wavelet-Galerkin method, similar wavelet-Galerkin equations can be deduced for nonlinear problems, which is in the form of

$$\left[\sum_{n=0}^{N} \mathbf{A_n}^T \odot \mathbf{b_n}\right] \mathbf{u} + \left[\sum_{n=0}^{N} \mathbf{A_n}^T \odot \mathbf{c_n}\right] \mathcal{N}[\mathbf{u}] = \mathbf{A_0}^T \mathbf{r}. \tag{47}$$

in which $\mathbf{c_n}$ and $\mathcal{N}[\mathbf{u}]$ are derived from the nonlinear part with non-constant coefficients. An application will be given later.

3.2.2. Approach for two-dimensional BVPs

Then, let us consider the generalized linear BVP on the region $[0, 1]^2$ governed by the partial differential equation

$$\sum_{m=0}^{M} \sum_{n=0}^{N} \frac{\partial^{m+n} \left(b_{m,n}(x, y)u(x, y)\right)}{\partial x^m \partial y^n} = r(x, y), \ 0 < x, y < 1 \tag{48}$$

with some boundary conditions.

Using wavelet approximation (24) and considering the boundary conditions, we similarly have the following equation

$$\sum_{k=0}^{2^j} \sum_{l=0}^{2^j} \left[u\left(\frac{k}{2^j}, \frac{k}{2^j}\right) \sum_{m=0}^{M} \sum_{n=0}^{N} b_{m,n}\left(\frac{k}{2^j}, \frac{l}{2^j}\right) \frac{d^m \varphi_{j,k}(x)}{dx^m} \frac{d^n \varphi_{j,l}(y)}{dy^n} \right]$$
$$= \sum_{k=0}^{2^j} \sum_{l=0}^{2^j} r\left(\frac{k}{2^j}, \frac{l}{2^j}\right) \varphi_{j,k}(x)\varphi_{j,l}(y), \tag{49}$$

which leads to

$$\sum_{m=0}^{M} \sum_{n=0}^{N} \mathbf{A_m}^T \left(\mathbf{B_{mn}} \circ \mathbf{U}\right) \mathbf{A_n} = \mathbf{A_0}^T \mathbf{R} \mathbf{A_0} \tag{50}$$

where

$$\begin{cases} \mathbf{A_n} = \left\{ a_{kl} = \Gamma_{k,l}^{j,n} = \int_0^1 \frac{d^n \varphi_{j,k}(\xi)}{dx^n} \varphi_{j,l}(\xi)d\xi \right\}_{(2^j+1)\times(2^j+1)} \\[4mm] \mathbf{B_{mn}} = \left\{ b_{kl} = b_{m,n}\left(\frac{k}{2^j}, \frac{l}{2^j}\right) \right\}_{(2^j+1)\times(2^j+1)} \\[4mm] \mathbf{U} = \left\{ u_{kl} = u\left(\frac{k}{2^j}, \frac{l}{2^j}\right) \right\}_{(2^j+1)\times(2^j+1)} \\[4mm] \mathbf{R} = \left\{ r_{kl} = r\left(\frac{k}{2^j}, \frac{l}{2^j}\right) \right\}_{(2^j+1)\times(2^j+1)}. \end{cases} \tag{51}$$

Taking $rvec(\cdot)$ operator on both sides and using related operation properties, we obtain the following wavelet-Galerkin equation

$$\left[\sum_{m=0}^{M} \sum_{n=0}^{N} (\mathbf{A_m} \otimes \mathbf{A_n})^T \odot rvec(\mathbf{B_{mn}})\right] rvec(\mathbf{U}) = (\mathbf{A_0} \otimes \mathbf{A_0})^T rvec(\mathbf{R}), \tag{52}$$

where $\otimes$ denotes the Kronecker tensor product of matrixes and $rvec(\cdot)$ pulls a matrix into a vector row by row.

It can be found that the connection coefficients are independent of the coefficients $b_{m,n}(x, y)$ of the differential equation at all. In other words, the connection coefficients are independent of differential equations. Note also that, according to Theorem 4, any two-dimensional differential equation (34) can be rewritten as (48). So, this approach is rather general. Also, in the frame of the generalized wavelet-Galerkin method, similar wavelet-Galerkin equations can be deduced for nonlinear problems. In addition, both linear and nonlinear boundary value problems in higher dimensions can also be solved in a similar way by means of the generalized wavelet-Galerkin method.

In addition, as there is no truncation in Theorems 1–4, the error of the generalized wavelet-Galerkin method can be estimated by Eqs. (23) and (26) only if m, n are replaced by the order of the differential equations.

4. Illustrative linear examples

To verify the validity and the computational efficiency of the generalized wavelet-Galerkin method, some examples of boundary value problems governed by linear ordinary and partial differential equations with different kinds of non-constant coefficients are solved. All results given in this section are obtained using a desktop (DELL Inspiron 3847, Intel (R) Core (TM) i5-4460 CPU@ 3.20 GHz).

