

Eigenvalue Computation of Regular 4th Order Sturm-Liouville Problems*

Ahmad Alalyani

Al Baha University, Al Baha, KSA

Email: aalalyani@hotmail.com

How to cite this paper: Alalyani, A. (2019) Eigenvalue Computation of Regular 4th Order Sturm-Liouville Problems. *Applied Mathematics*, 10, 784-803.
<https://doi.org/10.4236/am.2019.109056>

Received: July 21, 2019

Accepted: September 23, 2019

Published: September 26, 2019

Copyright © 2019 by author(s) and Scientific Research Publishing Inc. This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

In this paper we present and test a numerical method for computing eigenvalues of 4th order Sturm-Liouville (SL) differential operators on finite intervals with regular boundary conditions. This method is a 4th order shooting method based on Magnus expansions (MG4) which use MG4 shooting as the integrator. This method is similar to the SLEUTH (Sturm-Liouville Eigenvalues Using Theta Matrices) method of Greenberg and Marletta which uses the 2nd order Pruess method (also known as the MG2 shooting method) for the integrator. This method often achieves near machine precision accuracies, and some comparisons of its performance against the well-known SLEUTH software package are presented.

Keywords

4th Order Sturm-Liouville Problem, Magnus Methods (MG4), Regular Boundary Conditions

1. Introduction

In this paper¹ we consider the self-adjoint differential operators which arise from the 4th order differential equation

$$L(y) = y^{(4)}(x) - (s(x)y'(x))' + q(x)y(x) = \lambda y(x), \quad a \leq x \leq b \quad (1)$$

when separated, self-adjoint boundary conditions are imposed at each of the two regular endpoints $x = a$ and $x = b$.

We make the assumptions:

- 1) $q(x)$ is continuous on $[a, b]$.
- 2) $s(x)$ and $s'(x)$ are continuous on $[a, b]$.

*2010 Mathematics Subject Classification 34L15; 34L16; 65L15; 70J50.

¹The content of this paper is related to my Ph.D. dissertation [1].

3) $-\infty < a < b < \infty$.

Under these assumptions both endpoints a and b are regular endpoints. The most general separated, self-adjoint boundary conditions which can be imposed at $x = a$ and $x = b$ are

$$\begin{pmatrix} l_1(y) \\ l_2(y) \end{pmatrix} = A_1 \begin{pmatrix} y(a) \\ y''(a) \end{pmatrix} + A_2 \begin{pmatrix} -y'''(a) + s(a)y'(a) \\ -y'(a) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad (2)$$

and

$$\begin{pmatrix} r_1(y) \\ r_2(y) \end{pmatrix} = B_1 \begin{pmatrix} y(b) \\ y''(b) \end{pmatrix} + B_2 \begin{pmatrix} -y'''(b) + s(b)y'(b) \\ -y'(b) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad (3)$$

where A_1, A_2 and B_1, B_2 are any choice of real, 2×2 matrices satisfying the properties

$$A_1 A_2^T - A_2 A_1^T = 0 \quad (4)$$

$$A_1 A_1^T + A_2 A_2^T = I_2. \quad (5)$$

and

$$B_1 B_2^T - B_2 B_1^T = 0 \quad (6)$$

$$B_1 B_1^T + B_2 B_2^T = I_2. \quad (7)$$

The above boundary conditions can be shown to be equivalent to the general forms of boundary conditions used by Everitt [2] (in his PhD dissertation on 4th order Sturm-Liouville problems), Fulton [3] and many others.

