Research Paper
Computational Methods for Differential Equations
http://cmde.tabrizu.ac.ir
Vol. 10, No. 4, 2022, pp. 942-953
DOI:10.22034/cmde.2021.40288.1758

Combining the reproducing kernel method with a practical technique to solve the system of nonlinear singularly perturbed boundary value problems

Saeid Abbasbandy[1,*], **Hussein Sahihi**[2], **and Tofigh Allahviranloo**[3]

[1]Department of Applied Mathematics, Faculty of Science, Imam Khomeini International University, Qazvin 34149-16818, Iran.

[2]Department of Mathematics, Science and Research Branch, Islamic Azad University, Tehran, Iran.

[3]Bahcesehir University, Faculty of Engineering and Natural Sciences, Istanbul, Turkey.

Abstract

In this paper, a reliable new scheme is presented based on combining Reproducing Kernel Method (RKM) with a practical technique for the nonlinear problem to solve the System of Singularly Perturbed Boundary Value Problems (SSPBVP). The Gram-Schmidt orthogonalization process is removed in the present RKM. However, we provide error estimation for the approximate solution and its derivative. Based on the present algorithm in this paper, can also solve linear problem. Several numerical examples demonstrate that the present algorithm does have higher precision.

Keywords. Reproducing kernel method, Singularly perturbed BVPs, Convergence analysis, Error analysis, System of differential equations.

2010 Mathematics Subject Classification. 65L04, 65L11.

1. Introduction

The systems of singularly perturbed BVPs are used in various fields of engineering sciences, They have applications in modeling electrochemical reactions, turbulence in water wave when they interact with current, electroanalytical chemistry when investigating diffusion processes complicated by chemical reactions, equations of predator−prey population dynamics, control theory[19, 24, 27, 28, 30]. Many authors have presented various mathematical methods to solve SSPBVP [15, 26, 29, 34]. In [2–7, 13, 16–18, 21, 23, 25, 38], RKM is provided to solve some system of the differential and Volterra integral equations. Some applications of reproducing kernel method without using the orthogonalization process are given in [1, 8, 9, 31, 32]. Consider the following system of singularly perturbed differential equations,

$$\begin{cases} \mathcal{L}_1(u_1(t)) = \mathcal{N}_1(t, U(t)) + \mathcal{F}_1(t), \\ \mathcal{L}_2(u_2(t)) = \mathcal{N}_2(t, U(t)) + \mathcal{F}_2(t), \\ \qquad\vdots \\ \mathcal{L}_k(u_k(t)) = \mathcal{N}_k(t, U(t)) + \mathcal{F}_k(t), \\ \\ \qquad u_d(a) = u_d(b) = \gamma_d, \end{cases} \tag{1.1}$$

where $t \in \Omega = [a, b]$ and γ_d is constant,

$$\mathcal{L}_d(u_d(t)) \equiv \varepsilon_d u_d''(t) + \frac{1}{\mathcal{P}_d(t)} u_d'(t) + \frac{1}{\mathcal{Q}_d(t)} u_d(t),$$

Received: 11 June 2020 ; Accepted: 20 June 2021.
∗ Corresponding author. Email: abbasbandy@ikiu.ac.ir; abbasbandy@yahoo.com.

and $\mathcal{N}_d(t, U(t))$ for $d = 1, \dots, k$ are linear and nonlinear differential operators respectively,

$$\mathcal{F}_d(t),\ \mathcal{P}_d(t),\ \mathcal{Q}_d(t),$$

are sufficiently smooth functions and $U(t) = (u_1(t), u_2(t), \dots, u_k(t))^T$ is unknown vector functions which must be determined. ε_d is perturbation parameter and $0 < \varepsilon_d \ll 1$. The present method solve the equations in system of differential equations (1.1), simultaneously.In this paper, regardless of whether the system of differential equations is linear or nonlinear, we solve the system of differential equations with general technique for nonlinear problem. In fact, for solving any equation in the system of differential equations, we suppose only one of the functions and its derivatives as unknown that we call the linear operator $(\mathcal{L}_d(u_d(t)))$, and the remaining unknown functions are assumed as part of nonlinear term that we call the nonlinear operator $(\mathcal{N}_d(t, U(t)))$, where $d = 1, 2, \dots, k$ and eventually the problem is converted to nonlinear form and using general technique for the nonlinear problems we solve the system of differential equations. In fact, nonlinear problem turns into a number of iterations to solve its corresponding linear problem. However, based on fundamental concept of general technique for nonlinear problems, the number of iterations for the nonlinear terms are very important to solve the system of differential equations (1.1) by using present method. Obviously, if the number of iteration is higher, the approximate solution is more accurate.

2. Preliminaries

Definition 2.1. Consider the reproducing kernel space $\mathcal{W}_2^m[a, b]$ such that $y^{m-1}(t)$ is absolutely continuous and $y^{(m)}(t) \in \mathcal{L}^2[a, b]$ and $y(a) = y(b) = 0$ where $y(t) \in \mathcal{W}_2^m[a, b]$. The inner product and norm are given as follows:

$$< y_1(t), y_2(t) >_{\mathcal{W}_2^m[a,b]} = \sum_{i=0}^{m-1} y_1^{(i)}(a) y_2^{(i)}(a) + \int_a^b y_1^{(m)}(t) y_2^{(m)}(t) dt,$$
$$\|y(t)\|_{\mathcal{W}_2^m[a,b]} = \sqrt{< y, y >_{\mathcal{W}_2^m[a,b]}}.$$

Theorem 2.2. *Reproducing kernels $\mathcal{W}_2^3[a, b]$ and $\mathcal{W}_2^1[a, b]$ for $a = 0,\ b = 1$ are given respectively as [14],*

$$\mathcal{R}(t,x) = \frac{\begin{cases} 50tx^4 + 13t^5 - 15t^4x + 105t^3x^2 - 25t^2x^3 & t \le x \\ 13x^5 + 50t^4x - 25t^3x^2 + 105t^2x^3 - 15tx^4 & x < t \end{cases}}{1560} - \frac{t^5x^5}{18720} + \frac{t^5x^4}{3744} - \frac{t^5x^3}{1872} - \frac{t^5x^2}{624} - \frac{t^5x}{156} + \frac{t^4x^5}{3744} - \frac{5t^4x^4}{3744} + \frac{5t^4x^3}{1872} + \frac{5t^4x^2}{624} - \frac{t^3x^5}{1872} + \frac{5t^3x^4}{1872} - \frac{5t^3x^3}{936} - \frac{5t^3x}{78} - \frac{t^2x^5}{624} + \frac{5t^2x^4}{624} + \frac{21t^2x^2}{104} - \frac{5t^2x}{26} - \frac{tx^5}{156} - \frac{5tx^3}{78} - \frac{5tx^2}{26} + \frac{3tx}{13},$$

$$\mathcal{K}(t,x) = \begin{cases} 1+t, & t \le x, \\ 1+x, & x < t. \end{cases}$$

