

## Research Article

# Windowed Fourier Frames to Approximate Two-Point Boundary Value Problems

**Abdullah Aljouiee and Samir Kumar Bhowmik**

*Department of Mathematics, College of Science, Al Imam Mohammad Ibn Saud Islamic University, Riyadh, Saudi Arabia*

Correspondence should be addressed to Abdullah Aljouiee; joal11@gmail.com

Received 8 September 2014; Revised 6 January 2015; Accepted 6 January 2015

Academic Editor: Maria A. Ragusa

Copyright © 2015 A. Aljouiee and S. K. Bhowmik. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Boundary value problems arise while modeling various physical and engineering reality. In this communication we investigate windowed Fourier frames focusing two-point BVPs. We approximate BVPs using windowed Fourier frames. We present some numerical results to demonstrate the efficiency of such frame functions to approximate BVPs.

## 1. Introduction

Numerical approximation of various ordinary and partial differential equations is of ongoing interest [1–5]. There are several popular schemes to approximate such models. Schemes based on special functions are increasingly popular [6].

The windowed Fourier transform (Gabor transform) has been a widely used tool in signal processing. This technique uses a single window function to Fourier-transform a signal locally. This process is repeated while shifting the window through the real line. This single window shifting and modulation mechanism of the Gabor transform produces some undesirable effects [5]. A set of frame functions, the windowed Fourier frames (WFFs), have been used to serve such purpose as well [5].

In recent time, WFFs have been popularly used for solving partial differential equations (PDE) [7]. In [7], the authors consider an elliptic PDE and develop an efficient solver using a combination of the symbol of the operator and WFFs. They discuss window functions, discretisation, and implementations in detail. They also study the efficiency of using such functions.

The author develops a general recipe for higher order BVPs in [2]. He considers Tchebychev polynomials to approximate BVPs by reducing the order. In fact, the higher order BVPs problems have been converted to first-order BVPs to approximate the problem using global polynomials efficiently. The author exhibits some numerical results to demonstrate the efficiency of the proposed scheme.

Here, in this paper, we focus on approximating the solutions of two-point boundary value problems using windowed Fourier frames. We motivate ourselves to develop a scheme based on windowed frame functions to approximate various operators in a spare way for one-dimensional academic problems (with an aim to approximate higher dimensional operators using WFFs in the near future). One needs a single window function to generate a family of windowed Fourier frame functions. Thus presentation of the operator becomes neat and simple. The advantage of using windowed frame functions is that they have a flexibility to use for various purpose; the windowed Fourier transformation operator generates a spare differential operator which is easy to store; as a result computations become simple (compared to the spectral collocation/global polynomial approximations for the differential operator). The superiority of the technique has been well discussed in [5, 8]. In this paper we use tight frames to approximate a function, its derivatives, as well as various inner products. Then we apply the frame representations to approximate the solutions of the BVPs.

This paper is organized as follows.

- (i) We start by discussing WFFs with some properties, followed by an approximation of a function using WFFs in Section 2.
- (ii) We discuss representation of various operators using WFFs in Section 3.

- (iii) In Section 4 we approximate some two-point BVPs using WFFs.
- (iv) We finish with a conclusion in Section 5.

## 2. A Short Review of Frames

In this section we review in short frames, windowed Fourier frame functions, and the windowed Fourier frame transformation (WFFT) (to approximate any function  $f(x)$ ). We start by discussing frames and windowed Fourier frames. Then we discuss construction of an efficient window function briefly and use this function to construct windowed Fourier frames.

A frame is a family of vectors  $\{\phi_n\}_{n \in \Gamma}$  that characterizes any function  $f$  from its inner product  $\{\langle f, \phi_n \rangle\}_{n \in \Gamma}$ . It is possible to recover a vector  $f$  in a Hilbert space  $\mathbf{H}$  from its inner products with a family of vectors  $\{\phi_n\}_{n \in \Gamma}$ . The index set  $\Gamma$  might be finite or infinite and one can define a frame operator  $U$  so that

$$Uf[n] = \langle f, \phi_n \rangle, \quad \forall n \in \Gamma. \quad (1)$$

**Theorem 1.** *The sequence  $\{\psi_n\}_{n \in \Gamma}$  is a frame of  $\mathbf{H}$  if there exist two constants  $A > 0$ ,  $B > 0$  such that for any  $f \in \mathbf{H}$*

$$A \|f\|^2 \leq \sum_{n \in \Gamma} |\langle f, \psi_n \rangle|^2 \leq B \|f\|^2. \quad (2)$$

*If this condition is satisfied then  $U$  is called a frame operator. When  $A = B$  the frame is said to be tight [5, 9].*

It is well established that it is possible to reconstruct a signal  $f$  from its frame transformation  $Uf[n]$  using the concept of pseudo inverse which is a bounded operator expressed with a dual frame [5]. Note that a pseudo inverse is denoted by  $\tilde{U}^{-1}$  and satisfies  $\tilde{U}^{-1} = (U^*U)^{-1}U^*$  where  $U^*$  is the adjoint of  $U$ . If  $U$  is a frame operator with frame bounds  $A$  and  $B$  then  $\|\tilde{U}^{-1}\| \leq 1/\sqrt{A}$ . The pseudo inverse of a frame operator is related to a dual frame family, which is expressed by the following result.