4.1. An ODE problem with complicated coefficients

At first, let us consider the following ordinary differential equation (ODE) with complicated non-constant coefficients

$$x^2 \frac{d^2 u}{dx^2} + \sin \pi x \frac{du}{dx} + e^x u = \left(e^x + \pi \cos \pi x - \pi^2 x^2 \right) \sin \pi x, \tag{53}$$

subject to the boundary conditions

$$u(0) = u(1) = 0, \tag{54}$$

which has the exact solution

$$u(x) = \sin \pi x. \tag{55}$$

According to Theorem 2, Eq. (53) can be rewritten as

$$\frac{d^2 \left(b_2(x)u(x) \right)}{dx^2} + \frac{d \left(b_1(x)u(x) \right)}{dx} + b_0(x)u(x) = r(x), \tag{56}$$

where

$$\begin{cases} b_0(x) = e^x - \pi \cos \pi x + 2, \\ b_1(x) = \sin \pi x - 4x, \\ b_2(x) = x^2, \\ r(x) = \left(e^x + \pi \cos \pi x - \pi^2 x^2 \right) \sin \pi x. \end{cases} \tag{57}$$

Considering the boundary condition $u(0) = 0$ and the fact that $x = 0$ is a first order zero point of $b_1(x)$ and a second order zero point of $b_2(x)$, we have

$$\begin{cases} b_0(x)u(x)|_{x=0} = 0, \\ b_1(x)u(x)|_{x=0} = 0, \quad \dfrac{d \left(b_1(x)u(x) \right)}{dx} \bigg|_{x=0} = 0, \\ b_2(x)u(x)|_{x=0} = 0, \quad \dfrac{d \left(b_2(x)u(x) \right)}{dx} \bigg|_{x=0} = 0, \quad \dfrac{d^2 \left(b_2(x)u(x) \right)}{dx^2} \bigg|_{x=0} = 0. \end{cases} \tag{58}$$

So wavelet basis of $b_1(x)u(x)$ and $b_2(x)u(x)$ should be modified into $\tilde{\varphi}_{j,k}(x)$ and $\tilde{\tilde{\varphi}}_{j,k}(x)$, respectively, which are defined by Eqs. (21) and (22). Substituting these modified expressions into the wavelet approximation (42), we have

$$\sum_{k=0}^{2^j} u \left(\frac{k}{2^j} \right) \left[b_2 \left(\frac{k}{2^j} \right) \frac{d^2 \tilde{\tilde{\varphi}}_{j,k}(x)}{dx^2} + b_1 \left(\frac{k}{2^j} \right) \frac{d \tilde{\varphi}_{j,k}(x)}{dx} + b_0 \left(\frac{k}{2^j} \right) \varphi_{j,k}(x) \right] = 0, \tag{59}$$

where the boundary conditions can be embedded by directly substituting $u \left(0/2^j \right) = 0$ and $u \left(2^j/2^j \right) = 0$ into it.

Multiplying both sides of Eq. (59) by $\varphi_{j,l}(x)$ $(l = 1, 2, \ldots, 2^j - 1)$ respectively, and integrating over the interval [0, 1], we have a linear algebraic equation

$$\left[\sum_{n=0}^{2} \mathbf{A_n}^T \odot \mathbf{b_n} \right] \mathbf{u} = 0 \tag{60}$$

where

$$\begin{cases} \mathbf{A_0} = \left\{ a_{kl} = \Gamma_{k,l}^{j,0} = \displaystyle\int_0^1 \varphi_{j,k}(\xi)\varphi_{j,l}(\xi)d\xi \right\}_{(2^j-1)\times(2^j+1)} \\[2mm] \mathbf{A_1} = \left\{ a_{kl} = \tilde{\Gamma}_{k,l}^{j,1} = \displaystyle\int_0^1 \frac{d\tilde{\varphi}_{j,k}(\xi)}{d\xi}\varphi_{j,l}(\xi)d\xi \right\}_{(2^j-1)\times(2^j+1)} \\[2mm] \mathbf{A_2} = \left\{ a_{kl} = \tilde{\tilde{\Gamma}}_{k,l}^{j,2} = \displaystyle\int_0^1 \frac{d^2 \tilde{\tilde{\varphi}}_{j,k}(\xi)}{d\xi^2}\varphi_{j,l}(\xi)d\xi \right\}_{(2^j-1)\times(2^j+1)} \\[2mm] \mathbf{b_n} = \left\{ b_k = b_n \left(\dfrac{k}{2^j} \right) \right\}_{(2^j+1)\times 1} \\[2mm] \mathbf{u} = \left\{ u_k = u \left(\dfrac{k}{2^j} \right) \right\}_{(2^j+1)\times 1}. \end{cases} \tag{61}$$
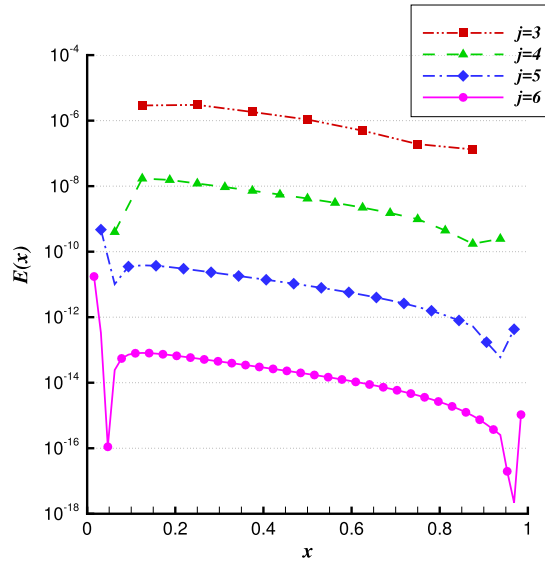
Fig. 1. The error distribution of the wavelet solution at different resolution levels for Eqs. (53) and (54). Square symbol with double dot-dashed line: $j = 3$; delta symbol with dashed line: $j = 4$; diamond symbol with dash-dotted line: $j = 5$; circle symbol with solid line: $j = 6$.

