Hindawi Publishing Corporation
Journal of Applied Mathematics
Volume 2014, Article ID 340567, 8 pages
http://dx.doi.org/10.1155/2014/340567

Research Article

Semidefinite Optimization Providing Guaranteed Bounds on Linear Functionals of Solutions of Linear Integral Equations with Smooth Kernels

Guangming Zhou, Chao Deng, and Kun Wu

School of Mathematics and Computational Science, Xiangtan University, Xiangtan, Hunan 411105, China

Correspondence should be addressed to Guangming Zhou; zhougmxtu@163.com

Received 2 November 2013; Accepted 20 January 2014; Published 3 March 2014

Academic Editor: Ning Hu

Based on recent progress on moment problems, semidefinite optimization approach is proposed for estimating upper and lower bounds on linear functionals defined on solutions of linear integral equations with smooth kernels. The approach is also suitable for linear integrodifferential equations with smooth kernels. Firstly, the primal problem with smooth kernel is converted to a series of approximative problems with Taylor polynomials obtained by expanding the smooth kernel. Secondly, two semidefinite programs (SDPs) are constructed for every approximative problem. Thirdly, upper and lower bounds on related functionals are gotten by applying SeDuMi 1.1R3 to solve the two SDPs. Finally, upper and lower bounds series obtained by solving two SDPs, respectively infinitely approach the exact value of discussed functional as approximative order of the smooth kernel increases. Numerical results show that the proposed approach is effective for the discussed problems.

1. Introduction

Semidefinite optimization has been successfully applied to deal with many important problems [1–10] since it was proposed in 1963 [11]. Recently, the authors in [10] presented the semidefinite optimization method for obtaining guaranteed bounds on linear functionals defined on solutions of linear differential equations with polynomial coefficients. Instead of directly handling linear differential equations, the approach gets the discussed bounds by solving SDPs based on these equations and related functionals. Their numerical results are very encouraging. The authors in [12] proposed the semidefinite optimization method for estimating bounds on linear functionals of solutions of linear integral and integrodifferential equations with polynomial kernels. The method does not directly solve these equations as successive approximations method, Runge-Kuta one, direct computation one, the Adomian decomposition one, the modified Adomian decomposition one, the variational iterative one, and so forth in the references [13–21]. Numerical results show that the proposed method can get guaranteed bounds on the discussed functionals.

In this paper, we will extend the polynomial kernels in [12] to the generally smooth kernels and propose semidefinite optimization method for providing guaranteed bounds on linear functionals defined on solutions of Fredholm and Volterra integral equations with generally smooth kernels. Firstly, we expand smooth kernel in Fredholm or Volterra integral equation as a series of Taylor polynomials, and then the Fredholm or Volterra integral equation with the smooth kernel is converted to a series of integral equations with polynomial kernels. Secondly, semidefinite programs (SDPs) are constructed based on these approximative equations and discussed functionals. Thirdly, we apply SeDuMi 1.1R3 [22] to solve these SDPs and get upper and lower bounds sequences which all converge to the exact value of related functional. Finally, we illustrate the effectiveness of the method by carrying out some numerical experiments.

The rest of this paper is organized as follows. In Section 2, we propose semidefinite optimization method for estimating guaranteed bounds on the linear functionals defined on solution of Volterra integral equation of the second kernel with smooth kernel. In Section 3, four numerical examples

are tested. We end the paper with some conclusions and discussions in the final section.

2. Semidefinite Optimization Method

In this section, we propose semidefinite optimization method for estimating guaranteed bounds on linear functionals defined on solutions of Volterra integral equation of the second kind with smooth kernel.

Throughout the work, we suppose that related integral equations make unique solutions exist in the distribution space $\mathscr{A}$, in which the polynomial ring $\mathbb{R}[x]$ is dense.

Primal Problem. Computing

$$\int_0^1 x^i \phi(x) \tag{1}$$

in which $\phi(x)$ satisfies

$$\phi(x) = f(x) + \int_0^x K(x,s)\phi(s)\,ds, \tag{2}$$

where $f(x)$ and the kernel $K(x,s)$ are given in advance, and the former is an integrable function on the interval $[0,+\infty)$, and the latter is an infinitely smooth function in variables x and s.

Equation (2) is called Volterra integral equation of the second kind [17].