The domain of the maximal operator L_1 associated with the Equation (1) on the closed interval $[a, b]$ is

$$D(L_1) = \{f \in L_2(a, b) : f, f', f'', f''' \in AC_{loc}(a, b), Lf \in L_2(a, b)\}, \quad (8)$$

where AC_{loc} is the space of functions which are absolutely continuous on compact subsets of (a, b) . The self-adjoint operators associated with Equation (1) are then obtained by restricting $D(L_1)$ by two boundary conditions at the left endpoint and two boundary conditions at the right endpoint as in (2) and (3), namely

$$D(L_{A_1, A_2, B_1, B_2}) = \{f \in D(L_1) : l_1(f) = 0, l_2(f) = 0, r_1(f) = 0, r_2(f) = 0\} \quad (9)$$

The Green's formula for the 4th order equation is

$$\int_a^b (zLy - yLz) dx = [y, z]_{(x)} \Big|_a^b, \quad (10)$$

where the bilinear concomitant is defined as

$$[y, z]_{(x)} := \begin{vmatrix} y'''(x) & z'''(x) \\ y(x) & z(x) \end{vmatrix} - \begin{vmatrix} y''(x) & z''(x) \\ y'(x) & z'(x) \end{vmatrix} + s(x) \begin{vmatrix} y(x) & z(x) \\ y'(x) & z'(x) \end{vmatrix}. \quad (11)$$

Using this definition, and the boundary conditions (2) and (3), it can be shown that the operators L_{A_1, A_2, B_1, B_2} on $L_2(a, b)$ are symmetric; that is, for all $f, g \in D(L_1)$:

$$(f, Lg) - (Lg, f) = \int_a^b f \cdot (Lg) - g \cdot (Lf) dx = 0. \quad (12)$$

This paper is devoted to the 4th order shooting method based on Magnus Expansions (MG4) for computation of eigenvalues of 4th order SL problems of the type (1), (2), (3) having regular endpoints.

For 2nd order SL problems, on both regular and singular intervals, there are several well developed software packages for eigenvalue and eigenfunction computations: SLEIGN [4], SLEIGN 2 [5], SLEDGE [6] [7], SLO2FM [8] [9] [10], MATSLISE [11] [12] [13]. The best source of information where their capabilities and general performance is discussed, is the book, Numerical Solution of Sturm-Liouville Problems, of John Pryce [8].

For the 4th order SL equation, the only reliable software package for eigenvalue and eigenfunction computations is ACM Algorithm 775: SUBROUTINE SLEUTH, produced by L. Greenberg and M. Marletta in 1997 [14]. This package restricts attention to problems on finite intervals with regular endpoints; at this writing there is still no readily available software for singular endpoints. The SLEUTH code handles eigenvalues and eigenfunctions only for SL problems with regular endpoints, and it is capable of handling a wide variety of possible boundary conditions. The Greenberg-Marletta algorithm is based on an integration scheme using piecewise trigonometric hyperbolic splines (the Pruess method), also known as the MG2 shooting method. This was also the underlying integration scheme in the SLEDGE package. The SLEUTH code is based on using formulas of Greenberg [14] [15] [16] for the number of eigenvalues less than λ .

Since eigenvalue/eigenfunction calculations for the 4th order equation have been tackled by many other methods we give here a brief overview of some of the existing competitive methods. The most prominent approaches to date, and those which continue to receive much attention, are as follows:

1) Extended Sampling Method (ESM) which relies on the classical Whittaker-Shannon-Kotelnikov sampling theorem [17] (also used for 2nd order SL problems).

2) Fliess Series Method [18] [19] [20] which represents the solution of an IVP for the 4th order Equation (1) in terms of iterated integrals involving the coefficient functions, $s(x)$ and $q(x)$.

3) Chebyshev Method [21]-[26] which approximates solutions of the 4th order Equation (1) by Chebyshev polynomials.

4) Boubaker Polynomials Expansion Scheme (BPES) [27] which approximates solutions of the 4th order Equation (1) using Boubaker polynomials and utilizes a Differential Quadrature Method (DQM).

5) Spectral Parameter Power Series (SPPS) Method [28] [29] [30] which expands the solutions of the SL equation (both second and 4th order) in a convergent Taylor expansion in the eigenparameter λ . This has recently proven to have much advantages both for theoretical problems and numerical computations for SL equations.

Selection of Test Problems

To investigate the performance of the method, we make the following selection of test problems. These problems are the square of a 2nd order SL problem.