**Theorem 2** (see [5]). *Let  $\{\psi_n\}_{n \in \mathbb{Z}}$  be a frame with bounds  $A$  and  $B$ . The dual frame defined by*

$$\tilde{\psi}_n = (U^*U)^{-1} \psi_n \quad (3)$$

*satisfies*

$$\begin{aligned} \frac{1}{B} \|f\|^2 &\leq \sum_{n \in \Gamma} |\langle f, \tilde{\psi}_n \rangle|^2 \leq \frac{1}{A} \|f\|^2, \quad \forall f \in \mathbf{H}, \\ f = \tilde{U}^{-1}Uf &= \sum_{n \in \Gamma} \langle f, \psi_n \rangle \tilde{\psi}_n = \sum_{n \in \Gamma} \langle f, \tilde{\psi}_n \rangle \psi_n. \end{aligned} \quad (4)$$

*Then the frame is tight (i.e.,  $A = B$ ). Here  $U^*$  is adjoint of  $U$  and  $\tilde{U}^{-1}$  is the pseudo inverse of  $U$ .*

We discuss windowed Fourier frame and its transformation next. For  $x, \omega \in \mathbb{R}$ , the translation can be defined by  $T_x f(t) = f(t - x)$  and modulation operator can be defined as  $M_\omega f(t) = e^{2\pi i \omega t} f(t)$ . Operators of the form  $T_x M_\omega$  or  $M_\omega T_x$

are called time-frequency shifts. Given a nonzero window function  $g \in L^2(\mathbb{R})$  and lattice parameters  $\alpha, \beta > 0$ , the set of time-frequency shifts  $\mathcal{G}(g, \alpha, \beta) = \{T_{\alpha k} M_{\beta n} g : k, n \in \mathbb{Z}\}$  is called a Gabor system. If  $\mathcal{G}(g, \alpha, \beta)$  is a frame for  $L^2(\mathbb{R})$ , it is called Gabor frame, Weyl-Heisenberg frame, or windowed Fourier frame [8]. From now we will be denoting such frames by the windowed Fourier frames. Figure 1 shows a sample window function and its Fourier transform. Gabor frame can be constructed using

$$g_{n,k}(t) = g(t - n) \exp(i2\pi kt), \quad n \in \mathbb{Z}, k \in \mathbb{Z}. \quad (5)$$

A detailed construction process of window functions can be found in [7].

**Theorem 3** (see [5]). *Let  $g$  be a window whose support is included in  $[-\pi/\xi_0, \pi/\xi_0]$ ,  $\xi_0 > 0$ . If*

$$\frac{2\pi}{\xi_0} \sum_{n=-\infty}^{\infty} |g(t - nu_0)|^2 = A > 0, \quad u_0 \in \mathbb{R}, \quad (6)$$

*for all  $t \in \mathbb{R}$ , then  $\{g_{n,k}\}_{(n,k) \in \mathbb{Z}^2}$  is a tight frame with a frame bound equal to  $A$ .*

Thus construction of a window function is important. So we aim to present some ideas to design window functions. Let us first give a brief explanation of forming a window function. Consider an interval  $[a, b]$ ,  $a, b \in \mathbb{R}$  and  $a < b < \infty$ . We want to form a function  $g$  so that  $g(x) = 0$  if  $x \notin [a, b]$  and there exists  $a < a_1 < b_1 < b$  such that we have the following.

Property 1  $g(x)$  is increasing monotonically in  $[a, a_1]$ .

Property 2  $g(x) = 1$  in  $[a_1, b_1]$ .

Property 3  $g(x)$  is decreasing monotonically in  $[b_1, b]$ .

Property 4  $g^2(x) + g^2((a+b)/2 - x) = 1$  for all  $x \in [a, b]$ .