We expand the kernel $K(x,s)$ as the following Taylor polynomial with orders n at $x=0$ and $s=0$:

$$K(x,s) = K_n(x,s) + R_n(x,s), \tag{3}$$

where

$$\begin{aligned} K_n(x,s) = {} & K(0,0) + \left(x\frac{\partial}{\partial x} + s\frac{\partial}{\partial s}\right)K(0,0) \\ & + \frac{1}{2!}\left(x\frac{\partial}{\partial x} + s\frac{\partial}{\partial s}\right)^2 K(0,0) \\ & + \cdots + \frac{1}{n!}\left(x\frac{\partial}{\partial x} + s\frac{\partial}{\partial s}\right)^n K(0,0), \end{aligned} \tag{4}$$

$$R_n(x,s) = \frac{1}{(n+1)!}\left(x\frac{\partial}{\partial x} + s\frac{\partial}{\partial s}\right)^{n+1} K(\theta x, \theta s), \quad 0 \le \theta \le 1, \tag{5}$$

$$\left(x\frac{\partial}{\partial x} + s\frac{\partial}{\partial s}\right)^i K(0,0) = \sum_{r=0}^{i} C_i^r \left(\left.\frac{\partial^i K(x,s)}{\partial x^{i-r}\partial s^r}\right|_{x=0,s=0}\right) x^{i-r}s^r. \tag{6}$$

Therefore, (2) can be approximated by the following integral equation:

$$\phi(x) = f(x) + \int_0^x K_n(x,s)\phi(s)\,ds, \tag{7}$$

in which $K_n(x,s)$ is of the form (4).

For simplicity, we rewrite $K_n(x,s)$ in (4) as

$$K_n(x,s) = \sum_{i=0}^{n}\left(\sum_{j=0}^{n} h_{ij}x^j\right)s^i. \tag{8}$$

Further, *Primal Problem* can be written as the approximative form.

Approximative Problem. Compute

$$\int_0^1 x^i \phi(x), \tag{9}$$

where $\phi(x)$, satisfies (7), in which $f(x)$ is just the one in (2) and the kernel $K_n(x,s)$ is defined by (8).

In [12], semidefinite optimization method has been proposed for estimating bounds on linear functionals defined on solution of linear integral equation with polynomial kernel. Now we present semidefinite optimization method for providing guaranteed bounds on the linear functional (1) in *Primal Problem*, which is a generalized form of the method proposed in [12].

Algorithm 1.

Step 0. Let $n=1$, $N=1$ and give a tolerance ϵ.

Step 1. Convert *Primal Problem* to *Approximative Problem* according to the above analysis.

Step 2. Generate linear equality constraints.

Suppose that the solution $\phi(x)$ of (7) is bounded from below; that is, there exists c such that

$$\phi(x) \ge c, \quad \forall x \in [0,1]. \tag{10}$$

We define

$$m_i = \int_0^1 x^i(\phi(x) - c), \quad i = 0,1,2,\ldots \tag{11}$$

which may be called moments even though $\phi(x)$ may not be a probability distribution.

Multiplying (2) by the testing functions $\tau(x) = x^l$, $l = 0,1,2,\ldots$ and integrating it over the interval $[0,1]$, we can get

$$\begin{aligned} & \int_0^1 x^l\phi(x) \\ & = \int_0^1 x^l f(x) + \int_0^1\left(x^l\int_0^x K_n(x,s)\phi(s)\,ds\right)dx \\ & = \int_0^1 x^l f(x) + \sum_{i=0}^{n}\int_0^1\left(\int_0^x\left(\sum_{j=0}^{n} h_{ij}x^{j+l}\right)s^i\phi(s)\,ds\right)dx \\ & = \int_0^1 x^l f(x) + \sum_{i=0}^{n}\int_0^1 s^i\phi(s)\,ds\int_s^1\left(\sum_{j=0}^{n} h_{ij}x^{j+l}\right)dx \\ & = \int_0^1 x^l f(x) + \sum_{i=0}^{n}\sum_{j=0}^{n}\frac{h_{ij}}{j+l+1} \\ & \quad \times\left(\int_0^1 s^i\phi(s)\,ds - \int_0^1 s^{i+j+l+1}\phi(s)\,ds\right). \end{aligned} \tag{12}$$

By (11), we obtain

$$\int_0^1 x^i \phi(x) = m_i + \frac{c}{i+1}. \tag{13}$$

Substituting (13) into (12), we can get

$$m_l + \sum_{i=0}^{n} \alpha_{il} m_i + \sum_{i=0}^{n}\sum_{j=0}^{n} \gamma_{ijl} m_{i+j+l+1} = \beta_l, \tag{14}$$

in which

$$\alpha_{il} = -\sum_{j=0}^{n} \frac{h_{ij}}{j+l+1},$$

$$\gamma_{ijl} = \frac{h_{ij}}{j+l+1}, \tag{15}$$

$$\beta_l = \sum_{i=0}^{n}\sum_{j=0}^{n} \frac{c h_{ij}}{(i+1)(i+j+l+2)} - \frac{c}{l+1} + \int_0^1 x^l f(x).$$

Because the solution $\phi(x)$ of (2) is in the distribution space $\mathscr{A}$, in which $\mathbb{R}[x]$ is dense, (7) can be transformed into the system which consists of the equations as described by (14) where $l = 0, 1, 2, \ldots$.

Step 3. Generate semidefinite constraints.