Table 1
The averaged square error and CPU time of wavelet solutions with different resolution levels of Eqs. (53) and (54).

j, resolution level	ErrSQ	CPU time (s)
3	1.1×10^{-6}	0.39
4	4.7×10^{-9}	0.41
5	2.6×10^{-11}	0.55
6	8.0×10^{-14}	1.01

Separating the known parts that are relevant to $u(0/2^j)$ and $u(2^j/2^j)$ from the left side, it is very easy to solve the linear algebraic equation (60).

Define the averaged square error *ErrSQ* and the error distribution $E(x)$ as follows:

$$ErrSQ = \frac{1}{2^j + 1} \sum_{k=0}^{2^j} \left[u_w(\frac{k}{2^j}) - u_e(\frac{k}{2^j}) \right]^2, \tag{62}$$

$$E(x) = \left| u_w(x) - u_e(x) \right|^2, \quad 0 < x < 1, \tag{63}$$

where $u_e(x)$ denotes the exact solution and $u_w(x)$ denotes the wavelet solution.

The error distribution of the wavelet solution at different resolution levels are shown in Fig. 1. The error distribution shows that the wavelet solutions agree very well with the exact solution. This verifies the validity of the generalized wavelet-Galerkin method. Also, by raising the resolution level j, the accuracy of the solution can be greatly improved. Table 1 clearly shows that by raising the resolution level j from 4 to 6, the averaged error quickly decreases from 10^{-6} to 10^{-14}. More importantly, the advantages of this method are revealed. Note that the connection coefficients $\Gamma_{k,l}^{j,n}$ are independent of the non-constant coefficients x^2, $\sin \pi x$ and e^x in Eq. (53), in other words, the connection coefficients are independent of the differential equations at all. Therefore, it is very easy to build a database for data sharing, which of course makes this method very efficient and adaptive. For example, in case of the resolution level $j = 4$, only 0.39 s CPU time is used to gain the wavelet solution. Even in case of the resolution level $j = 6$, it just takes 1.01 s CPU time. So, this approach is computationally rather efficient.

4.2. An ODE problem with special functions as non-constant coefficients

Let us further consider a BVP which contains non-constant coefficients expressed by special functions

$$\Gamma(x + 1)\frac{d^2u(x)}{dx^2} + \left[\Gamma(x + 1) + \Gamma'(x + 1) \right] \frac{du(x)}{dx} = -\Gamma'(x + 1) - \Gamma''(x + 1), \tag{64}$$
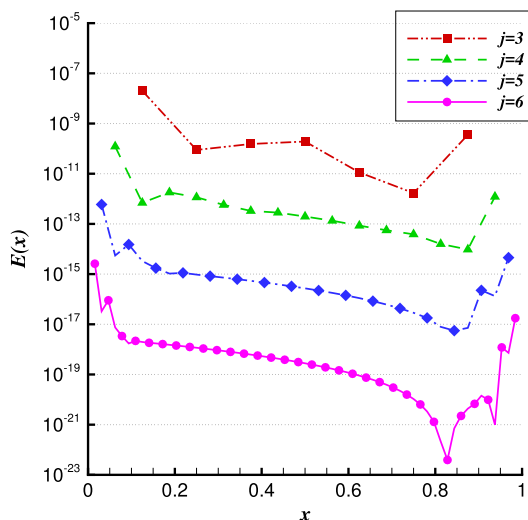
Fig. 2. The error distribution of the wavelet solution at different resolution levels for Eqs. (64) and (65). Square symbol with double dot-dashed line: $j = 3$; delta symbol with dashed line: $j = 4$; diamond symbol with dash-dotted line: $j = 5$; circle symbol with solid line: $j = 6$.

subject to the boundary conditions

$$u(0) = 0, \ u(1) = 0, \tag{65}$$

which has the exact solution

$$u(x) = \ln \frac{1}{\Gamma(x+1)}, \tag{66}$$

where

$$\Gamma(x) = \int_0^{+\infty} \xi^{x-1} e^{-\xi} d\xi. \tag{67}$$

According to Theorem 2, Eq. (64) can be rewritten as

$$\frac{d^2 (b_2(x)u(x))}{dx^2} + \frac{d (b_1(x)u(x))}{dx} + b_0(x)u(x) = r(x), \tag{68}$$

where

$$\begin{cases} b_0(x) = -\Gamma'(x+1), \\ b_1(x) = \Gamma(x+1) - \Gamma'(x+1), \\ b_2(x) = \Gamma(x+1), \\ r(x) = -\Gamma'(x+1) - \Gamma''(x+1). \end{cases} \tag{69}$$

For this problem, as $x = 0$ and $x = 1$ are not zero points of $b_0(x)$, $b_1(x)$ and $b_2(x)$, no modification on the wavelet basis is necessary, so the boundary conditions (65) can be embedded into the wavelet expressions of $b_0(x)u(x)$, $b_1(x)u(x)$ and $b_2(x)u(x)$ by directly substituting $u(0) = u(1) = 0$ into them. Thus, the solution of this boundary value problem can be obtained by substituting these coefficient functions into the generalized wavelet-Galerkin equation (45) and solving it.

The error distribution of the wavelet solutions at different resolution levels are presented in Fig. 2. Obviously, the error is rather small, which shows that the wavelet solutions agree very well with the exact solution. This once again verifies the validity of the generalized wavelet-Galerkin method.