**2.1. Computation of a Function and Sparsity.** Now we can define  $f(x)$  as

$$f(x) = \sum_{j,k} C_{j,k} g_{j,k}(x), \quad (7)$$

where  $g_{j,k}$  are the Fourier windowed frames and  $C_{j,k}$  are frame coefficients given by  $C_{j,k} = \langle f, g_{j,k} \rangle$ . Here many of the frame coefficients  $C_{j,k}$  are very close to 0 (see an example in Figure 2) and thus the coefficient matrix is sparse if we use  $g_{j,k}$  frames. To define windowed Fourier frames we consider the following steps. We fix

(i) the number of windows =  $K$ ,

(ii) the number of points on each window =  $N_l$ ,

(iii)  $N = KN_l/2$ ,

(iv) the window function  $w(x) = \sin((\pi/2)(e^{-c/x}/(e^{-c/x} + e^{-c/(1-x)})))$ , where  $x \in [0, 2]$ .

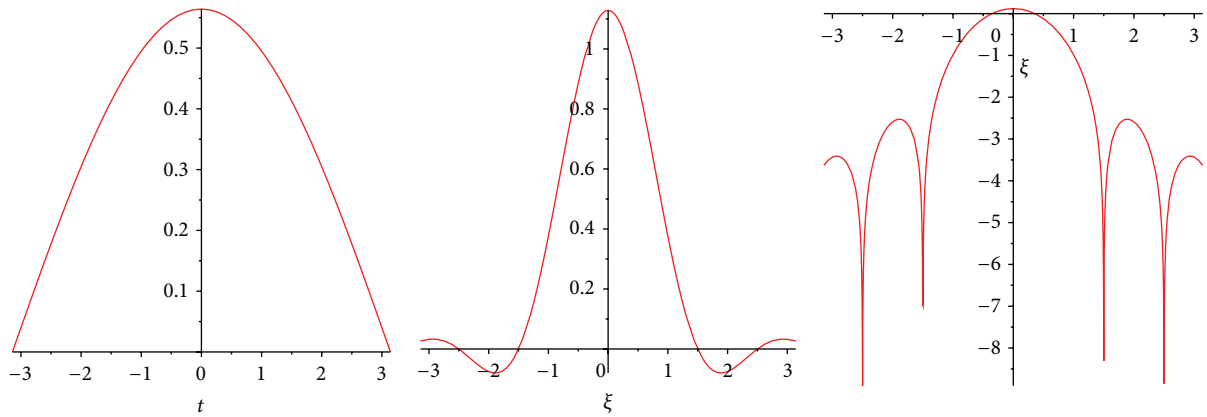


FIGURE 1: The figure shows window function  $g(t) = \sqrt{1/\pi} \cos(t/2) \mathbf{1}_{[-\pi, \pi]}(t)$ , its Fourier transform  $\hat{g}$ , and  $\log(|\hat{g}|)$ .

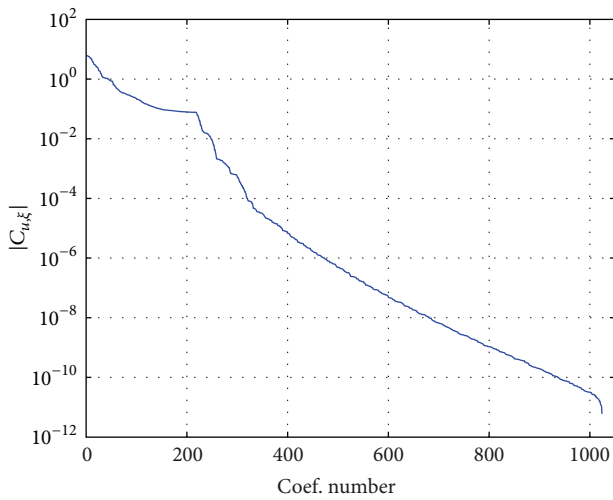


FIGURE 2: This figure shows frame coefficients  $|C_{j,k}|$  (sorted) for  $f = \exp(i\phi(x))$  where  $\phi(x) = -x^2 + x$ . Here we have considered 64 window functions with 128 grid points on each window.

We plot a cartoon of an original function  $f$  and  $f$  recovered ( $f_{rec}$ ) from its frame coefficients with sparsity set up

$$C_{j,k} = \begin{cases} C_{j,k} & \text{if } |C_{j,k}| \geq 10^{-4}, \\ 0 & \text{otherwise,} \end{cases} \quad (8)$$

in Figure 3. From this computation we notice at the approximation is accuracy as  $|f(\cdot) - f_{rec}(\cdot)| \leq C_1^* 10^{-5}$ ,  $C_1^* > 0$ . We notice that a large number of windowed Fourier frame coefficients are numerically zero or very small. This result motivates us to perform further study on the computation of operators using WFF functions.