It is obvious that $x \in [0,1]$ is equivalent to $x \in \{x \mid f_1(x) = x \geq 0, f_2(x) = 1 - x \geq 0\}$.

Denote

$$\mathbf{F}_M(x) = \begin{bmatrix} 1 & x & x^2 & \cdots & x^M \end{bmatrix}', \tag{16}$$

where M is a nonnegative integer.

By the method in [10], for $x \in [0,1]$, we have

$$\int_0^1 (\phi(x) - c)\, \mathbf{F}_M(x)\, \mathbf{F}_M(x)' \prod_{j\in\mathscr{I}} f_j(x) \succeq 0, \quad \mathscr{I} \subseteq \{1,2\}. \tag{17}$$

We can obtain the following positive semidefinite matrices:

$$\begin{aligned} Q_0 &\succeq 0, & Q_1 &\succeq 0, \\ Q_0 - Q_1 &\succeq 0, & Q_1 - Q_2 &\succeq 0, \end{aligned} \tag{18}$$

in which

$$Q_0 = \begin{pmatrix} m_0 & m_1 & \cdots & m_M \\ m_1 & m_2 & \cdots & m_{M+1} \\ \vdots & \vdots & \ddots & \vdots \\ m_M & m_{M+1} & \cdots & m_{2M} \end{pmatrix},$$

$$Q_1 = \begin{pmatrix} m_1 & m_2 & \cdots & m_{M+1} \\ m_2 & m_3 & \cdots & m_{M+2} \\ \vdots & \vdots & \ddots & \vdots \\ m_{M+1} & m_{M+2} & \cdots & m_{2(M+1)} \end{pmatrix}, \tag{19}$$

$$Q_2 = \begin{pmatrix} m_2 & m_3 & \cdots & m_{M+2} \\ m_3 & m_4 & \cdots & m_{M+3} \\ \vdots & \vdots & \ddots & \vdots \\ m_{M+2} & m_{M+3} & \cdots & m_{2(M+2)} \end{pmatrix},$$

where $M = 2n + N + 1$ in the three matrices Q_0, Q_1, and Q_2, by replacing $\int_0^1 (\phi(x) - c) x^i$ in (17) with m_i and setting the four subsets of the set $\{1,2\}$ to $\mathscr{I}$ in (17), respectively.

Step 4. Construct two SDPs.

Assuming that the testing function with the highest degree is x^N, we get the following two SDPs:

$$\begin{aligned} &\max / \min && m_i + \frac{c}{i+1}, \\ &\text{s.t.} && (14), \quad l = 0, 1, \ldots, N, \\ & && (18), \end{aligned} \tag{20}$$

where decision variables are m_i, $i = 0, 1, \ldots, 2(M+2)$.

Step 5. Apply SeDuMi 1.1R3 to solve the two SDPs.

Denote by $m_i^{(\max,N,n)}$ the decision variable m_i obtained by solving the above maximizing programming. $m_i^{(\min,N,n)}$ has similar meaning.

Step 6. Define whether the highest degree N of the testing function increases or not.

When $N = 1$, go to Step 7; or if

$$\begin{aligned} \left| m_i^{(\max,N,n)} - m_i^{(\max,N-1,n)} \right| &\leq \epsilon, \quad i = 0, 1, \ldots, 2n+N+1, \\ \left| m_i^{(\min,N,n)} - m_i^{(\min,N-1,n)} \right| &\leq \epsilon, \quad i = 0, 1, \ldots, 2n+N+1 \end{aligned} \tag{21}$$

all hold, go to Step 7, or let $N := N + 1$ and go to Step 4.

Step 7. Judge whether iteration goes on or not.

When $n = 1$, go to Step 2, or if

$$\begin{aligned} \left| m_i^{(\max,N,n)} - m_i^{(\max,N,n-1)} \right| &\leq \epsilon, \quad i = 0, 1, \ldots, 2n+N+1, \\ \left| m_i^{(\min,N,n)} - m_i^{(\min,N,n-1)} \right| &\leq \epsilon, \quad i = 0, 1, \ldots, 2n+N+1 \end{aligned} \tag{22}$$

all hold, stop the iteration and output $m_i^{(\max,N,n)}$ and $m_i^{(\min,N,n)}$, which are upper and lower bounds of $\int_0^1 x^i \phi(x)$, respectively; or let $n := n + 1$ and go to Step 1.

Remark 2. Obviously, $x \in [0,1]$ in Algorithm 1 can extend to $x \in [a,b]$ with $a < b$. Of course, some necessary modifications must be done.

Remark 3. The proposed method is also suitable for other linear integral and integrodifferential equations with smooth kernels.

Remark 4. In general, c in (5) is unknown. But we do not need infimum of $\phi(x)$ over $[0,1]$, so we can set a small value to c.