1) The square of the 2nd order Bessel equation

$$Ly = y^{(4)} - (s(x)y')' + q(x)y = \lambda y, \quad x \in [1, 5] \quad (13)$$

where

$$s(x) = \frac{-2}{4x^2}, \quad (14)$$

and

$$q(x) = \frac{1}{16x^4} + \frac{3}{2x^4}. \quad (15)$$

2) The square of the 2nd order Modified Harmonic Oscillator equation

$$Ly = y^{(4)} - (s(x)y')' + q(x)y = \lambda y, \quad x \in [1, 5] \quad (16)$$

where

$$s(x) = 2(x^2 + x^4), \quad (17)$$

and

$$q(x) = (x^2 + x^4)^2 - (2 + 12x^2). \quad (18)$$

3) The square of the 2nd order equation

$$Ly = y^{(4)} - (s(x)y')' + q(x)y = \lambda y, \quad x \in [0, \pi] \quad (19)$$

where

$$s(x) = 2(\cos(x) + 2\cos(2x) + 3\cos(3x)), \quad (20)$$

and

$$q(x) = (\cos(x) + 2\cos(2x) + 3\cos(3x))^2 - (-\cos(x) - 8\cos(2x) - 27\cos(3x)). \quad (21)$$

4) The square of the 2nd order Coffey-Evan equation with $\beta = 10$

$$Ly = y^{(4)} - (s(x)y')' + q(x)y = \lambda y, \quad x \in \left[\frac{-\pi}{2}, \frac{\pi}{2} \right] \quad (22)$$

where

$$s(x) = 2(\beta^2 \sin^2(2x) - 2\beta \cos(2x)), \quad (23)$$

and

$$q(x) = (\beta^2 \sin^2(2x) - 2\beta \cos(2x))^2 - (8\beta^2 \cos(4x) + 8\beta \cos(2x)). \quad (24)$$

5) The square of the 2nd order Legendre equation

$$Ly = y^{(4)} - (s(x)y')' + q(x)y = \lambda y, \quad x \in \left[0, \frac{\pi}{4} \right] \quad (25)$$

where

$$s(x) = \frac{1}{2} \sec^2(x), \tag{26}$$

and

$$q(x) = \left(\frac{\sec^4(x)}{16} \right) - \left(\sec^2(x) \tan^2(x) + \frac{\sec^4(x)}{2} \right). \tag{27}$$

Problem 5, the Legendre squared equation, arises from changes of variables to the non-LNF form discussed in [31].

2. The MG4 Shooting Method Associated with the 4th Order Sturm-Liouville Equation

In this section we describe an implementation of the MG4 shooting technique for the 4th order SL Equation (1) on regular intervals with $s(x), q(x)$ continuous. The Equation (1) can be converted to the 1st order system (Atkinson [32], pp. 323-324) $Y'(x, \lambda) = A(x) \cdot Y(x, \lambda)$, where

$$Y'(x, \lambda) = \begin{pmatrix} 0 & 0 & 0 & -1 \\ 0 & 0 & -1 & -s(x) \\ q(x) - \lambda & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{pmatrix} Y(x, \lambda) = A(x) \cdot Y(x, \lambda), \tag{1}$$

and

$$Y(x, \lambda) = \begin{pmatrix} y(x, \lambda) \\ y''(x, \lambda) \\ -y'''(x, \lambda) + s(x)y'(x, \lambda) \\ -y'(x, \lambda) \end{pmatrix}. \tag{2}$$

Remark 2.1 Currently the most reliable software package for eigenvalues and eigenfunctions of the 4th order Sturm-Liouville equation with regular endpoints is ACM Algorithm 775: SUBROUTINE SLEUTH, produced by L. Greenberg and M. Marletta in 1997 [14], which is available from NETLIB at ORNL. The SLEUTH (Sturm-Liouville Eigenvalues Using Theta Matrices) code employs an MG2 approximation for the solution $Y(x) = e^{\sigma(x)} Y_0$, (see [33], p. 283), on each mesh interval of a Hamiltonian system similar to the system (1) (the order of the derivatives in the above $Y(x, \lambda)$ vector being slightly different). The SLEUTH code is based on using formulas of Greenberg [15] [16] [34] [35] for the number of eigenvalues less than λ , so the eigenvalue algorithm is quite different than the method we are proposing here for eigenvalue computation for (1) using (1).