### 3. Computation of Operators Using Windowed Fourier Frames

Here we intend to define and compute the solution  $u(x)$  of a differential equation using the frame transformation of

the initial function  $u_0(x)$  over a bounded interval  $[A, B]$ . Actually we intend to present some operators over a bounded interval  $[A, B]$  to facilitate the approximation of several BVPs with boundary conditions at  $x = A$  and  $x = B$ . Let us approximate functions  $u_0(x)$  and  $u(x)$  by  $u_0(x) = \sum_{j,k} B_{j,k} g_{j,k}(x)$  where  $B_{j,k} = \langle u_0, g_{j,k} \rangle$  are frame coefficients and

$$u(x) = \sum_{j,k} C_{j,k} g_{j,k}(x), \quad (9)$$

where  $C_{j,k} = \langle u, g_{j,k} \rangle$  are frame coefficients, respectively. We define  $u(x) = \mathcal{L}u_0$  where  $\mathcal{L}$  is an operator. It is our goal to find the matrix that maps  $B_{j,k}$  to  $C_{j,k}$  and to find a way to compute the matrices. Here

$$u = \mathcal{L}u_0(x) = \mathcal{L} \left( \sum_{j,k} B_{j,k} g_{j,k}(x) \right) = \sum_{j,k} B_{j,k} \mathcal{L}g_{j,k}(x) \quad (10)$$

and thus the frame coefficients  $C_{l,m}$  to approximate  $u$  can be defined as

$$\begin{aligned} C_{l,m} &= \langle u, g_{l,m} \rangle = \left\langle \sum_{j,k} B_{j,k} \mathcal{L}g_{j,k}(x), g_{l,m} \right\rangle \\ &= \sum_{j,k} B_{j,k} \langle \mathcal{L}g_{j,k}(x), g_{l,m} \rangle, \end{aligned} \quad (11)$$

and the first approach of  $C_{l,m}$  is simpler since we need to compute the integral over the support of the window function only. In practice, we are interested in the following operators:

- (1) multiplication by a function,
- (2) differentiation operator (derivative as an operator),

which are used to formulate PDEs and to present solutions.

3.1.  $\mathcal{L}$  as Multiplication:  $\mathcal{L}g(x) = c(x)g(x)$ . We define  $\mathcal{L}$  as multiplication operator  $\mathcal{L}g(x) = c(x)g(x)$ , where  $c(x)$  is

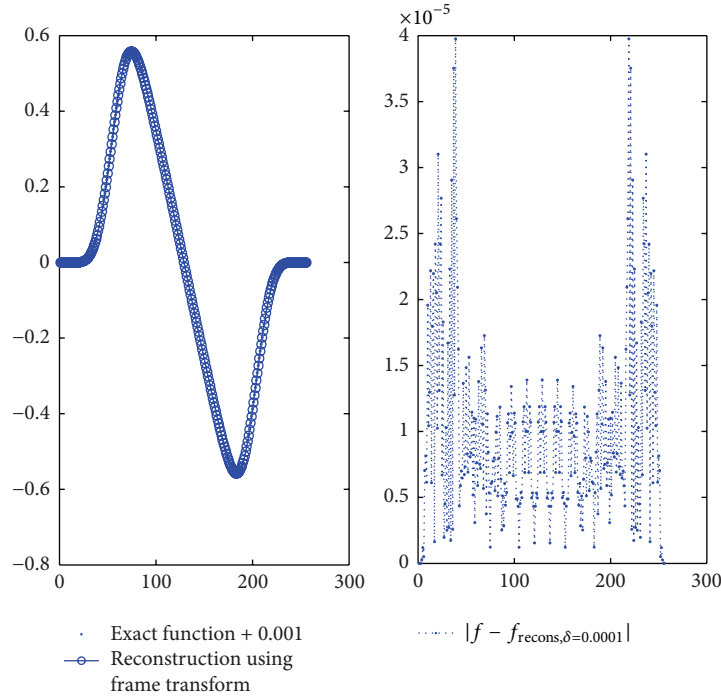


FIGURE 3: This figure shows the original function  $f$  and recovered function from the frame coefficients ( $|C_{j,k}| > \delta = 10^{-4}$ ) and the error we made with such an approximation.

an arbitrary function. Then the matrix elements  $\langle \mathcal{L}g_{l,m}(x), g_{j,k}(x) \rangle$  of (11) can be written as

$$M_{j,k;l,m} = \langle c(x)g_{l,m}(x), g_{j,k}(x) \rangle, \quad (12)$$

where  $c(x)$  is a function which depends on  $x$ . Thus we can think of the following three cases.