Remark 5. The semidefinite constraints (18) in Algorithm 1 only depend on the integral interval $[0,1]$. The moments $m_{2n+N+2}, \ldots, m_{2(2n+N+3)}$ in the semidefinite constraints (18) do not appear in the linear constraints. They are extension moments (see [9] or [10] for details).

Remark 6. In practical applications, for reducing computation amounts of Algorithm 1, we usually set two suitable positive integers to n and N, respectively.

Remark 7. In some cases, $K(x,s)$ in (2) can also be expanded as Taylor polynomials in variables x and s with different orders, respectively.

3. Numerical Experiments

In this section, we give four examples to illustrate the effectiveness of Algorithm 1. For simplicity, the interval $[0,1]$ is always taken as the integral interval in these examples.

3.1. Volterra Integral Equation of the First Kind with Smooth Kernel.

Example 1. Computing

$$\int_0^1 x^i \phi(x), \tag{23}$$

where $\phi(x)$, satisfies

$$\begin{aligned}\int_0^x \left(\cos x + e^t\right)\phi(t)\,dt \\ = -\frac{1}{2} + \frac{1}{2}e^x \cos x + \frac{1}{2}e^x \sin x + \sin x \cos x.\end{aligned} \tag{24}$$

The exact solution of (24) is $\phi(x) = \cos x$.

Multiplying (24) by x^i and integrating it over the interval $[0,1]$, we can get

$$\begin{aligned}\int_0^1 \left(\int_0^x x^i \left(\cos x + e^t\right)\phi(t)\,dt\right)dx \\ = \int_0^1 \left(\frac{1}{2} + \frac{1}{2}e^x \cos x + \frac{1}{2}e^x \sin x + \sin x \cos x\right)x^i.\end{aligned} \tag{25}$$

In this example, $u(x) \geq 0$ for all $x \in [0,1]$. Zero is set to c in (11). Then we define

$$m_i = \int_0^1 x^i \phi(x). \tag{26}$$

Taylor polynomial with degree 6 of $\cos x$ at $x = 0$ and Taylor polynomial with degree 7 of e^t at $t = 0$ are as follows:

$$\begin{aligned}\cos x \approx g_1(x) = a_0 + a_1 x^2 + a_2 x^4 + a_3 x^6, \\ e^t \approx g_2(t) = b_0 + b_1 t + b_2 t^2 + b_3 t^3 \\ + b_4 t^4 + b_5 t^5 + b_6 t^6 + b_7 t^7,\end{aligned} \tag{27}$$

where $a_0 = 1$, $a_1 = -1/2$, $a_2 = 1/24$, $a_3 = -1/720$, $b_0 = 1$, $b_1 = 1$, $b_2 = 1/2$, $b_3 = 1/6$, $b_4 = 1/24$, $b_5 = 1/120$, $b_6 = 1/720$, and $b_7 = 1/5040$, respectively.

Table 1: Upper and lower bounds for Example 1 for $N = 28$.

UN-Ob	LLFoS	ULFoS	ELFoS	Error
m_0	0.842	0.855	0.841	0.014
m_1	0.382	0.396	0.382	0.014
m_2	0.240	0.253	0.239	0.014
m_3	0.172	0.186	0.172	0.014
m_4	0.134	0.147	0.133	0.014
m_5	0.109	0.122	0.108	0.014
m_6	0.092	0.105	0.091	0.014
m_7	0.079	0.092	0.078	0.014
m_8	0.069	0.083	0.069	0.014
m_9	0.062	0.075	0.061	0.014
m_{10}	0.056	0.069	0.055	0.014
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
m_{20}	0.028	0.042	0.029	0.013

Substituting (27) into (25) and replacing $\int_0^1 t^i \phi(t)dt$ with m_i, we can obtain

$$\begin{aligned}&\left(\frac{a_0}{i+1} + \frac{a_1}{i+3} + \frac{a_2}{i+5} + \frac{a_3}{i+7} + \frac{b_0}{i+1}\right)m_0 \\ &\quad + \frac{b_1}{i+1}m_1 + \frac{b_2}{i+1}m_2 + \frac{b_3}{i+1}m_3 \\ &\quad + \frac{b_4}{i+1}m_4 + \frac{b_5}{i+1}m_5 + \frac{b_6}{i+1}m_6 + \frac{b_7}{i+1}m_7 \\ &\quad - \left(\frac{b_0}{i+1} + \frac{a_0}{i+1}\right)m_{i+1} - \frac{b_1}{i+1}m_{i+2} \\ &\quad - \left(\frac{b_2}{i+1} + \frac{a_1}{i+3}\right)m_{i+3} - \frac{b_3}{i+1}m_{i+4} \\ &\quad - \left(\frac{b_4}{i+1} + \frac{a_2}{i+5}\right)m_{i+5} - \left(\frac{b_6}{i+1} + \frac{a_3}{i+7}\right)m_{i+7} \\ &\quad - \frac{b_5}{i+1}m_{i+6} - \frac{b_7}{i+1}m_{i+8} \\ &= \int_0^1 \left(\frac{1}{2} + \frac{1}{2}e^x \cos x + \frac{1}{2}e^x \sin x + \sin x \cos x\right)x^i.\end{aligned} \tag{28}$$

For $x \in [0,1]$, we know that the semidefinite constraints are the same as (18).