Table 2 presents the averaged square error and the corresponding used CPU time for the wavelet solutions at different resolution levels. Obviously, the generalized wavelet-Galerkin method possesses very high efficiency. With the resolution level j varying from 3 to 6, all of the wavelet solutions can be obtained in less than one second while the computational accuracy rises significantly. At the resolution level $j = 6$, the averaged square error is rather small, say, 10^{-17}. Moreover, it can be seen in Fig. 2 that almost all the averaged square errors on the whole interval is smaller than 10^{-15} at the resolution level $j = 6$. It means that the error is rather small on the whole interval. These results not only verify the validity of the generalized wavelet-Galerkin method, but also illustrate the good convergence property, high accuracy and high efficiency of computation as well as the great potential of this method.

Table 2
The averaged square error and CPU time of wavelet solutions at different resolution levels of Eqs. (64) and (65).

j, resolution level	ErrSQ	CPU time (s)
3	2.3×10^{-09}	0.31
4	7.6×10^{-12}	0.37
5	1.9×10^{-14}	0.54
6	4.3×10^{-17}	0.95

4.3. A PDE problem

Furthermore, let us consider a two-dimensional boundary value problem governed by

$$\sqrt{x^2 + y^2 + 1}\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial x^2}\right) + \frac{1}{\sqrt{x^2 + y^2 + 1}}\left(x\frac{\partial u}{\partial x} + y\frac{\partial u}{\partial y} - 2u\right) = 0, \tag{70}$$

subject to the boundary conditions

$$\begin{cases} u(0, y) = \sqrt{1 + y^2}, \ u(1, y) = \sqrt{2 + y^2}, \\ u(x, 0) = \sqrt{1 + x^2}, \ u(x, 1) = \sqrt{2 + x^2}, \end{cases} \tag{71}$$

which has the exact solution

$$u(x, y) = \sqrt{x^2 + y^2 + 1}. \tag{72}$$

According to Theorem 4, Eq. (70) can be rewritten as

$$\frac{\partial^2 (b_{20}u)}{\partial x^2} + \frac{\partial^2 (b_{02}u)}{\partial y^2} + \frac{\partial^2 (b_{11}u)}{\partial x \partial y} + \frac{\partial (b_{10}u)}{\partial x} + \frac{\partial (b_{01}u)}{\partial y} + b_{00}u = 0, \tag{73}$$

where

$$\begin{cases} b_{20}(x, y) = \sqrt{x^2 + y^2 + 1}, \\ b_{02}(x, y) = \sqrt{x^2 + y^2 + 1}, \\ b_{11}(x, y) = 0, \\ b_{10}(x, y) = -\dfrac{x}{\sqrt{x^2 + y^2 + 1}}, \\ b_{01}(x, y) = -\dfrac{y}{\sqrt{x^2 + y^2 + 1}}, \\ b_{00}(x, y) = -\dfrac{2}{\sqrt{x^2 + y^2 + 1}}. \end{cases} \tag{74}$$

For this problem, Eq. (73) can be approximated by

$$\sum_{k=0}^{2^j}\sum_{l=0}^{2^j} u\left(\frac{k}{2^j}, \frac{l}{2^j}\right)\left[b_{20}\left(\frac{k}{2^j}, \frac{l}{2^j}\right)\frac{d^2\varphi_{j,k}(x)}{dx^2}\varphi_{j,l}(y) + b_{02}\left(\frac{k}{2^j}, \frac{l}{2^j}\right)\varphi_{j,k}(x)\frac{d^2\varphi_{j,l}(y)}{dy^2}\right]$$

$$+ \sum_{k=0}^{2^j}\sum_{l=0}^{2^j} u\left(\frac{k}{2^j}, \frac{l}{2^j}\right)\left[b_{10}\left(\frac{k}{2^j}, \frac{l}{2^j}\right)\frac{d\varphi_{j,k}(y)}{dx}\varphi_{j,l}(y) + b_{01}\left(\frac{k}{2^j}, \frac{l}{2^j}\right)\varphi_{j,k}(x)\frac{d\varphi_{j,l}(y)}{dy}\right]$$

$$+ \sum_{k=0}^{2^j}\sum_{l=0}^{2^j} u\left(\frac{k}{2^j}, \frac{l}{2^j}\right)b_{00}\left(\frac{k}{2^j}, \frac{l}{2^j}\right)\varphi_{j,k}(x)\varphi_{j,l}(y) = 0. \tag{75}$$

The boundary conditions (71) can be embedded by directly substituting

$$\begin{cases} u\left(\dfrac{k}{2^j}, 0\right) = \sqrt{1 + \left(\dfrac{k}{2^j}\right)^2}, \ u\left(\dfrac{k}{2^j}, 1\right) = \sqrt{2 + \left(\dfrac{k}{2^j}\right)^2}, \\ u\left(0, \dfrac{l}{2^j}\right) = \sqrt{1 + \left(\dfrac{l}{2^j}\right)^2}, \ u\left(1, \dfrac{l}{2^j}\right) = \sqrt{2 + \left(\dfrac{l}{2^j}\right)^2} \end{cases} \tag{76}$$

into wavelet approximation (75) for $k, l = 0, 1, 2, \ldots, 2^j$.

Table 3
The averaged square error and CPU time of wavelet solutions with different resolution levels of Eqs. (70) and (71).