- (1) When  $j = l$ , both the windows have identical support, and so the integral can be written as

$$\begin{aligned}
 M_{j,k;l,m} &= e^{i\xi_0 u_0(m-k)l} \int e^{-i(k-m)\xi_0 x} c(x + ju_0) g^2(x) dx \\
 &= e^{i\xi_0 u_0(m-k)l} \text{ times Fourier transform of} \\
 & c(x + ju_0) g^2(x).
 \end{aligned} \quad (13)$$

- (2) When  $|j - l| > 1$ , then the windows do not have overlapping support, and so

$$\begin{aligned}
 M_{j,k;l,m} &= e^{i\xi_0 u_0(m-k)l} \\
 & \cdot \int e^{-i(k-m)\xi_0 x} c(x + ju_0) g \\
 & \cdot (x - (j - l)u_0) g(x) dx = 0.
 \end{aligned} \quad (14)$$

- (3) When  $j - l = \pm 1$ , then two successive windows overlap and we can compute the elements by

$$\begin{aligned}
 M_{j,k;l,m} &= e^{i\xi_0 u_0(m-k)l} \\
 & \cdot \int e^{-i(k-m)\xi_0 x} c(x + ju_0) g \\
 & \cdot (x - (j - l)u_0) g(x) dx \\
 &= e^{i\xi_0 u_0(m-k)l} \text{ times Fourier transform of} \\
 & c(x + ju_0) g(x + (j - l)x) g_0(x).
 \end{aligned} \quad (15)$$

Thus the computation of the elements becomes the frame transformation of the functions  $c(x)g_{l,m}(x), l = 0, 1, \dots, K-1, m = 0, 1, \dots, N_l - 1$ .

3.2.  $\mathcal{L}u$  as Differential Operators  $du/dx$  and  $d^2u/dx^2$ . Let us start with computing  $\mathcal{L}u = \partial u/\partial x$  using windowed Fourier frames. Consider

$$u'(x) = \sum_{j,k} C_{j,k}^*(x) g_{j,k}(x) \quad (16)$$

and we find frame coefficients

$$\begin{aligned}
 C_{j,k}^* &= \langle u'(x), g_{j,k} \rangle = \sum_{l,m} C_{l,m} \langle g'_{l,m}, g_{j,k} \rangle \\
 &= \sum_{l,m} \langle u, g_{l,m} \rangle \langle g'_{l,m}, g_{j,k} \rangle,
 \end{aligned} \quad (17)$$

where  $u(x)$  is defined by (9). Then we can define

$$\frac{du}{dx} = \sum C_{j,k}^* g_{j,k} = \sum_{j,k} \sum_{l,m} \langle u, g_{l,m} \rangle \langle g_{l,m}, g_{j,k} \rangle g_{j,k}. \quad (18)$$

Thus the computation that needs to be focused on is  $M_{j,k;l,m} = \langle g_{l,m}, g_{j,k} \rangle$ . Now

$$\begin{aligned} M_{j,k;l,m} &= \left\langle \frac{\partial}{\partial x} g_{l,m}, g_{j,k} \right\rangle \\ &= \langle i\xi_m e^{i\xi_m x} g(x - lu_0) + e^{i\xi_m x} g'(x - lu_0), g_{j,k} \rangle \\ &= i\xi_m \langle e^{i\xi_m x} g(x - lu_0), g_{j,k} \rangle \\ &\quad + \langle e^{i\xi_m x} g'(x - lu_0), g_{j,k} \rangle = I^1 + I^2, \end{aligned} \quad (19)$$

where the integral  $I^1$  is exactly the same as the integral (12) we discussed above.  $I^2$  is also of the same form and can be computed similarly to  $I^1$  once we know  $g'(x)$ . Here we have to define two matrices, one with a derivative ( $I^2$ ) and the other integral that needs to be multiplied by frequencies  $\xi_m$  ( $I^1$ ).

Now

$$\begin{aligned} I^2 &= \sum_{n=0}^{N-1} e^{2\pi i(m-k)n/N_l} g'[n - lM] g[n - jM] \\ &= (-1)^{j(k-m)} H^2(\alpha, \beta), \end{aligned} \quad (20)$$

where

$$H^2(\alpha, \beta) = \sum_{\tilde{n}=0}^{N_l-1} e^{-2\pi i\beta\tilde{n}/N_l} g[\tilde{n}] g'[\tilde{n} + \alpha M], \quad (21)$$

$$H^2(\alpha, \beta) = 0 \quad \text{if } \alpha > 1 \text{ or } \alpha < -1,$$

and for each  $j$  and  $\beta$ , computation of  $H^2(\alpha, \beta)$  can be performed as of  $H^1(\alpha, \beta)$  if the window  $g(x)$  and its derivative  $g'(x)$  have the same support.

