Commun. Korean Math. Soc. **32** (2017), No. 1, pp. 215–224
https://doi.org/10.4134/CKMS.c150245
pISSN: 1225-1763 / eISSN: 2234-3024

THE COMPUTATION OF POSITIVE SOLUTIONS FOR A BOUNDARY VALUE PROBLEM OF THE LINEAR BEAM EQUATION

Jun Ji and Bo Yang

Abstract. In this paper, we propose a method of order two for the computation of positive solutions to a boundary value problem of the linear beam equation. The method is based on the Power method for the eigenvector associated with the dominant eigenvalue and the Crout-like factorization algorithm for the banded system of linear equations. It is extremely fast due to the linear complexity of the linear system solver. Numerical result of a test problem is included.

1. Introduction

The boundary value problem

$$u^{(4)}(t) = g(t)f(u(t)), \quad 0 \le t \le 1, \tag{1.1}$$

$$u(0) = u''(0) = u'(1) = u'''(1) = 0, \tag{1.2}$$

arises in the study of elasticity. The equation (1.1) is often referred to as the beam equation. The BVP (1.1)-(1.2) describes the deflection of a beam under a certain force when the beam is simply supported at the end $t = 0$ and fastened with a sliding clamp at $t = 1$ [6]. This problem (and its generalizations) has been widely investigated by many researchers [3, 4, 5, 10, 13]. In particular, Graef and Yang [5] obtained sufficient conditions for existence and nonexistence of positive solutions to the problem. We note that there are no efficient algorithms in the literature for the positive solutions of this problem when such positive solutions do exist. We will devote this paper to designing an efficient algorithm for positive solutions of the BVP (1.1)-(1.2) when f is linear, i.e., $f(u(t)) = \lambda u(t)$. The novel techniques developed in [8] for positive solutions of the boundary value problems of the second order difference equation will be adopted to the new situation for such a task.

Received December 23, 2015.
2010 *Mathematics Subject Classification.* 39A10, 65F05, 65F15.
Key words and phrases. boundary value problem, Crout-like factorization algorithm, linear beam equation, positive eigenvector.

For the case when $f(u(t)) = \lambda u(t)$, the problem (1.1)-(1.2) was discretized in [7] as follows:

$$\begin{cases} \Delta^4 y_i = \lambda a_{i+2} y_{i+2}, \quad -1 \le i \le n-2, \\ y_0 = \Delta^2 y_{-1} = \Delta y_n = \Delta^3 y_{n-1} = 0, \end{cases} \tag{1.3}$$

where λ is a parameter, and the forward difference operator Δ is defined as $\Delta y_i = y_{i+1} - y_i$. The monotone behavior of the eigenvalues of the discrete beam problem (1.3) as $\{a_i\}$ changes was carried out in [7]. The linear beam equation $u^{(4)}(t) + p(t)u(t) = \lambda g(t)u(t)$ when the beam is fastened with sliding clamps at both ends, i.e., $u'(0) = u'''(0) = u'(1) = u'''(1) = 0$ was discretized as

$$\begin{cases} \Delta^4 y_i + b_{i+2} y_{i+2} = \lambda a_{i+2} y_{i+2}, \quad -1 \le i \le n-2, \\ \Delta y_0 = \Delta^3 y_{-1} = \Delta y_n = \Delta^3 y_{n-1} = 0, \end{cases}$$

and the monotone behavior of the eigenvalues of this discrete model when $b_i \ge 0$ for $1 \le i \le n$ with $\sum_{i=1}^n b_i > 0$ can be found in [9].

A simple calculation indicates that the error involved in the approximation of the discrete model (1.3) to the problem (1.1)-(1.2) is of order one, i.e., $O(h)$. Thus, the discrete model (1.3) is not a good choice for any numerical methods for positive solutions of the boundary value problems of the linear beam equation. In this paper we will introduce a variant of the discrete model (1.3) with a better approximation to the boundary conditions (1.2). The improved discrete model is of order two and based on this model, a numerical method for the computation of positive solutions of the boundary value problems of the linear beam equation will be introduced. The core idea of the method is the Power method for the eigenvector associated with the dominant eigenvalue and the Crout-like factorization algorithm for a five-banded system of linear equations involved. The method is extremely fast due to the linear computational complexity of the linear system solver. Numerical result of a test problem is included.