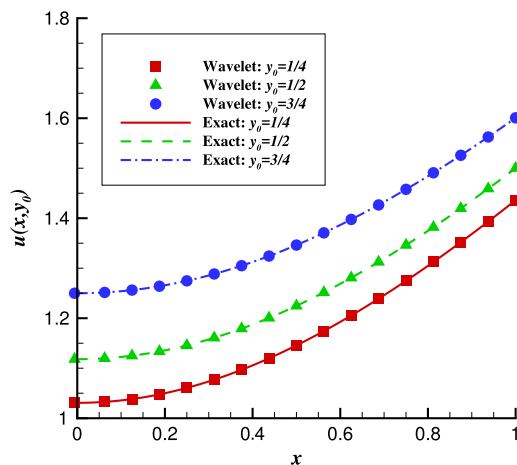| j, resolution level	ErrSQ	CPU time (s)
3	8.7×10^{-19}	0.36
4	7.5×10^{-19}	0.44
5	6.9×10^{-19}	0.74
6	6.6×10^{-19}	3.53

Fig. 3. Comparison of section curves of the wavelet solution at the resolution level $j = 4$ with the exact solution of Eqs. (70) and (71). Lines: exact solutions; square symbol: wavelet solution at $y_0 = 1/4$; delta symbol: wavelet solution at $y_0 = 1/2$; circle symbol: wavelet solution at $y_0 = 3/4$;.

Thus, we have

$$\left[\sum_{m=0}^{2} \sum_{n=0}^{2-m} (\mathbf{A_m} \otimes \mathbf{A_n})^T \odot rvec(\mathbf{B_{mn}}) \right] rvec(\mathbf{U}) = \mathbf{0}, \tag{77}$$

where the matrixes are defined by Eq. (51).

Separating the known parts relevant to the boundary values from the left side and solving this linear algebraic equation, the solution of the two-dimensional BVP is obtained. The three section curves at $y = 1/4$, $y = 1/2$ and $y = 3/4$ at the resolution level $j = 4$ are shown in Fig. 3. The corresponding averaged squared error is only 7.5×10^{-19} as presented in Table 3. Also, Fig. 4 shows that the error is evenly distributed. Obviously, the wavelet approximation agrees very well with the exact solution, which further indicates the validity of the generalized wavelet-Galerkin method in two-dimensional boundary value problems.

5. Application in nonlinear problems

For simplicity, the basic idea of the generalized wavelet-Galerkin method is mainly described by linear cases. However, the significance of the generalized wavelet-Galerkin method extends far beyond solving linear differential equations with non-constant coefficients. Actually, the generalized wavelet-Galerkin method is also a powerful tool to solve complicated nonlinear problems with non-constant coefficients.

To illustrate it more clearly, let us consider a nonlinear BVP which contains non-constant coefficients as follows:

$$\frac{d^2u}{dx^2} - 2(x+1)e^{2x}u\frac{du}{dx} + 2\ln u + \left(\sin \pi x + 2e^x\right)\frac{du}{dx} + (\sin \pi x - 1)u = 0, \tag{78}$$

subject to the boundary conditions

$$u(0) = 1, \ u(1) = \frac{1}{e}, \tag{79}$$

in which $-2(x+1)e^{2x}u\frac{du}{dx}$ is a nonlinear term with non-constant coefficient and $2\ln u$ is a transcendental nonlinear term.

It is known that one exact solution of the nonlinear BVP is

$$u(x) = e^{-x}. \tag{80}$$

However, as it is a complicated nonlinear boundary value problem, there might be more than one solution.
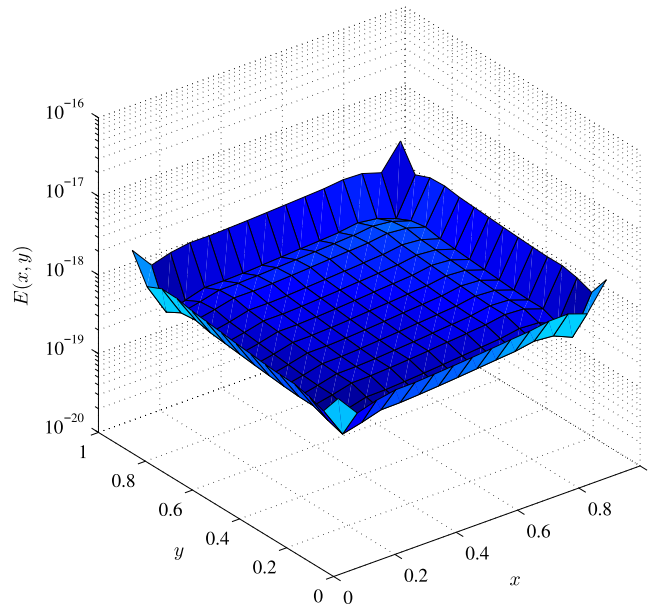
Fig. 4. The error distribution of the wavelet solution at the resolution level $j = 4$ for Eqs. (70) and (71).

Similar to the basic ideas of Theorems 1 and 2, Eq. (78) can be rewritten as

$$\frac{d^2 \left(b_2(x)u\right)}{dx^2} + \frac{d \left(b_1(x)u\right)}{dx} + b_0(x)u + \frac{d \left(c_1(x)u^2\right)}{dx} + c_0(x)u^2 + d_0(x)\ln u = 0, \tag{81}$$

where

$$\begin{cases} b_0(x) = \sin(\pi x) - \pi \cos(\pi x) - 2e^x - 1, \\ b_1(x) = \sin(\pi x) + 2e^x, \\ b_2(x) = 1, \\ c_0(x) = (2x + 3)e^{2x}, \\ c_1(x) = -(x + 1)e^{2x}, \\ d_0(x) = 2. \end{cases} \tag{82}$$

Using the generalized wavelet-Galerkin method, we have:

$$\left[\sum_{n=0}^{2} \mathbf{A_n}^T \odot \mathbf{b_n}\right] \mathbf{u} + \left[\sum_{n=0}^{1} \mathbf{A_n}^T \odot \mathbf{c_n}\right] \mathbf{u} \circ \mathbf{u} + \left[\mathbf{A_0}^T \odot \mathbf{d_0}\right] \ln(\mathbf{u}) = 0. \tag{83}$$

Separating the known parts relevant to the boundary values from the left side and solving this nonlinear algebraic equation, the solutions of the nonlinear BVP can be conveniently obtained.