We construct the following two SDPs:

$$\begin{aligned}&\max/\min \quad m_i \\ &\text{s.t.} \quad (28), \quad i = 0, 1, \ldots, N, \\ &\qquad (18).\end{aligned} \tag{29}$$

Letting $N = 28$, we apply SeDuMi 1.1R3 to solve the max / min SDPs. The partial numerical results are reported in Table 1.

In Table 1, UN-Ob, LLFoS, ULFoS, and ELFoS mean decision variables or objective functions in the above two

Table 2: Upper and lower bounds for Example 1 for $N = 50$.

UN-Ob	LLFoS	ULFoS	ELFoS	Error
m_0	0.84148	0.84148	0.84147	0.00001
m_1	0.38180	0.38180	0.38177	0.00003
m_2	0.23916	0.23916	0.23913	0.00003
m_3	0.17176	0.17176	0.17174	0.00002
m_4	0.13309	0.13309	0.13308	0.00001
m_5	0.10823	0.10823	0.10822	0.00001
m_6	0.09100	0.09100	0.09098	0.00002
m_7	0.07839	0.07839	0.07837	0.00002
m_8	0.06878	0.06878	0.06877	0.00001
m_9	0.06123	0.06123	0.06122	0.00001
m_{10}	0.05515	0.05515	0.05514	0.00001
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
m_{20}	0.02750	0.02750	0.02750	0.00000
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
m_{30}	0.01826	0.01826	0.01826	0.00000
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
m_{40}	0.01366	0.01366	0.01366	0.00000

Table 3: Upper and lower bounds for Example 2 for $N = 50$.

UN-Ob	LLFoS	ULFoS	ELFoS	Error
m_0	1.333335	1.333335	1.333333	0.000000
m_1	0.916669	0.916669	0.916667	0.000002
m_2	0.700002	0.700002	0.700000	0.000002
m_3	0.566668	0.566668	0.566667	0.000001
m_4	0.476192	0.476192	0.476190	0.000002
m_5	0.410716	0.410716	0.410714	0.000002
m_6	0.361113	0.361113	0.361111	0.000002
m_7	0.322224	0.322224	0.322222	0.000002
m_8	0.290910	0.290910	0.290909	0.000001
m_9	0.265153	0.265153	0.265152	0.000000
m_{10}	0.243591	0.243591	0.243590	0.000000
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
m_{20}	0.134388	0.134388	0.134387	0.000000
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
m_{30}	0.092804	0.092804	0.092803	0.000000
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
m_{40}	0.070875	0.070875	0.070875	0.000000

Table 4: Upper and lower bounds for Example 3 for $N = 50$.

UN-Ob	LLFoS	ULFoS	ELFoS	Error
m_0	0.666762	0.666762	0.666667	0.000095
m_1	0.291711	0.291711	0.291667	0.000044
m_2	0.183362	0.183362	0.183333	0.000029
m_3	0.133354	0.133354	0.133333	0.000021
m_4	0.104778	0.104778	0.104762	0.000016
m_5	0.086323	0.086323	0.086310	0.000013
m_6	0.073424	0.073424	0.073413	0.000009
m_7	0.0638985	0.063985	0.063889	0.000014
m_8	0.0565740	0.0566574	0.056566	0.000008
m_9	0.050765	0.050765	0.050758	0.000007
m_{10}	0.046044	0.046044	0.046037	0.000003
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
m_{20}	0.023907	0.023907	0.023904	0.000003
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
m_{30}	0.016162	0.016162	0.016159	0.000003
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
m_{40}	0.012210	0.012210	0.012209	0.000001

Table 5: Upper and lower bounds for Example 4 for $N = 50$.

UN-Ob	LLFoS	ULFoS	ELFoS	Error
m_0	1.7185	1.7185	1.7183	0.0002
m_1	1.0002	1.0002	1.0000	0.0002
m_2	0.7185	0.7185	0.7183	0.0002
m_3	0.5636	0.5636	0.5634	0.0001
m_4	0.4647	0.4647	0.4645	0.0001
m_5	0.3957	0.3957	0.3956	0.0001
m_6	0.3448	0.3448	0.3447	0.0001
m_7	0.3056	0.3056	0.3055	0.0001
m_8	0.2744	0.2744	0.2744	0.0000
m_9	0.2491	0.2491	0.2490	0.0001
m_{10}	0.2281	0.2280	0.2280	0.0000
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
m_{20}	0.1238	0.1238	0.1238	0.0000
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
m_{30}	0.0850	0.0850	0.0850	0.0000
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
m_{40}	0.0648	0.0648	0.0648	0.0000

SDPs, lower bounds, upper bounds, and exact values on linear functionals defined on solution of (24), respectively. *Error* means |ULFoS − ELFoS|. These signs in Tables 2, 3, 4, and 5 have the same meanings. From Table 1, we can see that for every m_i, |ULFoS−ELFoS| is more than 10^{-2}, and these errors do not decrease as N increases. Maybe the case is resulted in by accumulative error.