2. The method

The interval of interest, $[0, 1]$, is divided into n subintervals by specifying evenly spaced values of the independent variable, $t_0, t_1, t_2, \ldots, t_n$, with $t_0 = 0$ and $t_n = 1$. Each subinterval is of length $h = 1/n$ and $t_i = ih = i/n$. A simple calculation leads to the following:

$$\Delta^4 y_i / h^4 = y^{(4)}(t_{i+2}) + O(h^2), \tag{2.4}$$

$$\Delta^2 y_{-1} / h^2 = y''(0) + O(h^2), \tag{2.5}$$

$$\frac{y_{n+1} - y_{n-1}}{2h} = y'(1) + O(h^2), \tag{2.6}$$

$$\frac{y_{n+2} - 2y_{n+1} + 2y_{n-1} - y_{n-2}}{2h^3} = y'''(1) + O(h^2). \tag{2.7}$$

Ignoring error terms $O(h^2)$ in (2.6) and (2.7), we can use

$$y_{n+1} - y_{n-1} = 0 \quad \text{and} \quad y_{n+2} - 2y_{n+1} + 2y_{n-1} - y_{n-2} = 0 \tag{2.8}$$

to approximate $y'(1) = 0$ and $y'''(1) = 0$, respectively. The boundary conditions of (2.8) are equivalent to

$$y_{n+1} - y_{n-1} = y_{n+2} - y_{n-2} = 0.$$

Define $a_i = h^4 g(t_i)$ for $i = 1, 2, \ldots, n$. It is easily seen that the problem (1.1)-(1.2) with $f(u(t)) = \lambda u(t)$ can be approximated by

$$\begin{cases} \Delta^4 y_i = \lambda a_{i+2} y_{i+2}, \quad -1 \le i \le n-2, \\ y_0 = \Delta^2 y_{-1} = y_{n+1} - y_{n-1} = y_{n+2} - y_{n-2} = 0 \end{cases} \tag{2.9}$$

with an error of $O(h^2)$. Due to the fact that the boundary conditions in (2.9) can be written as

$$y_0 = 0,\ y_{-1} = -y_1,\ y_{n+1} = y_{n-1}, \quad \text{and} \quad y_{n+2} = y_{n-2}, \tag{2.10}$$

the problem (2.9) is equivalent to the linear system

$$(-\hat{D} + \lambda \hat{A})y = 0, \tag{2.11}$$

where $\hat{A} = \mathrm{diag}(a_1, a_2, \ldots, a_{n-1}, a_n/2)$ and $\hat{D}$ is a banded $n \times n$ matrix given by

$$\hat{D} = \begin{pmatrix} 5 & -4 & 1 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ -4 & 6 & -4 & 1 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 1 & -4 & 6 & -4 & 1 & \cdots & 0 & 0 & 0 & 0 \\ 0 & 1 & -4 & 6 & -4 & \cdots & 0 & 0 & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & 0 & \cdots & 1 & -4 & 6 & -4 & 1 \\ 0 & 0 & 0 & 0 & \cdots & 0 & 1 & -4 & 7 & -4 \\ 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 1 & -4 & 3 \end{pmatrix}. \tag{2.12}$$

Obviously, there is a one-to-one correspondence between the solution $(y_1, y_2, \ldots, y_{n-1}, y_n)^T$ to the problem (2.11) and the solution $(y_{-1}, y_0, y_1, \ldots, y_n, y_{n+1}, y_{n+2})^T$ to the problem (2.9) under the relationship (2.10). We will not distinguish them.

The matrix $\hat{D}$ shares the same structure as the matrix D of [7]:

$$D = \begin{pmatrix} 5 & -4 & 1 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ -4 & 6 & -4 & 1 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 1 & -4 & 6 & -4 & 1 & \cdots & 0 & 0 & 0 & 0 \\ 0 & 1 & -4 & 6 & -4 & \cdots & 0 & 0 & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & 0 & \cdots & 1 & -4 & 6 & -4 & 1 \\ 0 & 0 & 0 & 0 & \cdots & 0 & 1 & -4 & 6 & -3 \\ 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 1 & -3 & 2 \end{pmatrix}. \tag{2.13}$$

Notice that the only difference between $\hat{D}$ and D is the elements at positions $(n-1, n-1)$, $(n-1, n)$, $(n, n-1)$ and (n, n).