Next we consider

$$u''(x) = \sum_{j,k} C_{j,k}^*(x) g_{j,k}(x), \quad (22)$$

where the frame coefficients are defined by

$$C_{j,k}^* = \langle u''(x), g_{j,k} \rangle = \sum_{l,m} C_{l,m} \langle g_{l,m}, g_{j,k} \rangle, \quad (23)$$

and  $u(x)$  is defined by (9). Thus  $u''(x)$  can be presented using frames following similar steps as of  $u'(x)$ . Figure 4 shows a window function and its first and second derivatives computed using central difference formula.

**3.3. Inner Product of Two Derivatives.** There are some cases when the second derivative is replaced by its variational form. Now with appropriate boundary conditions the variational form can be written as  $(\partial/\partial x)u(x), (\partial/\partial x)v(x)$  where  $v(x)$  is

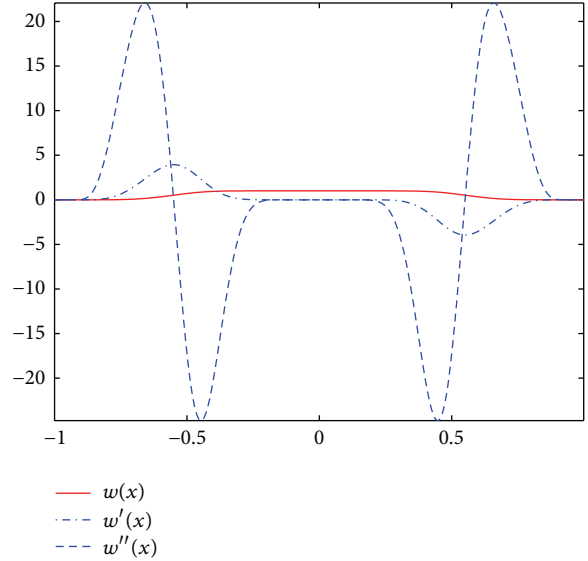


FIGURE 4: We show  $w(x) = \sin((\pi/2)(e^{-c/x}/(e^{-c/x} + e^{-c/(1-x)})))$ ,  $dw(x)/dx$ , and  $d^2w(x)/dx^2$  where  $c = 1.57$ .

a test function. We replace  $u(x)$  by  $\sum_{l,m} C_{l,m} g_{l,m}(x)$  and  $v(x)$  by  $g_{j,k}(x)$ , respectively. Thus we are interested in computing

$$\begin{aligned} &\left( \frac{\partial}{\partial x} u(x), \frac{\partial}{\partial x} g_{j,k}(x) \right) \\ &= \left( \sum_{l,m} C_{l,m} \frac{\partial}{\partial x} g_{l,m}(x), \frac{\partial}{\partial x} g_{j,k}(x) \right) \\ &= \sum_{l,m} C_{l,m} \left( \frac{\partial}{\partial x} g_{l,m}(x), \frac{\partial}{\partial x} g_{j,k}(x) \right) \\ &= \sum_{l,m} C_{l,m} M_{j,k;l,m}, \end{aligned} \quad (24)$$

for all  $j = 0, 1, \dots, K - 1$  or  $l = 0, 1, \dots, K - 1$ , where

$$M_{l,m;j,k} = \left\langle \frac{\partial}{\partial x} g_{l,m}(x), \frac{\partial}{\partial x} g_{j,k}(x) \right\rangle. \quad (25)$$

#### 4. Numerical Solution of Two-Point BVPs

In the previous section we discuss the frame representation of various functions and operators. Here we aim to use the representations to approximate two-point boundary value problems. We consider

$$-a(x) \frac{d^2u(x)}{dx^2} + b(x) \frac{du(x)}{dx} + c(x) u(x) = f(x) \quad (26)$$

for all  $x \in (A, B)$  with some boundary conditions  $u(A) = u(B) = 0$ . Let  $v$  be a test function; then the variational form of the solutions can be written as

$$\begin{aligned} & \left( a(x) \frac{du(x)}{dx}, \frac{dv(x)}{dx} \right) \\ & + \left( b(x) \frac{du(x)}{dx}, v(x) \right) + (c(x)u(x), v(x)) \quad (27) \\ & = (f(x), v(x)), \end{aligned}$$

and we consider  $u(x), v(x) \in H_0^1(\Omega)$ . Now considering

$$u(x) = \sum_{l,m} C_{l,m} g_{l,m}(x) \quad (28)$$

and replacing  $v(x)$  by  $g_{j,k}(x)$

$$\begin{aligned} & \sum_{l,m} C_{l,m} \left[ a(x) \left( \frac{\partial}{\partial x} g_{l,m}(x), \frac{\partial}{\partial x} g_{j,k}(x) \right) \right. \\ & \quad + c(x) (g_{l,m}(x), g_{j,k}(x)) \\ & \quad \left. + b(x) \left( \frac{\partial}{\partial x} g_{l,m}(x), g_{j,k}(x) \right) \right] \quad (29) \\ & = (f(x), g_{j,k}). \end{aligned}$$

Thus we need to find  $C_{l,m}$  of (29).