Consider one of the equation in the SSPBVP (1.1) in iterative scheme as

$$\mathcal{L}_d(u_{d,n}(t)) = \mathcal{N}_d(t, U_{n-1}(t)) + \mathcal{F}_d(t),$$

where $\mathcal{L}_d : \mathcal{W}_2^3[a, b] \longrightarrow \mathcal{W}_2^1[a, b]$ is a bounded linear operator, and $\mathcal{N}_d$ is a continuous nonlinear operator, and

$$U_{n-1}(t) = \left[u_{1,n-1}(t), u_{2,n-1}(t), \dots, u_{k,n-1}(t)\right]^T.$$

For $n = 1$, $U_0(t) = \left[u_{1,0}(t), u_{2,0}(t), \dots, u_{k,0}(t)\right]^T$ that satisfies boundary conditions of the problem (1.1), therefore $U_0(t) = \left[\gamma_1, \gamma_2, \dots, \gamma_k\right]^T$, and in each iteration $n = 2, 3, \dots$, we obtain $U_{n-1}(t)$. We choose a dense set $\{t_i\}_{i=1}^{\infty}$ on Ω and define, $\phi_{d,i}(t) = \mathcal{K}_x(t)|_{t=t_i}$, and $\psi_{d,i}(t) = \mathcal{L}_d^* \phi_{d,i}(t)$, where $\mathcal{L}_d^*$ is adjoint operator of $\mathcal{L}_d$ and suppose $\mathcal{L}_d^{-1}$ exists. Consider one of the following complete function systems that are obtained from reproducing kernel $\mathcal{R}_x(t)$ of the space $\mathcal{W}_2^3[a, b]$,

$$\psi_{d,i}(t) = \mathcal{L}_{d,x} \mathcal{R}_x(t)|_{x=t_i}, \quad \text{or} \quad \varphi_{d,i}(t) = \mathcal{R}_x(t)|_{x=t_i}.$$

Theorem 2.3. *The Exact solution of equation*

$$\mathcal{L}_d(u_{d,n}(t)) = \mathcal{N}_d(t, U_{n-1}(t)) + \mathcal{F}_d(t)$$

can be represented as,

$$u_{d,n}(t) = \sum_{i=1}^{\infty} c_{d,i,n}\psi_{d,i}(t), \qquad n = 1, 2, \dots, \tag{2.1}$$

where the unknown coefficients $c_{d,i,n}$ must be determined.

Proof. See [33, 35, 36]. □

3. Construction of the numerical method

Suppose the approximate solution with N collocation points throughout the interval Ω is,

$$u_{d,n,N}(t) = \sum_{i=1}^{N} c_{d,i,n}\psi_{d,i}(t), \tag{3.1}$$

and n must be sufficiently large. Now determine the unknown coefficients $c_{d,i,n}$ where $i = 1, 2, \dots, N$ and $n = 1, 2, \dots$. First consider following equation,

$$\mathbb{R}_{d,N}(t) = \mathcal{L}_d(u_{d,n,N}(t)) - \mathcal{N}_d(t, U_{n-1,N}(t)) - \mathcal{F}_d(t), \tag{3.2}$$

determine the unknown coefficients $c_{d,i,n}$ such that $< \mathbb{R}_{d,N}(t), \psi_{d,j}(t) >_{\mathcal{W}_2^3} = 0$ for $j = 1, 2, \dots, N$, therefore we have,

$$< \mathbb{R}_{d,N}(t), \psi_{d,j}(t) > = < \mathcal{L}_1(u_{d,n,N}(t)) - \mathcal{N}_d(t, U_{n-1,N}(t)) - \mathcal{F}_d(t), \psi_{d,j}(t) > =$$

$$< \mathcal{L}_1(u_{d,n,N}(t)), \psi_{d,j}(t) > - < \mathcal{N}_d(t, U_{n-1,N}(t)) + \mathcal{F}_d(t), \psi_{d,j}(t) > =$$
$$< \mathcal{L}_d(\sum_{i=1}^{N} c_{d,i,n}\psi_{d,i}(t)), \psi_{d,j}(t) > - < \mathcal{N}_d(t, U_{n-1,N}(t)) + \mathcal{F}_d(t), \psi_{d,j}(t) > =$$
$$\sum_{i=1}^{N} c_{d,i,n} < \mathcal{L}_1(\psi_{d,i}(t)), \psi_{d,j}(t) > - < \mathcal{N}_d(t, U_{n-1,N}(t)) + \mathcal{F}_d(t), \psi_{d,j}(t) > = 0,$$

and therefore we have following system of linear algebraic equations and solve it to obtain the unknown coefficients $c_{d,i,n}$ for $n = 1, 2, \dots$ and $j = 1, 2, \dots, N$ as follows

$$\sum_{i=1}^{N} c_{d,i,n}\mathcal{L}_d\psi_{d,i}(t)|_{t=t_j} = \mathcal{N}_d(t, U_{n-1,N}(t))|_{t=t_j} + \mathcal{F}_d(t_j), \tag{3.3}$$

and $\psi_{d,i}(t) = \mathcal{L}_{d,x}\mathcal{R}_x(t)|_{x=t_i}$ and $i = 1, 2, \dots, N$ and $\mathcal{L}_{d,x}$ is the linear operator in equation (1.1). We obtain a approximate solution for the SSPBVP (1.1), define the approximate solution in the form,

$$U_{n,N}(t) = \left[u_{1,n,N}(t), u_{2,n,N}(t), \dots, u_{k,n,N}(t)\right]^T, \ n = 1, 2, \dots,$$

where n is the number of iterations for nonlinear term $\mathcal{N}_d(t, U_{n-1,N}(t))$ and,

$$\begin{cases} u_{1,n,N}(t) = \sum_{i=1}^{N} c_{1,i,n}\psi_{1,i}(t), \\ u_{2,n,N}(t) = \sum_{i=1}^{N} c_{2,i,n}\psi_{2,i}(t), \\ \qquad \vdots \\ u_{k,n,N}(t) = \sum_{i=1}^{N} c_{k,i,n}\psi_{k,i}(t), \end{cases} \tag{3.4}$$

and we determine the coefficients $c_{d,i,n}$ where $d = 1, 2, \dots, k$ by solving the following linear system of algebraic equations,