Let e_i be the i-th column of the identity matrix I of order n. Define the elementary matrix $P_i = I + e_{i-1}e_i^T$ for $i \geq 2$. It is easily seen that $AP_i = A + (Ae_{i-1})e_i^T$ is the matrix obtained by adding the $(i-1)$th column of A to the ith column of A and that $P_iA = A + e_{i-1}(e_i^TA)$ is the matrix obtained by adding the ith row of A to the $(i-1)$th row of A. Similarly, AP_i^T and P_i^TA are matrices obtained by adding the ith column of A to the $(i-1)$th column of A and by adding the $(i-1)$th row of A to the ith row of A, respectively.

Lemma 2.1. *The matrix $\hat{D}$ is positive definite and each element of $\hat{D}^{-1}$ is positive.*

Proof. It is seen from (2.7)-(2.8) of [7] in the proof of [7, Lemma 2.1] that $D = W^TW$, where

$$W = P_n^{-1}P_{n-1}^{-1}\cdots P_3^{-1}P_2^{-1}P_2^{-T}P_3^{-T}\cdots P_{n-1}^{-T}P_n^{-T}. \tag{2.14}$$

Define $V = \hat{D} - D$ whose first $(n-2)$ rows are zeros and the last two rows are

$$e_{n-1}^TV = (0,\ \ldots, 0, 1, -1) \quad \text{and} \quad e_n^TV = (0,\ \ldots,\ 0,\ -1,\ 1).$$

We can write

$$\hat{D} = D + V = W^TW + V = W^T(I + W^{-T}VW^{-1})W. \tag{2.15}$$

Note that

$$W^{-1} = W^{-T} = P_n^TP_{n-1}^T\cdots P_3^TP_2^TP_2P_3\cdots P_{n-1}P_n. \tag{2.16}$$

Performing a sequence of elementary column and row operations specified by (2.16) on both sides of V, we have

$$I + W^{-T}VW^{-1} = I + e_ne_n^T \tag{2.17}$$

which, together with (2.15), implies that $\hat{D}$ is positive definite. It is seen from (2.15) and (2.17) that

$$\hat{D}^{-1} = W^{-1}(I + e_ne_n^T)^{-1}W^{-T} = W^{-1}\mathrm{diag}(1, 1, \ldots, 1, 0.5)W^{-T}. \tag{2.18}$$

Obviously, for each $i \geq 2$ we have

$$P_ie_j = \begin{cases} e_1 & \text{if } j = 1 \\ e_{i-1} + e_i & \text{if } j = i \\ e_j & \text{if } j \neq i, \end{cases}$$

from which, together with (2.16) and (2.18), we have $e_i^T\hat{D}^{-1}e_j > 0$ for all $i, j = 1, 2, \ldots, n$. The proof is complete. □

Throughout the remainder of the paper, we assume that

(H) $n \geq 3$ is a fixed integer and all elements of $\{a_i\}_{i=1}^n$ are non-negative with $\sum_{i=1}^n a_i > 0$.

Based on Lemma 2.1 and the Perron-Frobenius theorem [12], we can easily establish the following result with the same techniques employed in [7]. The details are omitted here.

Theorem 2.2. *Assume the hypotheses of* (H) *hold. If* λ_1 *is the smallest eigenvalue of the problem* (2.11)*, then*

(i) λ_1 *is simple and positive*;
(ii) *There exists a positive eigenvector* y *corresponding to* λ_1;
(iii) $\alpha_1 = 1/\lambda_1$ *is the simple, positive, and dominant eigenvalue of* $\hat{D}^{-1}\hat{A}$*. In addition, there is a positive eigenvector* y *of* $\hat{D}^{-1}\hat{A}$ *associated with* α_1.

Let λ_1 be the smallest eigenvalue of the problem (2.11) and y be an eigenvector associated with λ_1. It is seen from Lemma 2.1 that $\hat{D}$ is positive definite. Together with the fact that $\lambda_1 > 0$ in view of Theorem 2.2, we can write

$$\hat{D}^{-1}\hat{A}y = \frac{1}{\lambda_1}y. \tag{2.19}$$

That is, y is a positive eigenvector of $\hat{D}^{-1}\hat{A}$ corresponding to the eigenvalue $1/\lambda_1$. Due to the fact that $1/\lambda_1$ is simple and dominant in view of Theorem 2.2, the positive solution can be found by using Power method with any initial vector $y^{(0)}$ if its representation in terms of the eigenvectors of the matrix $\hat{D}^{-1}\hat{A}$ contains a nonzero contribution from the eigenvector associated with the dominant eigenvalue. Obviously, $y^{(0)} = e$, an n-dimensional vector of all ones, guarantees non-orthogonality with any positive vector. Therefore, the Power method with the initial vector e:

$$y^{(0)} = e,\ z^{(k)} = \hat{D}^{-1}\hat{A}y^{(k-1)},\ \lambda^{(k)} = ||z^{(k)}||_\infty,\ \ y^{(k)} = z^{(k)}/\lambda^{(k)} \tag{2.20}$$

is certainly convergent to a positive solution of the problem. Note that $z^{(k)} = \hat{D}^{-1}\hat{A}y^{(k-1)}$ is equivalent to

$$\hat{D}z^{(k)} = \hat{A}y^{(k-1)}. \tag{2.21}$$

The effectiveness of Crout Factorization Algorithm for tridiagonal linear systems (for example, see [2]) prompts us to explore the structure of the banded system of linear equations (2.21). Let us consider the LU decomposition of $\hat{D}$ where L is a lower triangular matrix and U is an upper triangular matrix with 1's along the diagonal. We note that $\hat{D}$ admits a unique LU factorization since all its leading principal minors are positive.

Performing the LU decomposition on $\hat{D}$, we find that all elements in L are zeros except for those in $\{(i,i) : i = 1, \ldots, n\}$, $\{(i, i-1), i = 2, \ldots, n\}$, and $\{(i, i-2), i = 3, \ldots, n\}$; while all elements in U are zeros except for those in $\{(i,i) : i = 1, 2, \ldots, n\}$, $\{(i, i+1), i = 1, 2, \ldots, n-1\}$, and $\{(i, i+2), i = 1, 2, \ldots, n-2\}$. That is, L and U can be found in the form of the following format:

$$L = \begin{pmatrix} l_{11} & 0 & 0 & \cdots & 0 & 0 & 0 \\ l_{21} & l_{22} & 0 & \cdots & 0 & 0 & 0 \\ l_{31} & l_{32} & l_{33} & \cdots & 0 & 0 & 0 \\ 0 & l_{42} & l_{43} & \cdots & 0 & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & l_{n-1,n-2} & l_{n-1,n-1} & 0 \\ 0 & 0 & 0 & \cdots & l_{n,n-2} & l_{n,n-1} & l_{nn} \end{pmatrix},$$

and

$$U = \begin{pmatrix} 1 & u_{12} & u_{13} & 0 & \cdots & 0 & 0 \\ 0 & 1 & u_{23} & u_{24} & \cdots & 0 & 0 \\ 0 & 0 & 1 & u_{34} & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & 0 & \cdots & u_{n-2,n-1} & u_{n-2,n} \\ 0 & 0 & 0 & 0 & \cdots & 1 & u_{n-1,n} \\ 0 & 0 & 0 & 0 & \cdots & 0 & 1 \end{pmatrix}.$$

We will focus on a procedure similar to the Crout factorization method for tridiagonal linear system, avoiding the computation of zeros in L and U. To get the nonzero entries of L and U, we explore $(\hat{d}_{ij}) = \hat{D} = LU$, resulting in the following relationships:

$$\hat{d}_{11} = l_{11},\ \hat{d}_{21} = l_{21},\ \hat{d}_{31} = l_{31},\ \hat{d}_{12} = l_{11}u_{12},\ \hat{d}_{13} = l_{11}u_{13}, \tag{2.22}$$

$$\hat{d}_{22} = l_{21}u_{12} + l_{22},\ \hat{d}_{32} = l_{31}u_{12} + l_{32},\ \hat{d}_{42} = l_{42}, \tag{2.23}$$

$$\hat{d}_{23} = l_{21}u_{13} + l_{22}u_{23},\ \hat{d}_{24} = l_{22}u_{24}, \tag{2.24}$$

$$\hat{d}_{i,i} = l_{i,i-2}u_{i-2,i} + l_{i,i-1}u_{i-1,i} + l_{ii} \quad \text{for} \quad 3 \le i \le n, \tag{2.25}$$

$$\hat{d}_{i+1,i} = l_{i+1,i-1}u_{i-1,i} + l_{i+1,i} \quad \text{for} \quad 3 \le i \le n-1, \tag{2.26}$$

$$\hat{d}_{i+2,i} = l_{i+2,i} \quad \text{for} \quad 3 \le i \le n-2, \tag{2.27}$$

$$\hat{d}_{i,i+1} = l_{i,i-1}u_{i-1,i+1} + l_{ii}u_{i,i+1} \quad \text{for} \quad 3 \le i \le n-1, \tag{2.28}$$

$$\hat{d}_{i,i+2} = l_{ii}u_{i,i+2} \quad \text{for} \quad 3 \le i \le n-2. \tag{2.29}$$