Now we aim to display some computational results obtained with the scheme discussed in this paper. Here we solve some one-dimensional BVPs of the form (26) to demonstrate the scheme.

*Example 4.* Let  $a = 1, b(x) = 0, c(x) = 0$ , and  $f(x) = 1$  where  $0 \leq x \leq 1$ . The variational form (29) is then

$$\sum_{l,m} C_{l,m} \left( \frac{\partial g_{l,m}(x)}{\partial x}, \frac{\partial g_{j,k}(x)}{\partial x} \right) = (f(x), g_{j,k}) \quad (30)$$

which can be written as a system of linear equations with unknowns  $\underline{C}$  as

$$M\underline{C} = \underline{f}. \quad (31)$$

We present the approximate solution in Figure 5. We compare the result with that of the exact solution.

*Example 5.* Consider the BVP

$$-\frac{d^2u}{dx^2} + u = f(x), \quad u(0) = u(1) = 0, \quad (32)$$

with  $f(x) = 20e^x$ , if  $0 < x \leq 1/4$ , and  $f(x) = (1/2)e^{-x}$ , if  $1/4 < x < 1$ . Using similar steps as of the previous discussion we present the approximate solution in Figure 6.

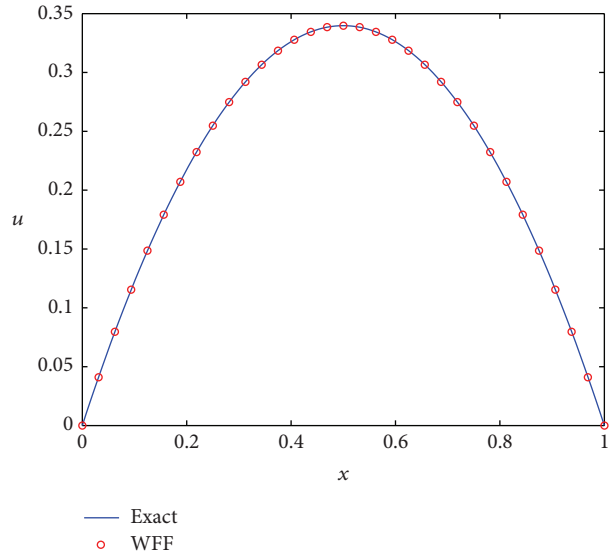


FIGURE 5: Solution of the BVP: using windowed Fourier frames.

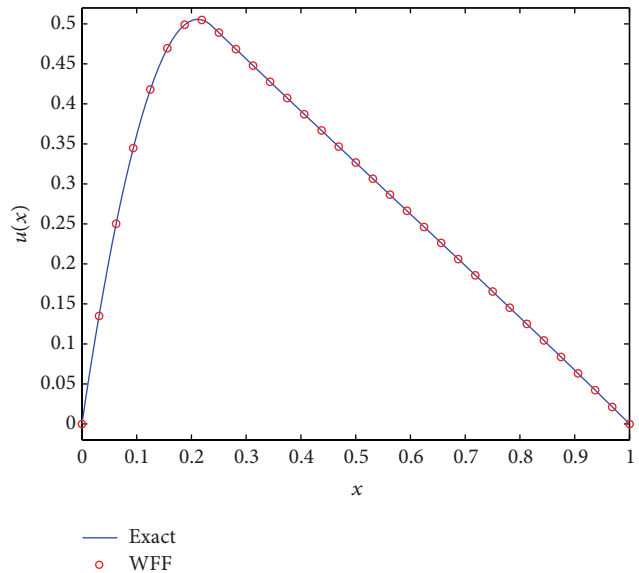


FIGURE 6: Solution of the BVP: using windowed Fourier frames.

*Example 6.* Consider the BVP

$$\begin{aligned} & -0.005 \frac{d^2u}{dx^2} + 0.1 \frac{du}{dx} + \frac{x^2 u(x)}{1+x^2} = e^{-(x-0.5)^2}, \quad (33) \\ & u(0) = u(1) = 0. \end{aligned}$$

We present the approximate solution in Figure 7. We compare the result with that of an approximation using a standard finite difference scheme.