$$\begin{cases} \sum_{i=1}^{N} c_{1,i,n}\mathcal{L}_1\psi_{1,i}(t)|_{t=t_j} = \mathcal{N}_1(t, U_{n-1,N}(t))|_{t=t_j} + \mathcal{F}_1(t_j), \\ \sum_{i=1}^{N} c_{2,i,n}\mathcal{L}_2\psi_{2,i}(t)|_{t=t_j} = \mathcal{N}_2(t, U_{n-1,N}(t))|_{t=t_j} + \mathcal{F}_2(t_j), \\ \qquad \vdots \\ \sum_{i=1}^{N} c_{k,i,n}\mathcal{L}_k\psi_{k,i}(t)|_{t=t_j} = \mathcal{N}_k(t, U_{n-1,N}(t))|_{t=t_j} + \mathcal{F}_k(t_j), \end{cases} \tag{3.5}$$

where $n = 1, 2, \ldots,\ j = 1, 2, \ldots, N$ and for $n = 1$ we choose the initial vector function $U_{0,N}(t) = \left[\gamma_1, \gamma_2, \ldots, \gamma_k\right]^T$ and we obtain $U_{n-1,N}(t)$ for each iteration, $n = 2, 3, \ldots$.

Remark 3.1. Based on the mentioned algorithm, obviously all of the equations in the system of differential equations (1.1) are solved simultaneously. This Property makes the present method easy to apply linear and nonlinear system of differential equations.

4. Convergence and error analysis

Lemma 4.1. *$\mathcal{E} = \{u_{d,n,N}(t) \mid \ \|u_{d,n,N}(t)\|_{\mathcal{W}_2^3[a,b]} \le \varrho_d,\ d = 1, 2, \ldots, k\}$ is compact in $C[a, b]$ and ϱ_d is a constant.*

Proof. See [10, 22, 33, 37]. □

Theorem 4.2. *The approximate solution $U_{n,N}(t)$ converge to the exact solution $U(t)$.*

Proof. From [33], $\|u_{d,n_l,N}(t) - u_d(t)\|_{\mathcal{W}_2^3} \longrightarrow 0$ when $N \longrightarrow \infty$. Now we prove that $u_{d,n_l,N}(t) \longrightarrow u_d(t)$ uniformly convergent when, $N \to \infty, l \to \infty$, from the reproducing properties we have,

$$\begin{aligned} |u_{d,n_l,N}(t) - u_d(t)| =& |< u_{d,n_l,N}(x) - u_d(x), \mathcal{R}_t(x) >_{\mathcal{W}_2^3}| \\ \le& \|u_{d,n_l,N}(x) - u_d(x)\|_{\mathcal{W}_2^3} \|\mathcal{R}_t(x)\|_{\mathcal{W}_2^3} \\ \le& m_d \|u_{d,n_l,N}(x) - u_d(x)\|_{\mathcal{W}_2^3}, \end{aligned}$$

therefore $u_{d,n_l,N}(t) \longrightarrow u_d(t)$ uniformly convergent when, $N \to \infty$. Similar to above equation we have $u^{(j)}_{d,n_l,N}(t) \longrightarrow u^{(j)}_d(t)$ uniformly convergent when, $N \to \infty$ for $j = 1, 2$. □

Theorem 4.3. *Let $U^{(j)}_{n,N}(t)$ be approximate solution in $\mathcal{W}_2^3[a, b]$ and $U^{(j)}(t)$ where $j = 0, 1,$ be exact solution for the system of differential equations (1.1), then*

$$\|u_{d,n,N}(t) - u_d(t)\|_\infty = max_{t\in[a,b]}|u_{d,n,N}(t) - u_d(t)| \le \hat{c}_d\, h_d^2,$$

$$\|u'_{d,n,N}(t) - u'_d(t)\|_\infty = max_{t\in[a,b]}|u'_{d,n,N}(t) - u'_d(t)| \le \check{c}_d\, h_d,$$

where $h_d = max|t_{i+1} - t_i|$ and $d = 1, 2, \ldots, k,\ i = 1, 2, \ldots, N$ and $\hat{c}_d, \check{c}_d$ are positive constants.

Proof. See [11, 12, 20, 33]. □

5. Numerical results

In this section several numerical examples are presented to show the validity of the theoretical results. Let $\mathcal{S}_1 = \{(\varepsilon_1, \varepsilon_2) \mid \varepsilon_1 = 1, 10^{-2}, 10^{-4}, \ldots, 10^{-30};\ \varepsilon_2 = 1, 10^{-2}, 10^{-4}, \ldots, 10^{-30}\}$, $\mathcal{S}_2 = \{(\varepsilon_1, \varepsilon_2, \varepsilon_3) \mid \varepsilon_1 = 1, 10^{-2}, 10^{-4}, \ldots, 10^{-30}; \varepsilon_2 = 1, 10^{-2}, 10^{-4}, \ldots, 10^{-30};\ \varepsilon_3 = 1, 10^{-2}, 10^{-4}, \ldots, 10^{-30}\}$. Absolute error for the approximate solution and its derivative are given with, $E^{u_N(t)} = |u_N(t) - u(t)|,\ E^{v_N(t)} = |v_N(t) - v(t)|,\ E^{z_N(t)} = |z_N(t) - z(t)|$ and $E^{u'_N(t)} = |u'_N(t) - u'(t)|,\ E^{v'_N(t)} = |v'_N(t) - v'(t)|,\ E^{z'_N(t)} = |z'_N(t) - z'(t)|$ respectively, and collocation points are $t_i = \frac{i}{N+1},\ \ i = 1, 2, \ldots, N$ and for the Figures 1,2,. . . ,10 suppose $\varepsilon_1, \varepsilon_2, \varepsilon_3 = 10^{-20}$.