For the IVP, $Y'(x, \lambda) = A(x) \cdot Y(x, \lambda)$, $Y(0) = I$, where A is a constant matrix is also basic to Magnus methods for (1). We introduce the following lemma and the definitions of the Lie-group and Lie-algebra (see [36], Prop. 7.2.3 and Def. 7.2.4).

Definition 2.1 $SL(4, \mathbb{R})$ is the Lie-group and defined as:

$$SL(4, \mathbb{R}) := \{A \mid A \text{ is } 4 \times 4 \text{ matrix with real entries and } \det(A) = 1\}.$$

Definition 2.2 $sl(4, \mathbb{R})$ is the Lie-algebra and defined as:

$$sl(4, \mathbb{R}) := \{A \mid A \text{ is } 4 \times 4 \text{ matrix with real entries and } \operatorname{tr}(A) = 0\}.$$

Lemma 2.1 If $X \in sl(4, \mathbb{R})$, then $\exp(X) \in SL(4, \mathbb{R})$, i.e.

$$\det(\exp(X)) = 1. \quad (3)$$

Remark 2.2 For any constant matrix $X \in sl(4, \mathbb{R})$, it follows from this lemma that the solution

$$Y(t, \lambda) = e^{Xt} \quad (4)$$

of the IVP,

$$Y'(t, \lambda) = X(t) \cdot Y(t, \lambda), \quad (5)$$

$$Y(0) = I,$$

lies in the Lie Group $SL(4, \mathbb{R})$.

The Magnus methods originate (see [37]) with the observation that an analytical solution of (1) with initial condition $Y(0) = Y_0$ can be written as, (see [33], p. 283),

$$Y(x) = e^{\sigma(x)} Y_0 \quad (6)$$

where

$$\begin{aligned} \sigma(x) = & \int_0^x A(k) dk + \frac{1}{2} \int_0^x \left[A(k), \int_0^k A(\xi) d\xi \right] dk \\ & + \frac{1}{4} \int_0^x \left[A(k), \int_0^k \left[A(\xi), \int_0^\xi A(\eta) d\eta \right] d\xi \right] dk \\ & + \frac{1}{12} \int_0^x \left[\left[A(k), \int_0^k A(\xi) d\xi \right], \int_0^k A(\eta) d\eta \right] dk + \dots \end{aligned} \quad (7)$$

and where the square brackets denote the matrix commutator and are defined as:

$$[A, B] := A \cdot B - B \cdot A \quad (8)$$

The MG4 method is a well known 4th order method obtained by truncation of the above Magnus series, together with evaluation of the A matrix in (1) at two gaussian points A_1 and A_2 :

For the Hamiltonian system (1), we put

$$A_1 := A \left(x_n + h \left(\frac{\sqrt{3}-1}{2\sqrt{3}} \right) \right) = A(x'), \quad (9)$$

$$A_2 := A \left(x_n + h \left(\frac{\sqrt{3}+1}{2\sqrt{3}} \right) \right) = A(x''), \quad (10)$$

(meaning that $q(x)$ and $s(x)$ in (1) are to be evaluated at $x = x'$ and $x = x''$ in the n th mesh interval $[x_n, x_{n+1}]$). Then the MG4 method of Iserles and Norsett ([38], p. 1012), and ([33], p. 288) for (1) takes the form (for a fixed step size h)

$$Y_{n+1} = \exp(h\tilde{A}) \cdot Y_n, \quad (11)$$

where the transfer matrix for passing from x_n to x_{n+1} is $M := \exp(h\tilde{A})$ with