For the coefficient matrix $\hat{D}$ specified in (2.12), in view of (2.22), (2.23), and (2.27), we have $l_{i+2,i} = \hat{d}_{i+2,i} = 1$ for $i = 1, 2, \ldots, n-2$. The other non-zero elements in L and U can be obtained alternatively, first entry $l_{ii}, l_{i+1,i}$ in L and then entries $u_{i,i+1}, u_{i,i+2}$ in U for each i. It is easily seen from (2.22) that

$$l_{11} = \hat{d}_{11} = 5,\ l_{21} = -4,\ u_{12} = \hat{d}_{12}/l_{11} = -0.8,\quad u_{13} = \hat{d}_{13}/l_{11} = 0.2\ . \tag{2.30}$$

We will use the fact that $l_{i+2,i} = 1$ for $1 \le i \le (n-2)$ in (2.23)-(2.26) to further reduce the computational complexity of LU decomposition of $\hat{D}$.

Procedure 1: Crout-like Factorization of $\hat{D}$

Step 0 Input n.

Step 1 Compute $l_{11},\ l_{21},\ u_{12}$, and u_{13} by (2.30).
Step 2 Set $l_{22} = 6 - l_{21}u_{12},\ l_{32} = -4 - u_{12},\ u_{23} = (-4 - l_{21}u_{13})/l_{22}$, and $u_{24} = 1/l_{22}$.
Step 3 For $i = 3 : n-2$, compute

$$l_{ii} = 6 - u_{i-2,i} - l_{i,i-1}u_{i-1,i}, \quad l_{i+1,i} = -4 - u_{i-1,i},$$
$$u_{i,i+1} = (-4 - l_{i,i-1}u_{i-1,i+1})/l_{ii}, \quad u_{i,i+2} = 1/l_{ii}.$$

Step 4 Compute

$$l_{n-1,n-1} = 7 - u_{n-3,n-1} - l_{n-1,n-2}u_{n-2,n-1}, \quad l_{n,n-1} = -4 - u_{n-2,n-1},$$
$$u_{n-1,n} = (-4 - l_{n-1,n-2}u_{n-2,n})/l_{n-1,n-1}.$$

Step 5 Compute $l_{nn} = 3 - u_{n-2,n} - l_{n,n-1}u_{n-1,n}$.
Step 6 Return

$$l1 = (l_{11}, l_{22}, \ldots, l_{nn})^T, \qquad l2 = (l_{21}, l_{32}, \ldots, l_{n,n-1})^T,$$
$$u1 = (u_{12}, u_{23}, \ldots, l_{n-1,n})^T,\ u2 = (u_{13}, u_{24}, \ldots, l_{n-2,n})^T.$$

After factorizing $\hat{D}$ into specially structured L and U, we need to solve two triangular systems of linear equations $Lx = b$ and $Uz = x$ to get the solution of $\hat{D}z = b$.

Procedure 2: Crout-like Factorization Algorithm for $\hat{D}z = b$.

Step 0 Input b and four vectors $l1, l2, u1, u2$ obtained from Procedure 1.
Step 1 Set $x_1 = b_1/l_{11}$ and $x_2 = (b_2 - l_{21}x_1)/l_{22}$.
Step 2 For $i = 3, 4, \ldots, n$, set $x_i = (b_i - x_{i-2} - l_{i,i-1}x_{i-1})/l_{ii}$.
Step 3 Set $z_n = x_n$ and $z_{n-1} = x_{n-1} - u_{n-1,n}z_n$.
Step 4 For $i = n-2, n-3, \ldots, 1$, set $z_i = x_i - u_{i,i+1}z_{i+1} - u_{i,i+2}z_{i+2}$.
Step 5 Return z, the solution to $\hat{D}z = b$.

Procedure 1 is the LU factorization method but it avoids the computation of the zero elements in L and U. Let us count the total number of multiplications and divisions by ignoring the additions and subtractions. It is easy to see that there are $4n - 8$ multiplications or divisions in Procedure 1 and $4n - 4$ multiplications or divisions in Procedure 2. Thus, both procedures have linear computational complexities.