Algorithm

1. Choose N points in the interval Ω;
2. Put $\psi_{d,i}(t) = \mathcal{L}_{d,x}\mathcal{R}_x(t)|_{x=t_i}$ and $d = 1, 2, \ldots, k$; $i = 1, 2, \ldots, N$;
3. Put $\mathbb{G}_d = \Big[\mathcal{L}_d\psi_{d,j}(t)|_{t=t_i}\Big]_{i,j=1,2,\ldots,N}$ and $d = 1, 2, \ldots, k$;
4. Choose a proper value of n, such that n is the number of iterations for the nonlinear terms;
5. Put $\ell = 0$;
6. Choose the initial functions $U_{\ell,N}(t) = \Big[u_{1,\ell}(t), u_{2,\ell}(t), \ldots, u_{k,\ell}(t)\Big]^T$ (for the problem (1.1), suppose $U_{0,N}(t) = \Big[0, 0, \ldots, 0\Big]^T$);
7. Set $\ell = \ell + 1$;
8. For $d = 1, 2, \ldots, k$ compute $\mathbb{F}_d = \Big[\mathcal{N}_d(t, U_{\ell-1,N}(t))|_{t=t_i} + \mathcal{F}_d(t_i)\Big]^T_{i=1,2,\ldots,N}$;
9. For $d = 1, 2, \ldots, k$ solve system $\mathbb{G}_d\mathbb{A}_d = \mathbb{F}_d$, where

$$\mathbb{A}_d = \Big[c_{d,1,\ell}(t), c_{d,2,\ell}(t), \ldots, c_{d,N,\ell}(t)\Big]^T;$$

10. Put $u_{d,\ell,N}(t) = \sum_{j=1}^{N} c_{d,j,\ell}\psi_{d,j}(t)$ where $d = 1, 2, \ldots, k$;
11. Put $U_{\ell,N}(t) = \Big[u_{1,\ell,N}(t), u_{2,\ell,N}(t), \ldots, u_{k,\ell,N}(t)\Big]^T$;
12. If $\ell < n$ then go to step 7 else stop the algorithm.

Example 5.1.

$$\begin{cases} \varepsilon_1 u''(t) + \frac{1}{t}u'(t) + t^2 v(t) = f_1(t), & 0 \le t \le 1, \\ \varepsilon_2 v''(t) + \frac{1}{t-1}v'(t) + tu'(t) = f_2(t), & \\ u(0) = u(1) = 0, \qquad v(0) = v(1) = 0. & \end{cases}$$

The exact solutions for Example 5.1 are $u(t) = t(t-1)e^{-t}$, $v(t) = t^2 - t^3$.

Example 5.2.

$$\begin{cases} \varepsilon_1 u''(t) + \frac{1}{t(t-1)}u'(t) + e^{-t}v'(t) = f_1(t), & 0 \le t \le 1, \\ \varepsilon_2 v''(t) + \frac{1}{t^2}v'(t) + sin(\sqrt{t})u'(t) = f_2(t), & \\ u(0) = u(1) = 0, \qquad v(0) = v(1) = 0. & \end{cases}$$

The exact solutions for Example 5.2 are

$$u(t) = sin(\pi t)e^{-t},\ \ v(t) = cos(\pi t(t-1)) - 1.$$

Example 5.3.

$$\begin{cases} \varepsilon_1 u''(t) + \frac{1}{sin(\pi t)}u'(t) + u'(t)^2\sqrt{v(t)} = f_1(t), & 0 \le t \le 1, \\ \varepsilon_2 v''(t) + \frac{1}{(t^2-1)}v'(t) + tv'(t)u'(t)^2 = f_2(t), & \\ u(0) = u(1) = 0, \qquad v(0) = v(1) = 0. & \end{cases}$$

The exact solutions for Example 5.3 are

$$u(t) = sin(\frac{\pi}{2}t)(t-1),\ \ v(t) = (t^2 - t^3)e^{-sin(t(t-1))}.$$

Example 5.4.

$$\begin{cases} \varepsilon_1 u''(t) + \frac{1}{t\sqrt{(t-1)}}u'(t) + v'(t)^3u'(t) + e^{sin(v'(t)u'(t)^3)} = f_1(t), \\ \varepsilon_2 v''(t) + \frac{1}{2t^2(t-1)^3}v'(t) - sin(v(t)u'(t)^2) + e^{v'(t)^3}\sqrt{u'(t)} = f_2(t), \\ u(0) = u(1) = 0, \qquad v(0) = v(1) = 0, \qquad 0 \le t \le 1. \end{cases}$$

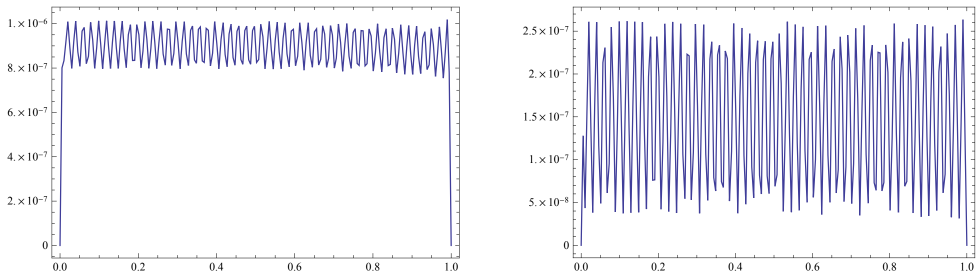

FIGURE 1. Absolute error for Example 5.1 with $N = 100$ and $n = 10$ (Left: $E^{u_N(t)}$; Right: $E^{v_N(t)}$).

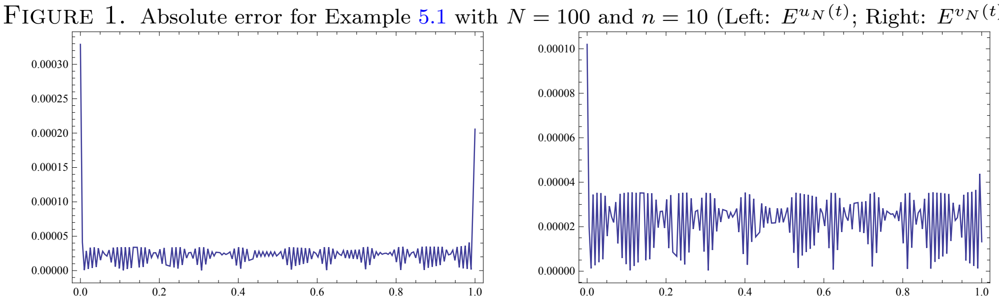

FIGURE 2. Absolute error for Example 5.1 with $N = 100$ and $n = 10$ (Left: $E^{u'_N(t)}$; Right: $E^{v'_N(t)}$).