$$\begin{aligned} \tilde{A} &:= \frac{1}{2}(A_1 + A_2) + \frac{1}{4\sqrt{3}}[A_2, A_1]h \\ &= \frac{1}{2}(A(x') + A(x'')) + \frac{1}{4\sqrt{3}}h(A(x'')A(x') - A(x')A(x'')) \quad (12) \\ &= \begin{pmatrix} 0 & 0 & 0 & -1 \\ \frac{-\sqrt{3}h(q_1 - q_2)}{12} & \frac{-\sqrt{3}h(s_1 - s_2)}{12} & -1 & \frac{-s_1 - s_2}{2} \\ \frac{q_1 + q_2}{2} - \lambda & 0 & 0 & \frac{\sqrt{3}h(q_1 - q_2)}{12} \\ 0 & -1 & 0 & \frac{\sqrt{3}h(s_1 - s_2)}{12} \end{pmatrix}, \end{aligned}$$

where the square bracket $[A_2, A_1]$ denotes the matrix commutator and is defined as:

$$[A_2, A_1] = A_2A_1 - A_1A_2, \quad (13)$$

and

$$q_1 = q(x'), q_2 = q(x''), s_1 = s(x'), s_2 = s(x'').$$

The four eigenvalues of \tilde{A} are

$$\lambda_1 = -\left[A - \frac{\sqrt{B}}{2}\right]^{\frac{1}{2}}, \quad (14)$$

$$\lambda_2 = -\left[A + \frac{\sqrt{B}}{2}\right]^{\frac{1}{2}}, \quad (15)$$

$$\lambda_3 = -\lambda_1, \quad (16)$$

$$\lambda_4 = -\lambda_2, \quad (17)$$

where

$$A = \frac{s_1 + s_2}{4} + \frac{h^2(s_1^2 + s_2^2)}{96} - \frac{h^2s_1s_2}{48}, \quad (18)$$

and

$$\begin{aligned} B = 4\lambda + \frac{h^4(s_1^4 + s_2^4)}{2304} - \frac{h^4(s_1^3s_2 + s_1s_2^3)}{576} + \frac{h^4s_1^2s_2^2}{384} + \frac{h^2(s_1^3 + s_2^3 - s_1^2s_2 - s_1s_2^2)}{48} \\ + \frac{s_1^2 + s_2^2}{4} + \frac{s_1s_2}{2} - 2(q_1 + q_2). \end{aligned} \quad (19)$$

Eigenvalues of \tilde{A} matrix:

Let us define

$$D := A - \frac{\sqrt{B}}{2}, \quad (20)$$

and

$$E := A + \frac{\sqrt{B}}{2}. \quad (21)$$

Then the following four cases of eigenvalues of \tilde{A} arise, involving both

complex and real eigenvalues:

Case 1: $B > 0$ and $(D > 0, E > 0)$.

$$\lambda_1 = -|D|^{\frac{1}{2}}, \quad (22)$$

$$\lambda_2 = -|E|^{\frac{1}{2}}, \quad (23)$$

$$\lambda_3 = |D|^{\frac{1}{2}}, \quad (24)$$

$$\lambda_4 = |E|^{\frac{1}{2}}. \quad (25)$$

Case 2: $B > 0$ and $(D < 0, E < 0)$.

$$\lambda_1 = -i|D|^{\frac{1}{2}}, \quad (26)$$

$$\lambda_2 = -i|E|^{\frac{1}{2}}, \quad (27)$$

$$\lambda_3 = i|D|^{\frac{1}{2}}, \quad (28)$$

$$\lambda_4 = i|E|^{\frac{1}{2}}. \quad (29)$$

Case 3: $B > 0$ and $(D < 0, E > 0, \text{ where } D < E)$.

$$\lambda_1 = -i|D|^{\frac{1}{2}}, \quad (30)$$

$$\lambda_2 = -|E|^{\frac{1}{2}}, \quad (31)$$

$$\lambda_3 = i|D|^{\frac{1}{2}}, \quad (32)$$

$$\lambda_4 = |E|^{\frac{1}{2}}. \quad (33)$$

Case 4: $B < 0$.