Combining the Power method in (2.20) with the Crout-like Procedures described above, we have the following algorithm for a positive solution to the problem (1.3).

Algorithm 1. The Power method for Problem (2.9)

Step 0 Input $n,\ \hat{A} = \mathrm{diag}(a_1, a_2, \ldots, a_n/2)$. Set $\epsilon = 10^{-8}$ and $y = e$, a vector of all ones.
Step 1 Call Procedure 1.
Step 2 Repeat
Step 2.1 Form $b = \hat{A}y$ and solve $\hat{D}z = b$ with Procedure 2.

Step 2.2 Set $\beta = ||z||_\infty$ and $y^+ = z/\beta$.
Step 2.3 If $||y^+ - y||_2 < \epsilon$, then stop and goto Step 3
else set $y = y^+$ and goto Step 2.1.
Step 3 Return the eigenvalue $\lambda = 1/\beta$ and a positive solution y^+.

Algorithm 1 is based on the Power method for the eigenvector associated with the dominant eigenvalue and the Crout-like factorization algorithm for the sparse system of linear equations involved. It requires one structured LU decomposition in Step 1 with a cost of $4n - 8$ multiplications or divisions. Once the non-zero elements of L and U are obtained, they can be repeatedly used in the major loop of the algorithm, i.e., the Step 2. We note that n multiplications are needed in forming b and $4n - 4$ multiplications or divisions in a call to Procedure 2 at each iteration of the loop. Though there may be a few iterations in Step 2, Algorithm 1 is very fast due to the fact that each repetition of Step 2 only requires $O(n)$ multiplications and divisions. Moreover, this algorithm is extremely suitable for large scale problems since only a few vector variables are needed in its implementation.

The algorithm was implemented in Matlab 6.1 and executed in Lenovo X201 Tablet PC which is equipped with Intel Core $i7$ CPU at 2.00 GHz, with dual-core, 4 logical processors, and 4 GB physical memory (RAM). A few problems were tested and all were solved successfully. The speed of convergence depends on the dimension n and the other data of the problems.

Consider the boundary value problem of the linear beam equation:

$$\begin{cases} u'''' = \frac{12\lambda}{(5-t^2)^2} u(t), \quad t \in [0,1] \\ u(0) = u''(0) = u'(1) = u'''(1) = 0. \end{cases} \tag{2.31}$$

It is easy to check that one eigenvalue of the problem is $\lambda^t = 10$ and a corresponding positive eigenfunction is

$$u(t) = t(t^2 - 5)^2, \quad 0 \le t \le 1. \tag{2.32}$$

For this example, $a_i = 12h^4/(5 - t_i^2)^2 = 12/(5n^2 - i^2)^2$. Let $\hat{\lambda}$ and $\hat{y}$ be the eigen-pair obtained by applying Algorithm 1 to this test problem and $u = (u(t_1), u(t_2), \ldots, u(t_n))^T$ be the vector of true solution obtained by evaluating $u(t)$ of (2.32) at nodes. Since the infinity norm of the computed solution $\hat{y}$ is 1, we also need to scale the vector u accordingly. Let $y^t = u/||u||_\infty$ and define

$$\text{error}_1 = \max\left\{ \left|(\hat{y}_i - y_i^t)/y_i^t\right| \ : \ i = 1, 2 \ldots, n \right\},$$
$$\text{error}_2 = |(\hat{\lambda} - \lambda^t)/\lambda^t|.$$

We note that error_1 represents the maximum relative error of all components of the computed positive solution and error_2 represents the relative error of the computed eigenvalue. Algorithm 1 is called for a positive solution of the test problem (2.31) through the discrete model (2.9). In the following table, we will report error_1 and error_2 along with the number of calls to Procedure 2 and the time in seconds of each run for each choice of h.

h (or $1/n$)	10^{-1}	10^{-2}	10^{-3}	5×10^{-4}	2.5×10^{-4}
error_1	2.57×10^{-4}	2.53×10^{-6}	3.99×10^{-8}	5.07×10^{-7}	1.17×10^{-6}
error_2	0.0062	6.18×10^{-5}	1.29×10^{-6}	6.32×10^{-6}	3.38×10^{-4}
time	0.002030	0.026830	0.063650	0.12940	0.277370
# of calls	6	6	6	6	6