TABLE 1. Max absolute errors with $N = 100$

$(\varepsilon_1, \varepsilon_2) \in \mathcal{S}_1$	Present Method			
	Example 5.1 $n = 10$	Example 5.1 $n = 15$	Example 5.3 $n = 20$	Example 5.4 $n = 25$
$E^{u_N(t)}$	1.50×10^{-6}	2.30×10^{-6}	2.00×10^{-6}	6.40×10^{-6}
$E^{v_N(t)}$	8.50×10^{-7}	1.00×10^{-5}	9.00×10^{-7}	2.00×10^{-5}
$E^{u'_N(t)}$	3.50×10^{-4}	6.60×10^{-4}	3.30×10^{-4}	1.80×10^{-3}
$E^{v'_N(t)}$	2.70×10^{-4}	3.00×10^{-3}	2.90×10^{-4}	3.20×10^{-3}

The exact solutions for Example 5.4 are

$$u(t) = sin(\pi t^2(t-1)^2),\ v(t) = cos(sin(\pi t(t-1)) + \pi) + 1.$$

Example 5.5.

$$\begin{cases} \varepsilon_1 u''(t) + \frac{1}{sin(\pi t)} u'(t) + z'(t)^2 v'(t)^3 u'(t) + e^{sin(z'(t)v'(t)u'(t)^3)} = f_1(t), \\ \varepsilon_2 v''(t) + \frac{1}{t^3(t-1)^3} v'(t) - z(t) sin(v(t)u'(t)^2) + e^{v'(t)^3 u'(t)} + \\ \sqrt{z(t)^3 v(t)^2 u'(t)} = f_2(t), \\ \varepsilon_3 z''(t) + \frac{1}{(t-1)\sqrt{t^2}} z'(t) - sin(e^{z(t)u(t)^2}) + v(t)^2 u'(t) e^{v'(t)\sqrt{z'(t)}} = f_3(t), \\ u(0) = u(1) = 0,\ \ v(0) = v(1) = 0,\ \ z(0) = z(1) = 0,\ \ 0 \le t \le 1. \end{cases}$$

The exact solutions for Example 5.5 are

$$u(t) = cos(sin(\pi t(t-1)) + \pi) + 1,\ v(t) = sin(\pi t^2(t-1)^2),\ z(t) = t^2 sin(e^{t(t-1)} - 1).$$

TABLE 2. Max absolute error and convergence order ($Log_2^{E^N/E^{2N}}$)

$(\varepsilon_1, \varepsilon_2 = 10^{-30})$	Present Method $N = 16$	$N = 32$	$N = 64$	$N = 128$
Example 5.1				
$n = 10$				
$E^{u_N(t)}$	2.00×10^{-4}	2.90×10^{-5}	3.80×10^{-6}	4.80×10^{-7}
	2.78588	2.93198	2.98489	
$E^{v_N(t)}$	5.20×10^{-5}	7.50×10^{-6}	1.00×10^{-6}	1.30×10^{-7}
	2.79355	2.90689	2.94342	
$E^{u'_N(t)}$	1.10×10^{-2}	3.00×10^{-3}	8.00×10^{-4}	2.10×10^{-4}
	1.87447	1.90689	1.92961	
$E^{v'_N(t)}$	3.50×10^{-3}	9.50×10^{-4}	2.50×10^{-4}	6.40×10^{-5}
	1.88136	1.926	1.96578	
Example 5.2				
$n = 15$				
$E^{u_N(t)}$	3.40×10^{-4}	5.00×10^{-5}	6.80×10^{-6}	8.60×10^{-7}
	2.76553	2.87832	2.98313	
$E^{v_N(t)}$	2.00×10^{-3}	2.80×10^{-4}	3.70×10^{-5}	4.70×10^{-6}
	2.8365	2.91983	2.97679	
$E^{u'_N(t)}$	1.70×10^{-2}	4.80×10^{-3}	1.30×10^{-3}	3.30×10^{-4}
	1.82443	1.88452	1.97797	
$E^{v'_N(t)}$	1.00×10^{-1}	2.70×10^{-2}	7.00×10^{-3}	1.80×10^{-3}
	1.88897	1.94753	1.95936	
Example 5.3				
$n = 20$				
$E^{u_N(t)}$	1.90×10^{-4}	2.70×10^{-5}	3.60×10^{-6}	4.60×10^{-7}
	2.81497	2.90689	2.96829	
$E^{v_N(t)}$	1.90×10^{-4}	2.60×10^{-5}	3.40×10^{-6}	4.30×10^{-7}
	2.86942	2.9349	2.98313	
$E^{u'_N(t)}$	1.00×10^{-2}	2.70×10^{-3}	7.00×10^{-4}	1.80×10^{-4}
	1.88897	1.94753	1.95936	
$E^{v'_N(t)}$	1.00×10^{-2}	2.70×10^{-3}	6.80×10^{-4}	1.75×10^{-4}
	1.88897	1.98935	1.95818	
Example 5.4				
$n = 25$				
$E^{u_N(t)}$	1.20×10^{-3}	1.75×10^{-4}	2.30×10^{-5}	3.00×10^{-6}
	2.77761	2.92765	2.9386	
$E^{v_N(t)}$	2.30×10^{-3}	3.00×10^{-4}	3.80×10^{-5}	4.80×10^{-6}
	2.9386	2.98089	2.98489	
$E^{u'_N(t)}$	6.00×10^{-2}	1.70×10^{-2}	4.40×10^{-3}	1.15×10^{-3}
	1.81943	1.94996	1.93587	
$E^{v'_N(t)}$	1.10×10^{-1}	2.90×10^{-2}	7.30×10^{-3}	1.90×10^{-3}
	1.92338	1.99008	1.9419	

TABLE 3. Max absolute error for Example 5.5 with $N = 100$ and $n = 30$ and $(\varepsilon_1, \varepsilon_2, \varepsilon_3) \in \mathcal{S}_2$

Present Method					
$E^{u_N(t)}$	$E^{v_N(t)}$	$E^{z_N(t)}$	$E^{u'_N(t)}$	$E^{v'_N(t)}$	$E^{z'_N(t)}$
1.00×10^{-5}	6.20×10^{-6}	3.20×10^{-6}	3.00×10^{-3}	2.00×10^{-3}	1.00×10^{-3}

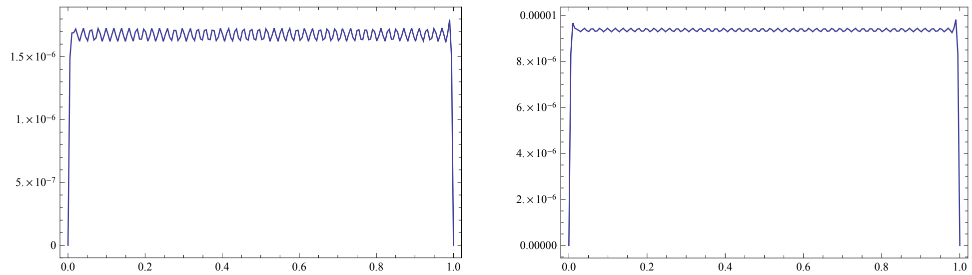

FIGURE 3. Absolute error for Example 5.2 with $N = 100$ and $n = 15$ (Left: $E^{u_N}(t)$; Right: $E^{v_N}(t)$).