From Figures 5, 6, and 7 we see that the approximate solutions agree with the exact solutions, which show the efficiency of the numerical scheme presented in this paper.

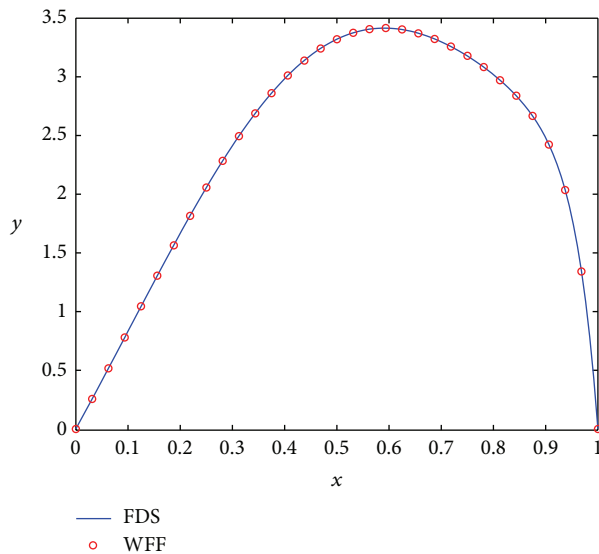


FIGURE 7: Solution of the BVP: using windowed Fourier frames and using a standard finite difference scheme.

## 5. Conclusion

In the paper windowed Fourier frames have been used to approximate two-point BVPs. From the approximation of functions we notice that the results agree with the exact solutions. We also note that a small amount of WFF coefficients is needed to reconstruct a function, and huge storage costs can be minimized. The scheme also does not require a lot of knowledge concerning the behavior of the solutions. The illustrative examples have been included to demonstrate the validity and applicability of the technique. These examples also exhibit the efficiency of the present method. There are some drawbacks: design of an efficient window function is very important. For one-dimensional problems the advantage of using the method is not very highly visible compared to the other existing numerical schemes, but we have a conjecture (our computational experiences) that this scheme can be used for higher dimensional problem where storage is a real problem for numerical computations. A multidimensional approximation using WFF would be of interest which is left as an open problem.

In this study we apply WFFs to some second-order linear BVPs. WFFs are not limited to these problems only. There are many linear higher order BVPs and nonlinear BVPs modeling scientific and engineering problems where WFFs can be applied to approximate the solutions which are left as open research problems.

## Disclosure

Samir Kumar Bhowmik is on leave from the Department of Mathematics, University of Dhaka, Dhaka, Bangladesh.

## Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

## Acknowledgment

Samir Kumar Bhowmik would like to thank Chris C. Stolk of University of Amsterdam for his kind and cordial help.

## References

- [1] A. Ashyralyev and O. Yildirim, "On stability of a third order of accuracy difference scheme for hyperbolic nonlocal BVP with self-adjoint operator," *Abstract and Applied Analysis*, vol. 2013, Article ID 959216, 15 pages, 2013.
- [2] S. K. Bhowmik, "Tchebychev polynomial approximations for  $m$ th order boundary value problems," *International Journal of Pure and Applied Mathematics*, vol. 98, no. 1, pp. 45–63, 2015.
- [3] S. K. Bhowmik, F. M. Al Faqih, and N. Islam, "A note on some numerical approaches to solve a  $\theta$  neuron networks model," *Abstract and Applied Analysis*, vol. 2014, Article ID 863842, 7 pages, 2014.
- [4] S. Kumar, S. Dhawan, and S. Kapoor, "Numerical method for advection diffusion equation using fem and b-splines," *Journal of Computational Science*, vol. 3, no. 5, pp. 429–437, 2012.
- [5] S. Mallat, *A Wavelet Tour of Signal Processing*, Academic Press, Amsterdam, The Netherlands, 3rd edition, 2009.
- [6] C. C. Stolk, "A fast method for linear waves based on geometrical optics," *SIAM Journal on Numerical Analysis*, vol. 47, no. 2, pp. 1168–1194, 2009.
- [7] S. K. Bhowmik and C. C. Stolk, "Preconditioners based on windowed Fourier frames applied to elliptic partial differential equations," *Journal of Pseudo-Differential Operators and Applications*, vol. 2, no. 3, pp. 317–342, 2011.
- [8] K. Gröchenig, *Foundations of Time-Frequency Analysis*, Birkhäuser, Boston, Mass, USA, 2000.
- [9] O. Christensen, *An Introduction to Frames and Riesz Bases*, Birkhauser, 2004.



# Hindawi

Submit your manuscripts at  
<http://www.hindawi.com>