In order to increase the precision of numerical results of Example 1, we first convert (24) to the following equivalent integral equation (30) and then apply Algorithm 1 to estimate (23), where $\phi(x)$ satisfies (30).

Differentiating both sides of (24) with respect to x gives

$$\begin{aligned}(\cos x + e^x)\,\phi(x) - \int_0^x \sin x\phi(t)\,dt \\ = \cos x \cdot e^x + \cos^2 x - \sin^2 x.\end{aligned} \tag{30}$$

Use the two polynomials

$$\cos x + e^x \approx g_1(x) = a_0' + a_1' x + a_2' x^3 + a_3' x^4 + a_4' x^5 + a_5' x^7, \quad (31)$$
$$\sin x \approx g_2(x) = b_0' x + b_1' x^3,$$

where $a_0' = 2$, $a_1' = 1$, $a_2' = 1/6$, $a_3' = 1/12$, $a_4' = 1/120$, $a_5' = 1/5040$, $b_0' = 0.9973$, and $b_1' = -0.1563$, to approximate $\cos x + e^x$ and $\sin x$, respectively.

Substituting (31) into (30), multiplying its both sides by x^i, and integrating it over the interval $[0, 1]$, we have

$$\int_0^1 g_1(x) x^i \phi(x) - \int_0^1 \left(x^i g_2(x) \int_0^x \phi(t)\, dt \right) dx = \int_0^1 \left(\cos x \cdot e^x + \cos^2 x - \sin^2 x \right) x^i. \quad (32)$$

Let

$$m_i = \int_0^1 x^i \phi(x). \quad (33)$$

Substituting (33) into (32) and simplifying it, we get

$$-\left(\frac{b_0'}{i+2} + \frac{b_1'}{i+4} \right) m_0 + a_0' m_i + a_1' m_{i+1} + b_0' m_{i+2} + a_3' m_{i+3} + b_1' m_{i+4} + a_4' m_{i+5} + a_5' m_{i+7} = \int_0^1 \left(\cos x \cdot e^x + \cos^2 x - \sin^2 x \right) x^i. \quad (34)$$

According to $x \in [0, 1]$, we have semidefinite optimization as stated by (18).

We construct the SDPs:

$$\begin{aligned} &\max / \min \quad m_i \\ &\text{s.t.} \quad (34), \quad i = 0, 1, \ldots, N, \\ &\qquad (18). \end{aligned} \quad (35)$$

Letting $\epsilon = 10^{-8}$ and $N = 50$, we apply SeDuMi 1.1R3 to solve the above SDPs. Some numerical results are listed in Table 2.

From Table 2 we can see that the numerical results are accurate to four decimal points of the related exact values. Numerical results in Table 2 show that the proposed approach can efficiently estimate upper and lower bounds on the linear functionals $\int_0^1 x^i \phi(x)$ defined on solution of Volterra integral equation of the first kind. If we want to increase precision of the numerical results, we can reach the goal by expanding $\cos x$ and e^t in the kernel of (24) as Taylor polynomials with higher degrees. We also solved the max/min programs when $N = 49$. Numerical results show that $|m_j^{(\max,50,7)} - m_j^{(\max,49,7)}|$ and $|m_j^{(\min,50,7)} - m_j^{(\min,49,7)}|$, $j = 0, 1, \ldots, 40$, are all less than 10^{-9}. So the examples are as follows.

3.2. Volterra Integral Equation of the Second Kind with Smooth Kernel

Example 2. Computing

$$\int_0^1 x^i \phi(x), \quad (36)$$

where $\phi(x)$, satisfies

$$\phi(x) = \int_0^x \left(1 + xe^t\right) \phi(t)\, dt + 2x - \frac{1}{3}x^3 - x^3 e^x. \quad (37)$$

The exact solution of the equation is $\phi(x) = x^2 + 2x$. Define m_i as stated by (26). We expand e^t as follows:

$$e^t \approx a_0 + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4, \quad (38)$$

where $a_0 = 1$, $a_1 = 1$, $a_2 = 1/2$, $a_3 = 1/6$, and $a_4 = 1/24$.