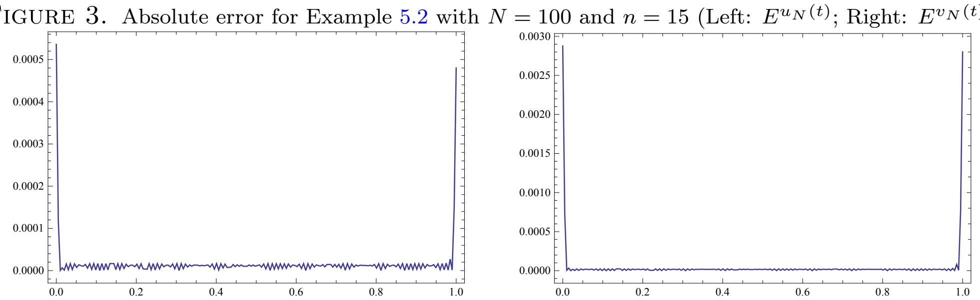

FIGURE 4. Absolute error for Example 5.2 with $N = 100$ and $n = 15$ (Left: $E^{u'_N}(t)$; Right: $E^{v'_N}(t)$).

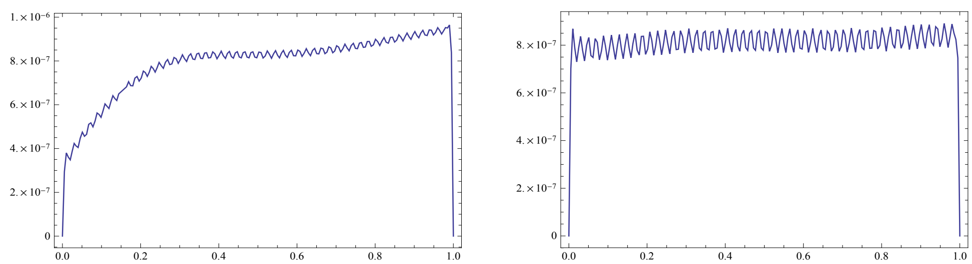

FIGURE 5. Absolute error for Example 5.3 with $N = 100$ and $n = 20$ (Left: $E^{u_N}(t)$; Right: $E^{v_N}(t)$).

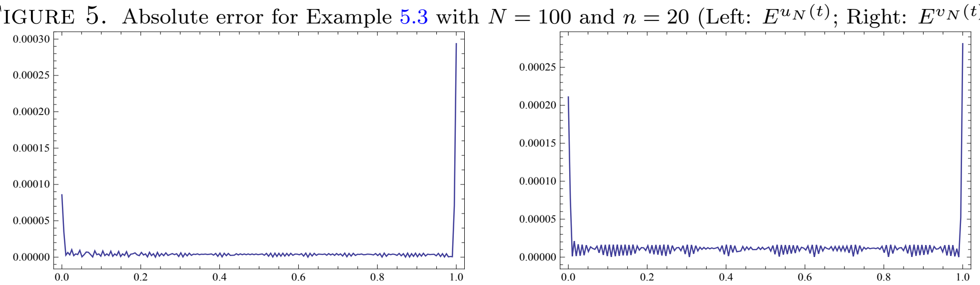

FIGURE 6. Absolute error for Example 5.3 with $N = 100$ and $n = 20$ (Left: $E^{u'_N}(t)$; Right: $E^{v'_N}(t)$).

Remark 5.6. All numerical examples are designed for the first time. A great deal of effort has been made to solve these examples using NDSolve and DSolve commands of Wolframe Mathematica software, and even all of the options in the NDSolve and DSolve commands are tested for these examples. Since, numerical examples have a strong singularity, so they are not solvable with the numerical commands in Wolframe Mathematica software, and we only compare the results with the exact solutions. We solved five numerical examples to illustrate the efficiency of the present method.

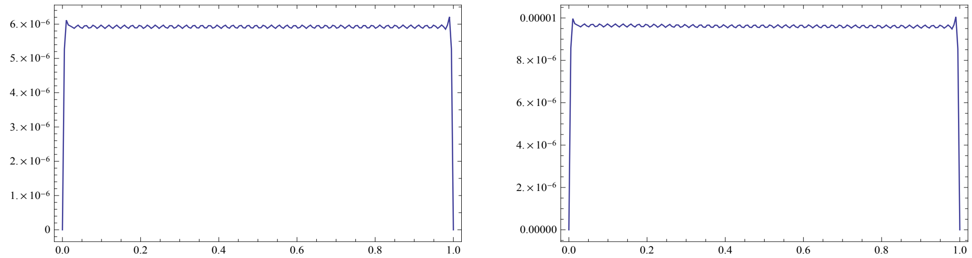

FIGURE 7. Absolute error for Example 5.4 with $N = 100$ and $n = 25$ (Left: $E^{u_N(t)}$; Right: $E^{v_N(t)}$).

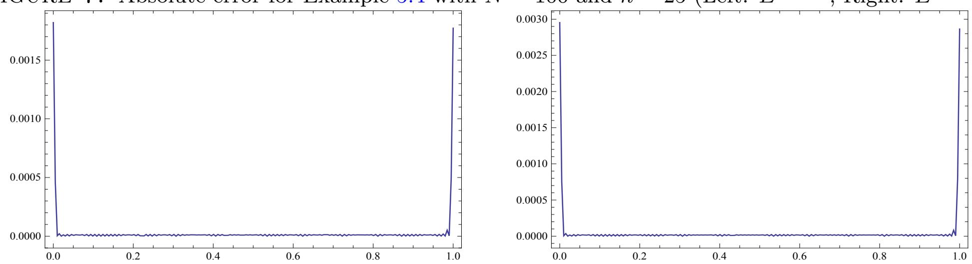

FIGURE 8. Absolute error for Example 5.4 with $N = 100$ and $n = 25$ (Left: $E^{u'_N(t)}$; Right: $E^{v'_N(t)}$).