What is of special interest is that by the generalized wavelet-Galerkin method, more than one solution is obtained, as is shown in Fig. 5. Obviously, wavelet solution I (the lower branch) agrees very well with the exact solution, and the wavelet solution II (the upper branch) is undoubtedly a newly found solution, though the corresponding exact one of which is unknown. It verifies that the generalized wavelet-Galerkin method is a powerful tool to solve nonlinear differential equations with non-constant coefficients. Especially, it is possible to find new solutions of multi-solution problems. It means that the generalized wavelet-Galerkin method possesses great potential for application in nonlinear problems.

To verify the accuracy of the generalized wavelet-Galerkin method for nonlinear cases, the error distribution curves of the wavelet solution I (with known exact solution) at different resolution levels are shown in Fig. 6. It strongly verifies the validity as well as the high accuracy of the generalized wavelet-Galerkin method. In addition, the averaged errors at different resolution levels and the corresponding CPU time are present in Table 4. It shows that with the resolution level raising from 3 to 6, the averaged error decreases from 10^{-8} to 10^{-16}, while the CPU time is less than 2 s, which strongly indicates that the generalized wavelet-Galerkin method possesses both high accuracy and high efficiency not only for linear problems, but also for nonlinear ones. This once again shows the great potential of the generalized wavelet-Galerkin method.
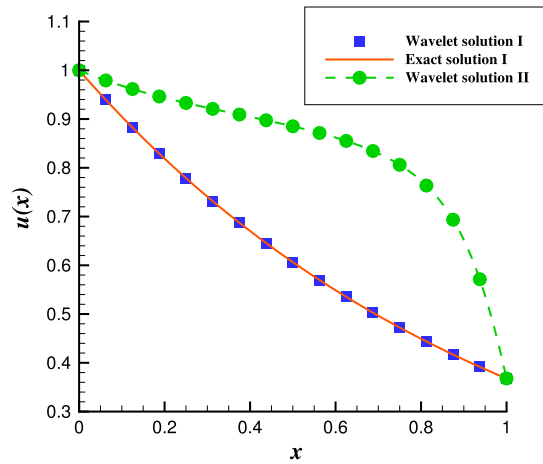
Fig. 5. Comparison of the wavelet solution at the resolution level $j = 4$ with the exact solution, and the newly found wavelet solution for Eqs. (78) and (79). Solid line: the known exact solution; square symbol: wavelet solution I; circle symbol with dashed line: wavelet solution II (newly found).
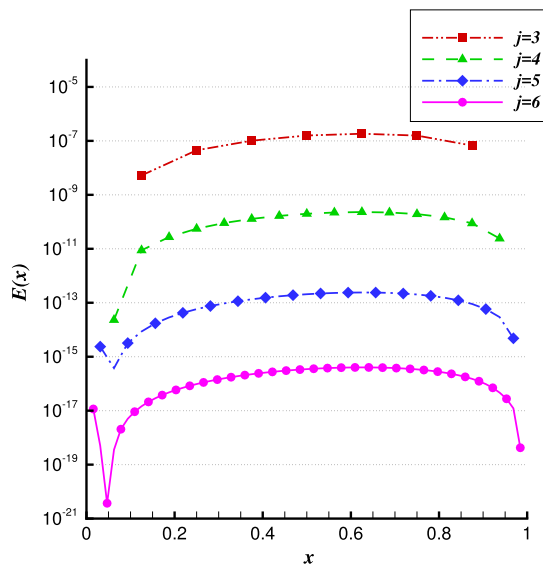

Fig. 6. The error distribution of the wavelet solution I (the lower branch) at different resolution levels for Eqs. (78) and (79). Square symbol with double dot-dashed line: $j = 3$; delta symbol with dashed line: $j = 4$; diamond symbol with dash-dotted line: $j = 5$; circle symbol with solid line: $j = 6$.

Table 4
The averaged square error and CPU time of wavelet solutions with different resolution levels of Eqs. (78) and (79).

j, resolution level	ErrSQ	CPU time (s)
3	8.0×10^{-8}	0.72
4	1.1×10^{-10}	0.80
5	1.2×10^{-13}	0.95
6	1.9×10^{-16}	1.52

6. Concluding remarks and discussions

The computation of connection coefficients plays an important role in the wavelet-Galerkin method. However, differential equations with non-constant coefficients lead to very complicated connection coefficients, which seriously limits the application of the traditional wavelet-Galerkin method based on compactly supported wavelets. In this paper, a generalized wavelet-Galerkin method is proposed to overcome these restrictions of the traditional one.