Multiplying (37) in which xe^t is replaced with $x(a_0 + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4)$ by x^i and integrating it over the interval $[0, 1]$, we can get

$$\begin{aligned} &-\left(\frac{1}{i+1} + \frac{1}{i+2} \right) m_0 - \frac{a_1}{i+1} m_1 - \frac{a_2}{i+1} m_2 \\ &\quad - \frac{a_3}{i+1} m_3 - \frac{a_4}{i+1} m_4 - \frac{a_5}{i+1} m_5 \\ &\quad - \frac{a_6}{i+1} m_6 - \frac{a_7}{i+1} m_7 + m_i + \frac{1}{i+1} m_{i+1} \\ &\quad + \frac{a_0}{i+2} m_{i+2} + \frac{a_1}{i+2} m_{i+3} + \frac{a_2}{i+2} m_{i+4} \\ &\quad + \frac{a_3}{i+5} m_{i+1} + \frac{a_4}{i+2} m_{i+6} + \frac{a_5}{i+2} m_{i+7} \\ &\quad + \frac{a_6}{i+8} m_{i+1} + \frac{a_7}{i+2} m_{i+9} \\ &= \int_0^1 \left(2x - \frac{1}{3}x^3 - x^3 e^x \right) x^i. \end{aligned} \quad (39)$$

Because $x \in [0, 1]$, we obtain semidefinite constraints as stated by (18).

We construct the following SDPs:

$$\begin{aligned} &\max / \min \quad m_i \\ &\text{s.t.} \quad (39) \quad i = 0, 1, \ldots, N, \\ &\qquad (18). \end{aligned} \quad (40)$$

Letting $\epsilon = 10^{-8}$ and $N = 50$, we apply SeDuMi 1.1R3 to solve the above SDPs. The partial numerical upper and lower bounds are listed in Table 3.

Numerical results in Table 3 show that the proposed method is very effective for obtaining guaranteed upper and lower bounds whose precision can reach 10^{-5}.

3.3. Fredholm Integral Equation of the Second Kind with Smooth Kernel

Example 3. Computing

$$\int_0^1 x^i \phi(x), \tag{41}$$

where $\phi(x)$, satisfies

$$\phi(x) = \int_0^1 \left(x^2 \sin t + 4\cos t\right)\phi(t)\,dt + \frac{1-\cos 1}{2}x^2 - x - 3 + 2\sin 1. \tag{42}$$

The exact solution of (42) is $\phi(x) = (x^2/2) - x + 1$.

Chebyshev polynomial with degree 3 of $\sin t$ at $t = 0$ and Chebyshev polynomial with degree 4 of $\cos t$ at $t = 0$ are as follows:

$$\sin t \approx 0.9974t - 0.1563t^3, \qquad \cos t \approx 1.0005 - 0.5040t^2 + 0.0440t^4. \tag{43}$$

Define m_i as (26). Substituting (43) into (42), multiplying (42) by x^i, and integrating it over the interval $[0, 1]$, we have

$$m_i - \frac{0.9974}{i+3}m_1 + \frac{0.1563}{i+3}m_3 - \frac{4.002}{i+1}m_0 + \frac{2.016}{i+1}m_2 - \frac{0.176}{i+1}m_4 = \frac{1-\cos 1}{2(i+3)} - \frac{1}{i+2} - \frac{3-2\sin 1}{i+1}. \tag{44}$$

For $x \in [0, 1]$, we get semidefinite constraints as stated by (18).

We construct the following SDPs:

$$\begin{aligned} &\max / \min \quad m_i \\ &\text{s.t.} \quad (44), \quad i = 0, 1, \ldots, N, \\ &\qquad (18). \end{aligned} \tag{45}$$

Letting $\epsilon = 10^{-8}$ and $N = 50$, we apply SeDuMi 1.1R3 to solve the above SDPs and get numerical upper and lower bounds of related functionals which are partially listed in Table 4.

Obviously, the numerical results in Table 4 are accurate to four decimal points of the exact functional values.

3.4. Volterra Integrodifferential Equation with Smooth Kernel

Example 4. Computing

$$\int_0^1 x^i \phi(x), \tag{46}$$

where $\phi(x)$, satisfies

$$\phi'(x) - \int_0^x \left(e^x + 2\right)\phi(t)\,dt = 2 - e^{2x} \tag{47}$$

with boundary condition $\phi(0) = 1$.

The exact solution of (47) is $\phi(x) = e^x$.

Integrating (47) over the interval $[0, 1]$, we get

$$\phi(x) - \int_0^x \left(e^x + 2x - e^t - 2t\right)\phi(t)\,dt = 2x - \frac{e^{2x}}{2} + \frac{3}{2}. \tag{48}$$

Replace e^t in the kernel of (48) with (38), and do the similar replacement for e^x in the kernel. Applying SeDuMi 1.1R3 to solve the corresponding approximative problem which is just Volterra integral equation of the second kind, we get numerical results partially reported in Table 5.

Numerical results in Table 5 show that the semidefinite optimization method can provide guaranteed bounds on linear functionals defined on solution of (47). If we hope to obtain more accurate numerical results, we can expand e^t and e^x in the kernel of (47) as Taylor polynomials with higher degrees.