TABLE 4. Max absolute error for Example 5.5 and convergence order $(Log_2^{E^N/E^{2N}})$

	Present Method			
$(\varepsilon_1, \varepsilon_2, \varepsilon_3 = 10^{-30})$	$N = 16$ $n = 15$	$N = 32$ $n = 20$	$N = 64$ $n = 25$	$N = 128$ $n = 30$
$E^{u_N(t)}$	2.30×10^{-3} 2.9386	3.00×10^{-4} 2.98089	3.80×10^{-5} 2.98489	4.80×10^{-6}
$E^{v_N(t)}$	1.8×10^{-3} 3.40439	1.70×10^{-4} 2.88583	2.30×10^{-5} 2.98751	2.90×10^{-6}
$E^{z_N(t)}$	5.60×10^{-4} 2.8255	8.00×10^{-5} 2.00	2.00×10^{-5} 3.32193	2.00×10^{-6}
$E^{u'_N(t)}$	2.00×10^{-1} 2.78588	2.90×10^{-2} 2.00998	7.30×10^{-3} 2.0199	1.80×10^{-3}
$E^{v'_N(t)}$	6.00×10^{-2} 1.81943	1.70×10^{-2} 1.88583	4.60×10^{-3} 1.9386	1.20×10^{-3}
$E^{z'_N(t)}$	2.90×10^{-2} 1.85798	8.00×10^{-3} 1.92961	2.10×10^{-3} 1.66985	6.60×10^{-4}

6. Conclusion

In this paper, we have solved SSPBVP using a reliable new technique based on the RKM and general method for nonlinear problems. The convenient implementation of the linear or nonlinear system of differential equations is one of the advantages of the present method. Numerical examples demonstrated the accuracy of the present method is high and it is practical for the different types of linear or nonlinear systems of differential equations. However, we provide error estimations for the approximate solution and its derivative. Therefore from the theoretical results, the convergence order for the approximate solution and its derivative are at least $O(h^2)$ and $O(h)$ respectively.

It is important to note here that many of the singularly perturbed problems with severe boundary layer behaviors are solvable with high precision using numerical methods in mathematical software which are freely available (such

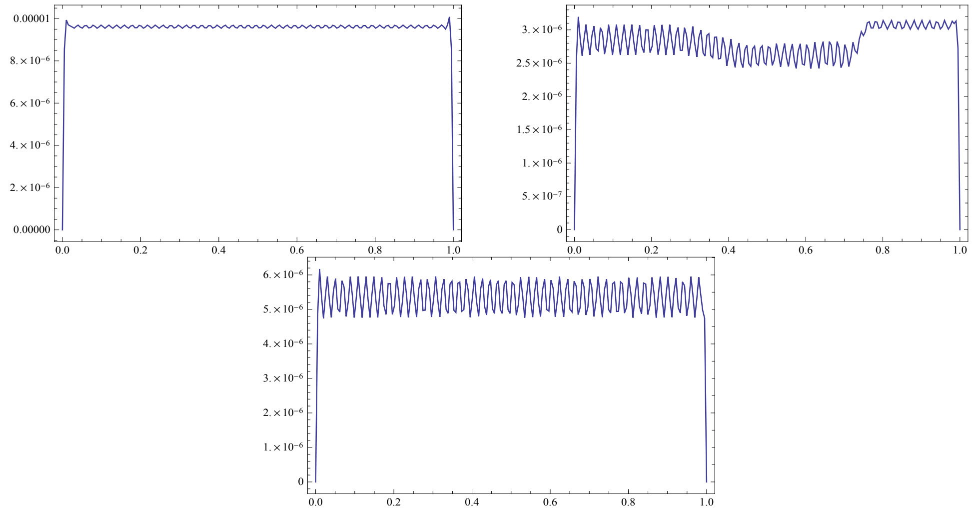

FIGURE 9. Absolute error for Example 5.5 with $N = 100$ and $n = 30$ (Left: $E^{u_N(t)}$; Middle: $E^{v_N(t)}$; Right: $E^{z_N(t)}$).

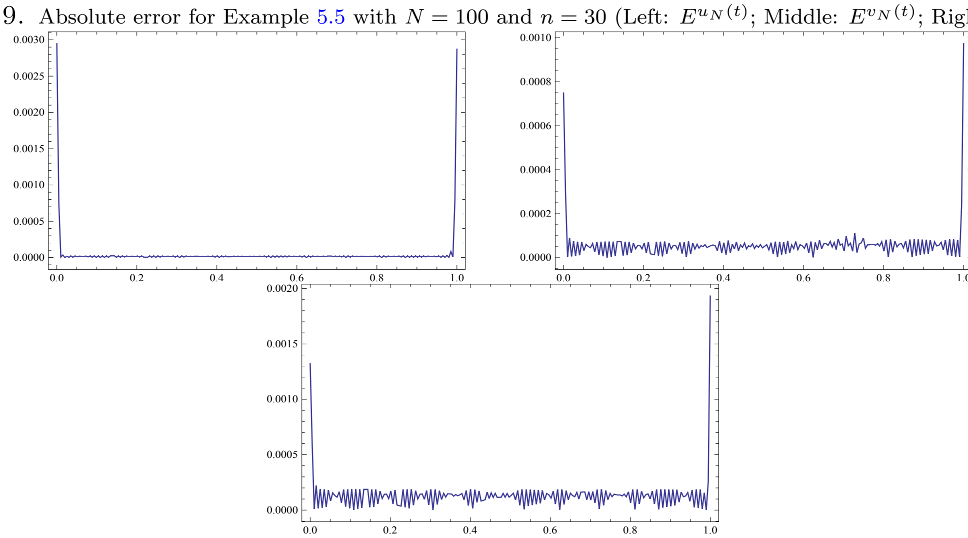

FIGURE 10. Absolute error for Example 5.5 with $N = 100$ and $n = 30$ (Left: $E^{u'_N(t)}$; Middle: $E^{v'_N(t)}$; Right: $E^{z'_N(t)}$).

as the NDSolve command in Wolfram Mathematica). Since the considered problem (1.1) has very severe singularity without boundary layer behavior, this software is not currently able to solve the problem (1.1), therefore for very severe singularities in singularly perturbed problem without layer behavior, we recommend using the present method.

Acknowledgment

We thank the anonymous reviewers for helpful comments, which lead to definite improvement in the manuscript.

References

[1] S. Abbasbandy, H. Sahihi, and T. Allahviranloo, *Implementing reproducing kernel method to solve singularly perturbed convection-diffusion parabolic problems*, Math. Model. Anal., *26* (2021), 116–134.

[2] A. Akgül, M. Inc, and E. Karatas, *Reproducing Kernel Functions for Difference Equations*, Disc. Cont. Dynam. System. Serie. S., *8* (2015), 1055–1064.

[3] A. Akgül, M. Inc, E. Karatas, and D. Baleanu, *Numerical solutions of fractional differential equations of Lane-Emden type by an accurate technique*, Advan. Diff. Equ., *220* (2015), 1–12.

[4] A. Akgül, E. K. Akgül, D. Baleanu, and M. Inc, *New Numerical Method for Solving Tenth Order Boundary Value Problems*, Math., *6* (2018), 1–9.

[5] E. K. Akgül, A. Akgül, Y. Khan, and D. Baleanu, *Representation for the reproducing kernel Hilbert space method for a nonlinear system*, Math. Statis., *3* (2019), 1345–1355.

[6] A. Akgül and E. K. Akgül, *A Novel Method for Solutions of Fourth-Order Fractional Boundary Value Problems*, Fractal. Fractional., *3* (2019), 1–13.