$$\lambda_1 = -\left[A - \frac{i\sqrt{|B|}}{2}\right]^{\frac{1}{2}}, \quad (34)$$

$$\lambda_2 = -\left[A + \frac{i\sqrt{|B|}}{2}\right]^{\frac{1}{2}}, \quad (35)$$

$$\lambda_3 = -\lambda_1, \quad (36)$$

$$\lambda_4 = -\lambda_2. \quad (37)$$

Remark 2.3 It follows from (12) that

$$\text{trace}((x - x_n)\tilde{A}) = 0, \quad (38)$$

so that $(x - x_n)\tilde{A} \in sl(4, \mathbb{R})$. Also we observe that on diagonalization we have (in all the above 4 cases),

$$\begin{aligned} \det(\exp[(x-x_n)\tilde{A}]) &= \det(P) \cdot \det(\exp[(x-x_n)D]) \cdot \det(P^{-1}) \\ &= \exp\left((x-x_n)\sum_{j=1}^4 \lambda_j\right) = 1, \end{aligned} \tag{39}$$

where $\{\lambda_j\}, j = 1, 2, 3, 4$, are the eigenvalues of \tilde{A} . Hence on each mesh interval,

$$Y(x, \lambda) = \exp[(x-x_n)\tilde{A}] \cdot Y_n(x, \lambda) \tag{40}$$

is a solution of

$$Y'(x, \lambda) = \tilde{A}(x) \cdot Y(x, \lambda) \tag{41}$$

which remains in the Lie Group, $SL(4, \mathbb{R})$.

We consider the SL problem for Equation (1) with the following choices of Dirichlet boundary conditions at the left and right endpoints (compare (2) and (3)).

$$L(y) = y^{(4)}(x) - (s(x)y'(x))' + q(x)y(x) = \lambda y(x), \quad a \leq x \leq b, \tag{42}$$

$$A_1 \begin{pmatrix} y(a) \\ y''(a) \end{pmatrix} + A_2 \begin{pmatrix} -y'''(a) + s(a)y'(a) \\ -y'(a) \end{pmatrix} = \begin{pmatrix} y(a) \\ y''(a) \end{pmatrix} = \begin{pmatrix} l_1(y) \\ l_2(y) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \tag{43}$$

$$B_1 \begin{pmatrix} y(b) \\ y''(b) \end{pmatrix} + B_2 \begin{pmatrix} -y'''(b) + s(b)y'(b) \\ -y'(b) \end{pmatrix} = \begin{pmatrix} y(b) \\ y''(b) \end{pmatrix} = \begin{pmatrix} r_1(y) \\ r_2(y) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}. \tag{44}$$

where

$$A_1 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \tag{45}$$

$$A_2 = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \tag{46}$$

and

$$B_1 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \tag{47}$$

$$B_2 = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}. \tag{48}$$

The left boundary conditions are implemented by fixing initial conditions for two solutions $Y_1(x, \lambda)$ and $Y_2(x, \lambda)$ of (1) at $x = a$, namely we define solutions $Y_1(x, \lambda)$ and $Y_2(x, \lambda)$ at $x = a$ by requiring

$$\begin{aligned} (Y_1(a, \lambda), Y_2(a, \lambda)) &= \begin{pmatrix} y_1(a, \lambda) & y_2(a, \lambda) \\ y_1''(a, \lambda) & y_2''(a, \lambda) \\ -y_1'''(a, \lambda) + s(a)y_1'(a, \lambda) & -y_2'''(a, \lambda) + s(a)y_2'(a, \lambda) \\ -y_1'(a, \lambda) & -y_2'(a, \lambda) \end{pmatrix} \\ &= \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 0 \\ I_2 \end{pmatrix}. \end{aligned} \tag{49}$$

Then the corresponding solutions $\{y_1(x, \lambda), y_2(x, \lambda)\}$ of (1) automatically satisfy the boundary conditions (43) at $x = a$. Using the solutions $\{Y_1(x, \lambda), Y_2(x, \lambda)\}$ of (1) defined by (49) we define the 2×2 matrices

$$U(x, \lambda) := \begin{pmatrix} y_1(x, \lambda) & y_2(x, \lambda) \\ y_1''(x, \lambda) & y_2''(x, \lambda) \end{pmatrix}, \quad (50)$$

and

$$V(x, \lambda) := \begin{pmatrix} -y_1'''(x, \lambda) + s(x)y_1'(x, \lambda) & -y_2'''(x, \lambda) + s(x)y_2'(x, \lambda) \\ -y_1'(x, \lambda) & -y_2'(x, \lambda) \end{pmatrix}. \quad (51)$$