The method proposed solves the problem successfully and very quickly for each choice of n. With $h = 0.001$, an approximate positive solution is found within a relative error less than 4×10^{-8} and corresponding eigenvalue is found within a relative error of 1.3×10^{-6}. We observed that the rounding errors are not significant for a moderate stepsize h. However, for a very small h, say 2.5×10^{-4}, the rounding errors begin to ruin the solution. Since the discrete model proposed in this section is obtained by ignoring the terms involving $O(h^2)$, the method has an error of $O(h^2)$ theoretically. In the following two tables, the rows labeled Ratio give the factors by which the errors decreased when n was doubled (or h was halved). Empirically, the factors approach 4.0 as h approaches 0, and that is what we observe in the tables below.

n (or $1/h$)	8	16	32	64	128	256
error_1	4.04×10^{-4}	9.94×10^{-5}	2.47×10^{-5}	6.18×10^{-6}	1.54×10^{-6}	3.86×10^{-7}
Ratio		4.06	4.02	4.00	4.01	3.99

n (or $1/h$)	8	16	32	64	128	256
error_2	0.0096	0.0024	6.03×10^{-4}	1.51×10^{-4}	3.77×10^{-5}	9.42×10^{-6}
Ratio		4	3.98	3.99	4.01	4.00

Finally, we comment that the method proposed in this paper starts from the vector of all ones as an initial estimate of the positive solution. The method was tested with a randomly generated vector of components in $[0, 1]$ as a starting point. We observed that it is insensitive to the choice of the starting point.

Acknowledgment. The research of Jun Ji was partly supported by the Kennesaw State University Tenured Faculty Professional Development Full Paid Leave Program in Fall 2015.

References

[1] R. P. Agarwal, D. O'Regan, and P. J. Y. Wong, *Positive Solutions of Differential, Difference, and Integral Equations*, Kluwer Academic, Dordrecht, 1998.

[2] R. L. Burden and J. D. Faires, *Numerical Analysis*, Thomson Brooks/Cole, Belmont, CA, 2005.

[3] J. M. Davis, L. H. Erbe, and J. Henderson, *Multiplicity of positive solutions for higher order Sturm-Liouville problems*, Rocky Mountain J. Math. **31** (2001), no. 1, 169–184.

[4] J. R. Graef and J. Henderson, *Double solutions of boundary value problems for* $2m^{\text{th}}$*-order differential equations and difference equations*, Comput. Math. Appl. **45** (2003), no. 6-9, 873–885.

[5] J. R. Graef and B. Yang, *Existence and nonexistence of positive solutions of fourth order nonlinear boundary value problems*, Appl. Anal. **74** (2000), no. 1-2, 201–214.

[6] C. P. Gupta, *Existence and uniqueness theorems for the bending of an elastic beam equation*, Appl. Anal. **26** (1988), no. 4, 289–304.
[7] J. Ji and B. Yang, *Eigenvalue comparisons for boundary value problems of the discrete beam equation*, Adv. Difference Equ. **2006** (2006), Article ID 81025, 1–9.
[8] ———, *Positive solutions for boundary value problems of second order difference equations and their computation*, J. Math. Anal. Appl. **367** (2010), no. 2, 409–415.
[9] ———, *Eigenvalue Comparisons for a class of boundary value problems of discrete beam equation*, Appl. Math. Comput. **218** (2012), no. 9, 5402–5408.
[10] P. Pietramala, *A note on a beam equation with nonlinear boundary conditions*, Bound. Value Probl. **2011** (2011), Article ID 376782, 14 pages.
[11] R. A. Usmani, *Discrete variable methods for a boundary value problem with engineering applications*, Math. Compu. **32** (1978), no. 144, 1087–1096.
[12] R. S. Varga, *Matrix Iterative Analysis*, Prentice-Hall, Inc., Englewood Cliffs, N.J., 1962.
[13] Q. Yao, *An existence theorem for a nonlinear elastic beam equations with all order derivatives*, J. Math. Study **38** (2005), no. 1, 24–28.

Jun Ji
Department of Mathematics
Kennesaw State University
Kennesaw, GA 30144, USA
E-mail address: `jji@kennesaw.edu`

Bo Yang
Department of Mathematics
Kennesaw State University
Kennesaw, GA 30144, USA
E-mail address: `byang@kennesaw.edu`