Some basic theorems are proposed and proved. Based on these theorems, the generalized wavelet-Galerkin method is developed. In the frame of the generalized wavelet-Galerkin method, complicated connection coefficients are always independent of the type of linear differential equations. Therefore, it is easy to build a database of the basic connection coefficients for data sharing, because no new connection coefficient arises no matter what kinds of non-constant coefficients are contained in a differential equation. That makes the generalized wavelet-Galerkin method very efficient and adaptive for various types of boundary value problems. The generalized wavelet-Galerkin equations are derived in a very concise form, which are of great significance both theoretically and practically. Using the generalized Coiflet-type wavelet, several linear boundary value problems governed by ordinary and partial differential equations with non-constant coefficients are solved by the generalized wavelet-Galerkin method. All of these examples verify the validity as well as the advantages of the generalized wavelet-Galerkin method. More importantly, the nonlinear application indicates that the generalized wavelet-Galerkin method not only keeps the same advantages for nonlinear problems, but also possesses the ability to find new solutions for multi-solution problems.

Note that the generalized wavelet-Galerkin method possesses very great potential for extension. Although only one-dimensional and two-dimensional boundary value problems are illustrated in this paper, the generalized wavelet-Galerkin method is valid for higher dimensional problems. Actually, the generalized wavelet-Galerkin equation (52) can be extended to K-dimensional BVPs:

$$
\left[\sum_{n_1=0}^{N_1} \sum_{n_2=0}^{N_2} \cdots \sum_{n_K=0}^{N_K} \left(\bigotimes_{k=1}^{K} \mathbf{A_{n_k}} \right)^T \odot rvec \left(\mathbf{B_{n_1 n_2 \cdots n_K}} \right) \right] rvec \left(\mathbf{U} \right) = \left(\bigotimes_{k=1}^{K} \mathbf{A_0} \right)^T rvec \left(\mathbf{R} \right),
$$
(84)

where $\bigotimes_{k=1}^{K} \mathbf{A_{n_k}} = \mathbf{A_{n_1}} \otimes \mathbf{A_{n_2}} \otimes \cdots \otimes \mathbf{A_{n_K}}$, and the operator $rvec(\cdot)$ pulls a K-dimensional matrix into a vector dimension by dimension. Of course, this equation can be further extended and applied to nonlinear cases. On the other hand, though only the generalized Coiflet-type wavelet is discussed in this paper, theoretically it is possible to extend the generalized wavelet-Galerkin method for other kinds of wavelets with interpolation or quasi-interpolation properties.

Note that Yang and Liao [31,32] successfully combined the wavelet with the homotopy analysis method (HAM) [33–36], an analytic approximation method for highly nonlinear problems developed by Liao [37] in 1992. Unlike perturbation methods, the HAM has nothing to do with the small/large physical parameters. Besides, different from all other approximation techniques, the HAM provides us a convenient way to guarantee the convergence of solution series. As illustrated by Zhong and Liao [38], perturbation methods are often only special cases of the HAM, but the HAM still works well even if perturbation methods fail. More importantly, as a new analytic method, the HAM can provide us something completely new/different [39–41]. So, we can combine the generalized Wavelet-Galerkin method with the HAM to solve highly nonlinear equations.

Acknowledgment

References

[1] I. Daubechies, Orthonormal bases of compactly supported wavelets, Comm. Pure Appl. Math. 41 (7) (1988) 909–996.
[2] I. Daubechies, et al., Ten Lectures on Wavelets, Vol. 61, SIAM, 1992.
[3] J. Ko, A. Kurdila, M. Pilant, A class of finite element methods based on orthonormal, compactly supported wavelets, Comput. Mech. 16 (4) (1995) 235–244.
[4] J.X. Ma, J.J. Xue, S.J. Yang, Z.J. He, A study of the construction and application of a daubechies wavelet-based beam element, Finite Elem. Anal. Des. 39 (10) (2003) 965–975.
[5] J.-G. Han, W.-X. Ren, Y. Huang, A spline wavelet finite-element method in structural mechanics, Internat. J. Numer. Methods Engrg. 66 (1) (2006) 166–190.
[6] S. Bertoluzza, G. Naldi, A wavelet collocation method for the numerical solution of partial differential equations, Appl. Comput. Harmon. Anal. 3 (1) (1996) 1–9.
[7] O.V. Vasilyev, S. Paolucci, A dynamically adaptive multilevel wavelet collocation method for solving partial differential equations in a finite domain, J. Comput. Phys. 125 (2) (1996) 498–512.
[8] O.V. Vasilyev, N.K.-R. Kevlahan, An adaptive multilevel wavelet collocation method for elliptic problems, J. Comput. Phys. 206 (2) (2005) 412–431.
[9] K. Amaratunga, J.R. Williams, S. Qian, J. Weiss, Wavelet–Galerkin solutions for one-dimensional partial differential equations, Internat. J. Numer. Methods Engrg. 37 (16) (1994) 2703–2716.
[10] A. Avudainayagam, C. Vani, Wavelet–Galerkin method for integro–differential equations, Appl. Numer. Math. 32 (3) (2000) 247–254.
[11] J.-Z. Wang, X.-M. Wang, Y.-H. Zhou, A wavelet approach for active-passive vibration control of laminated plates, Acta Mech. Sinica 28 (2) (2012) 520–531.
[12] X.J. Liu, Y.H. Zhou, X.M. Wang, J.Z. Wang, A wavelet method for solving a class of nonlinear boundary value problems, Commun. Nonlinear Sci. Numer. Simul. 18 (8) (2013) 1939–1948.
[13] L. Zhang, J.Z. Wang, Y.-H. Zhou, Wavelet solution for large deflection bending problems of thin rectangular plates, Arch. Appl. Mech. 85 (3) (2015) 355–365.