4. Conclusions and Discussions

In this paper, we have presented the semidefinite optimization method for providing guaranteed bounds on linear functionals defined on solutions of linear integral equations with smooth kernels. Four examples show that the proposed approach is effective for estimating bounds on linear integral and integrodifferential equations with smooth kernels. The proposed approach requires that the related integral equation is linear. It cannot be directly applied to solve the nonlinear integral equation. So next work is to improve the proposed method, so that it can handle nonlinear problems.

Conflict of Interests

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

Acknowledgments

The authors thank the anonymous reviewers for their helpful comments. This work is partially supported by the National Natural Science Foundation, Grants nos. 61202397 and 11301445, the Research Foundation of Education Bureau of Hunan Province, Grant no. 13B121, the Research Foundation for Doctoral Program of Higher Education of China Grant no. 20114301120001, and the Science and Technology Project of Hunan Province.

References

[1] F. Alizzdeh, *Combinatorial optimization with interior point methods and semidefinite matrices [Ph.D. thesis]*, University of Minnesota, Minneapolis, Minn, USA, 1991.

[2] F. Alizadeh, "Interior point methods in semidefinite programming with applications to combinatorial optimization," *SIAM Journal on Optimization*, vol. 5, no. 1, pp. 13–51, 1995.

[3] S.-P. Wu, S. Boyd, and L. Vandenberghe, "FIR filter design via semidefinite programming and spectral factorization," in

Proceedings of the 35th IEEE Conference on Decision and Control (CDC '96), vol. 1, pp. 271–276, Kobe, Japan, December 1996.

[4] M. L. Overton, "On minimizing the maximum eigenvalue of a symmetric matrix," *SIAM Journal on Matrix Analysis and Applications*, vol. 9, no. 2, pp. 256–268, 1988.

[5] H. J. Peng and L. K. Rasmussen, "The application of semidefinite programming for detection in CDMA," *IEEE Journal on Selected Areas in Communications*, vol. 19, no. 8, pp. 1442–1449, 2001.

[6] J. B. Lasserre, "Global optimization with polynomials and the problem of moments," *SIAM Journal on Optimization*, vol. 11, no. 3, pp. 796–817, 2001.

[7] J. B. Lasserre, "Bounds on measures satisfying moment conditions," *The Annals of Applied Probability*, vol. 12, no. 3, pp. 1114–1137, 2002.

[8] J. B. Lasserre, "Robust global optimization with polynomials," *Mathematical Programming A*, vol. 107, no. 1-2, pp. 275–293, 2006.

[9] J. B. Lasserre, *Moments, Positive Polynomials and Their Applications*, Imperial College Press, London, UK, 2010.

[10] D. Bertsimas and C. Caramanis, "Bounds on linear PDEs via semidefinite optimization," *Mathematical Programming A*, vol. 108, no. 1, pp. 135–158, 2006.

[11] R. Bellman and K. Fan, "On systems of linear inequalities in Hermitian matrix variables," in *Convexity: Proceedings of Symposia in Pure Mathematics*, pp. 1–11, American Mathematical Society, Providence, RI, USA, 1963.

[12] G. M. Zhou, C. Deng, and K. Wu, "Bounds on linear functionals of solutions of linear integral equations with polynomial kernels via semidefinite optimization," In press.

[13] C. Baker, *The Numerical Treatment of Integral Equations*, Oxford University Press, London, UK, 1977.

[14] A. J. Jerri, *Introduction to Integral Equations with Applications*, Wiley-Interscience, New York, NY, USA, 1999.

[15] R. P. Kanwal and K. C. Liu, "A Taylor expansion approach for solving integral equations," *International Journal of Mathematical Education in Science and Technology*, vol. 20, no. 3, pp. 411–414, 1989.

[16] A.-M. Wazwaz, *A First Course in Integral Equations*, World Scientific, Singapore, 1997.

[17] A.-M. Wazwaz, *Linear and Nonlinear Integral Equations: Methods and Applications*, Higher Education Press, Beijing, China, 2011.

[18] P. Linz, *Analytical and Numerical Methods for Volterra Equations*, SIAM, Philadelphia, Pa, USA, 1985.

[19] G. Adomian, *Solving Frontier Problems of Physics: The Decomposition Method*, Kluwer Academic, Boston, Mass, USA, 1994.

[20] A.-M. Wazwaz, *Partial Differential Equations and Solitary Waves Theory*, Higher Education Press, Beijing, China, 2009.

[21] J.-H. He, "Homotopy perturbation technique," *Computer Methods in Applied Mechanics and Engineering*, vol. 178, no. 3-4, pp. 257–262, 1999.

[22] J. F. Sturm, "SeDuMi version 1.1R3," The Advanced Optimization Laboratory at McMaster University, Ontario, Canada, 2006.