[7] A. Akgül, E. K. Akgül, and S. Korhan, *A New reproducing kernel functions in the reproducing kernel Sobolev spaces*, AIMS. Math., *5* (2020), 482–496.

[8] T. Allahviranloo and H. Sahihi, *Reproducing kernel method to solve parabolic partial differential equations with nonlocal conditions*, Num. Method. Partial. Diff. Equ., *36* (2020), 1758–1772.

[9] T. Allahviranloo and H. Sahihi, *Reproducing kernel method to solve fractional delay differential equations*, Appl. Math. Comput., *400* (2021), 126095.

[10] K. Atkinson and W. Han, *Theoretical Numerical Analysis A Functional Analysis Framework*, Third Edition, Springer Science, New York, USA, (2009).

[11] E. Babolian, S. Javadi, and E. Moradi, *Error analysis of reproducing kernel Hilbert space method for solving functional integral equations*, J. Comput. Appl. Math., *300* (2016), 300–311.

[12] E. Babolian and D. Hamedzadeh, *A splitting iterative method for solving second kind integral equations in reproducing kernel spaces*, J. Comput. Appl. Math., *326* (2017), 204–216.

[13] Z. Chen and Z. J. Chen, *The exact solution of system of linear operator equations in reproducing kernel spaces*, Appl. Math. Comput., *203* (2008), 56–61.

[14] M. Cui and Y. Lin *Nonlinear numerical analysis in the reproducing kernel space*, Nova Science, Hauppauge, New York, United States, (2009).

[15] P. Das and S. Natesan, *Optimal error estimate using mesh equidistribution technique for singularly perturbed system of reaction-diffusion boundary-value problems*, Appl. Math. Comput., *249* (2014), 265–277.

[16] F. Z. Geng and M. G. Cui, *Solving a nonlinear system of second order boundary value problems*, J. Math. Anal. Appl., *327* (2007), 1167–1181.

[17] R. K. Ghaziani, M. Fardi, and M. Ghasemi, *Solving multi-order fractional differential equations by reproducing kernel Hilbert space method*, Comput. Method. Diff. Equ., *4* (2016), 170–190.

[18] W. Jiang and Z. Chen, *solving a system of linear Volterra integral equations using the new reproducing kernel method*, Appl. Math. Comput., *219* (2013), 10225–10230.

[19] Y. Kan-On and M. Mimura, *Singular perturbation approach to a 3-component reaction-diffusion system arising in population dynamics*, SIAM. J. Math. Anal., *29* (1998), 1519–1536.

[20] R. Ketabchi, R. Mokhtari, and E. Babolian, *Some error estimates for solving Volterra integral equations by using the reproducing kernel method*, J. Comput. Appl. Math., *273* (2015), 245–250.

[21] X. Li, Y. Gao, and B. Wu, *Mixed reproducing kernel-based iterative approach for nonlinear boundary value problems with nonlocal conditions*, Comput. Method. Diff. Equ., *9*(3) (2020), 649-658, DOI: 10.22034/CMDE.2020.38153.1681.

[22] Y. Z. lin, Y. L. Wang, F. G. Tan, X. H. Wan, H. Yu, and J. S. Duan, *Solving a class of linear nonlocal boundary value problems using the reproducing kernel*, Appl. Math. Comput., *265* (2015), 1098–1105.

[23] X. Lu and M. G. Cui, *Solving a singular system of two nonlinear ODEs*, Appl. Math. Comput., *198* (2008), 534–543.

[24] N. Madden, M. Stynes, and G. P. Thomas, *On the application of robust numerical methods to a complete-flow wave-current model, in: Boundary and Interior Layers (BAIL)*, Toulouse, Hauppauge, New York, United States, (1998).

[25] L. Mei and Y. Lin, *Simplified reproducing kernel method and convergence order for linear Volterra integral equations with variable coefficients*, J. Comput. Appl. Math., *346* (2019), 390–398.

[26] S. Matthews, E. O'Riordan, and G. I. Shishkin, *A numerical method for a system of singularly perturbed reaction-diffusion equations*, J. Comput. Appl. Math., *145* (2002), 151–166.
[27] D.S. Naidu and K. A. Rao, *Singular perturbation analysis of the closed-loop discrete optimal control problem*, Opt. Cont. Appl. Method., *5* (1984), 19–37.
[28] S. Natesan and N. Ramanujam, *A booster method for singular perturbation problems arising in chemical reactor theory by incorporation of asymptotic approximations*, Appl. Math. Comput., *100* (1999), 27–48.
[29] S. Natesan and B. S. Deb, *A robust computational method for singularly perturbed coupled system of reaction-diffusion boundary-value problems*, Appl. Math. Comput., *188* (2007), 353–364.
[30] W. H. Ruan and C. V. Pao, *Asymptotic behavior and positive solutions of a chemical reaction diffusion system*, J. Math. Anal. Appl., *159* (1992), 157–178.
[31] H. Sahihi, S. Abbasbandy, and T. Allahviranloo, *Reproducing kernel method for solving singularly perturbed differential-difference equations with boundary layer behavior in Hilbert space*, J. Comput. Appl. Math., *328* (2018), 30–43.
[32] H. Sahihi, S. Abbasbandy, and T. Allahviranloo, *Computational method based on reproducing kernel for solving singularly perturbed differential-difference equations with a delay*, Appl. Math. Comput., *361* (2019), 583–598.
[33] H. Sahihi, T. Allahviranloo, and S. Abbasbandy, *Solving system of second-order BVPs using a new algorithm based on reproducing kernel Hilbert space*, Appl. Num. Math., *151* (2020), 27–39.
[34] N. Sharp and M. Trummer, *A Spectral Collocation Method for Systems of Singularly Perturbed Boundary Value Problems*, Proc. Comput. Sci., *1080* (2017), 725–734.
[35] Y. Wang, T. Chaolu, and P. Jing, *New algorithm for second-order boundary value problems of integro-differential equation*, J. Comput. Appl. Math., *229* (2009), 1–6.
[36] Y. Wang, L. Su, X. Cao, and X. Li, *Using reproducing kernel for solving a class of singularly perturbed problems*, Comput. Math. Appl., *61* (2011), 421–430.
[37] Y. Wang, L. Su, and Z. Chen, *Using reproducing kernel for solving a class of singular weakly nonlinear boundary value problems*, Int. J. Comput. Math., *87* (2010), 367–380.
[38] L. H. Yang, J. H. Shen, and Y. Wang, *The reproducing kernel method for solving the system of the linear Volterra integral equations with variable coefficients*, J. Comput. Appl. Math., *236* (2012), 2398–2405.