[14] S.M. Gomes, E. Cortina, Convergence estimates for the wavelet Galerkin method, SIAM J. Numer. Anal. 33 (1) (1996) 149–161.
[15] T. Regińska, L. Eldén, Stability and convergence of the wavelet-Galerkin method for the sideways heat equation, J. Inverse Ill-Posed Probl. 8 (1) (2000) 31–49.
[16] Y.N. Liu, F. Qin, Y.H. Liu, Z.Z. Cen, A daubechies wavelet-based method for elastic problems, Eng. Anal. Bound. Elem. 34 (2) (2010) 114–121.
[17] M.Q. Chen, C. Hwang, Y.P. Shih, The computation of wavelte-Galerkin approximation on a bounded interval, Internat. J. Numer. Methods Engrg. 39 (17) (1996) 2921–2944.
[18] X.J. Liu, A Wavelet Method for Uniformly Solving Nonlinear Problems and Its Application to Quantitative Research on Flexible Structures with Large Deformation (Ph.D. thesis), Lanzhou University, 2014.
[19] G. Beylkin, R. Coifman, V. Rokhlin, Fast wavelet transforms and numerical algorithms I, Comm. Pure Appl. Math. 44 (2) (1991) 141–183.
[20] S.-J. Huang, C.-T. Hsieh, Coiflet wavelet transform applied to inspect power system disturbance-generated signals, IEEE Trans. Aerosp. Electron. Syst. 38 (1) (2002) 204–210.
[21] J.Z. Wang, Generalized Theory and Arithmetic of Orthogonal Wavelets and Applications to Researches of Mechanics Including Piezoelectric Smart Structures (Ph.D. thesis), Lanzhou University, 2001.
[22] F. Bulut, An alternative approach to compute wavelet connection coefficients, Appl. Math. Lett. 53 (2016) 1–9.
[23] C.K. Chui, E. Quak, Wavelets on a bounded interval, in: Numerical Methods in Approximation Theory, Vol. 9, Springer, 1992, pp. 53–75.
[24] A. Cohen, I. Daubechies, P. Vial, Wavelets on the interval and fast wavelet transforms, Appl. Comput. Harmon. Anal. 1 (1) (1993) 54–81.
[25] T.H. Zhang, Y.C. Tian, M.O. Tade, J. Utomo, Comments on: The computation of wavelet-Galerkin approximation on a bounded interval, Internat. J. Numer. Methods Engrg. 72 (2) (2007) 244–251.
[26] X.J. Liu, Y.H. Zhou, L. Zhang, J.Z. Wang, Wavelet solutions of burgers equation with high reynolds numbers, Sci. China Technol. Sci. 57 (7) (2014) 1285–1292.
[27] L. Zhang, X.J. Liu, Y.H. Zhou, J.Z. Wang, Influence of vanishing moments on the accuracy of a modified wavelet Galerkin method for nonlinear boundary value problems, in: 11th International Conference of Numerical Analysis and Applied Mathematics 2013, Vol. 1558, AIP Publishing, 2013, pp. 942–945.
[28] Y.H. Zhou, X.M. Wang, J.Z. Wang, X.J. Liu, A wavelet numerical method for solving nonlinear fractional vibration, diffusion and wave equations, Comput. Model. Eng. Sci. 77 (2) (2011) 137.
[29] W. Sweldens, R. Piessens, Quadrature formulae and asymptotic error expansions for wavelet approximations of smooth functions, SIAM J. Numer. Anal. 31 (4) (1994) 1240–1264.
[30] Y. Meyer, Wavelets and Operators, Vol. 1, Cambridge University Press, 1995.
[31] Z. Yang, S. Liao, A HAM-based wavelet approach for nonlinear ordinary differential equations, Commun. Nonlinear Sci. Numer. Simul. 48 (2017) 439–453.
[32] Z. Yang, S. Liao, A HAM-based wavelet approach for nonlinear partial differential equations: two dimensional Bratu problem as an application, Commun. Nonlinear Sci. Numer. Simul. 53 (2017) 249–262.
[33] S. Liao, On the homotopy analysis method for nonlinear problems, Appl. Math. Comput. 147 (2004) 499–513.
[34] S. Liao, Beyond Perturbation: Introduction to the Homotopy Analysis Method, Chapman & Hall/CRC, Boca Raton, 2003.
[35] S. Liao, Homotopy Analysis Method in Nonlinear Differential Equations, Springer-Verlag, New York, 2011.
[36] K. Vajravelu, R. Van Gorder, Nonlinear Flow Phenomena and Homotopy Analysis: Fluid Flow and Heat Transfer, Springer, Heidelberg, 2012.
[37] S. Liao, Proposed homotopy analysis techniques for the solution of nonlinear problem (Ph.D. thesis), Shanghai Jiao Tong University, 1992.
[38] X. Zhong, S. Liao, Analytic approximations of von Karman plate under arbitrary uniform pressure –equations in integral form, Sci. China-Phys. Mech. Astron. 61 (2018) 014711. http://dx.doi.org/10.1007/s11433-017-9096-1.
[39] D. Xu, Z. Lin, S. Liao, M. Stiassnie, On the steady-state fully resonant progressive waves in water of finite depth, J. Fluid Mech. 710 (2012) 379.
[40] S. Liao, D. Xu, Z. Liu, in: E. Tobisch (Ed.), On the Discovery of Steady-state Resonant Water Waves, in: Elena Tobisch (Ed.), Lecture Notes in Physics, vol. 908, Springer, Heidelberg, 2015, pp. 43–82.
[41] S. Liao, D. Xu, M. Stiassnie, On the steady-state nearly resonant waves, J. Fluid Mech. 794 (2016) 175–199.