The two-dimensional subspace,

$$\mathfrak{S} = \{y(x, \lambda) : y(x, \lambda) = Ay_1(x, \lambda) + By_2(x, \lambda) \text{ for some choice of } A \text{ and } B\}, \quad (52)$$

where $y_1(x, \lambda)$ and $y_2(x, \lambda)$ are solutions of (1) corresponding to the $Y_1(x, \lambda)$ and $Y_2(x, \lambda)$ solutions of the IVP, (1) and (49), of the four-dimensional solution space of (1) satisfies the boundary conditions (43) at $x = a$.

We have the following theorem giving necessary sufficient conditions (N.S.Cs) for the eigenvalues of the SL problem for Equation (1) which has boundary conditions at $x = a$ (43) and boundary conditions at $x = b$ (44).

Theorem 2.1 1) A N.S.C. for $\lambda \in (-\infty, \infty)$ to be an eigenvalue of the SL problem for Equation (1) with boundary conditions at $x = a$ (43) and boundary conditions at $x = b$ (44), and having multiplicity one, is:

$$\begin{vmatrix} r_1(y_1) & r_1(y_2) \\ r_2(y_1) & r_2(y_2) \end{vmatrix} = \begin{vmatrix} y_1(b, \lambda) & y_2(b, \lambda) \\ y_1''(b, \lambda) & y_2''(b, \lambda) \end{vmatrix} = 0 \quad (53)$$

and

$$\text{rank} \begin{pmatrix} r_1(y_1) & r_1(y_2) \\ r_2(y_1) & r_2(y_2) \end{pmatrix} = \text{rank} \begin{pmatrix} y_1(b, \lambda) & y_2(b, \lambda) \\ y_1''(b, \lambda) & y_2''(b, \lambda) \end{pmatrix} = 1. \quad (54)$$

2) A N.S.C. for $\lambda \in (-\infty, \infty)$ to be an eigenvalue of the SL problem for Equation (1) with boundary conditions at $x = a$ (43) and boundary conditions at $x = b$ (44), and having multiplicity one, is:

$$\begin{vmatrix} r_1(y_1) & r_1(y_2) \\ r_2(y_1) & r_2(y_2) \end{vmatrix} = \begin{vmatrix} y_1(b, \lambda) & y_2(b, \lambda) \\ y_1''(b, \lambda) & y_2''(b, \lambda) \end{vmatrix} = 0 \quad (55)$$

and

$$\text{rank} \begin{pmatrix} r_1(y_1) & r_1(y_2) \\ r_2(y_1) & r_2(y_2) \end{pmatrix} = \text{rank} \begin{pmatrix} y_1(b, \lambda) & y_2(b, \lambda) \\ y_1''(b, \lambda) & y_2''(b, \lambda) \end{pmatrix} = 0. \quad (56)$$

Proof Let $\{y_1(x, \lambda), y_2(x, \lambda)\}$ be the unique solutions of the 4th order Equation (1) which are defined by the initial conditions at $x = a$ and $\{Y_1(x, \lambda), Y_2(x, \lambda)\}$ be the corresponding solutions of the Hamiltonian system (1):

$$\begin{pmatrix} y_1(a, \lambda) & y_2(a, \lambda) \\ y_1''(a, \lambda) & y_2''(a, \lambda) \\ -y_1'''(a, \lambda) + s(a)y_1'(a, \lambda) & -y_2'''(a, \lambda) + s(a)y_2'(a, \lambda) \\ -y_1'(a, \lambda) & -y_2'(a, \lambda) \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix} \quad (57)$$

By fixing (57) the two-dimensional space \mathfrak{S} is fixed by the 4×2 matrix (57). The constants in the (57) matrix were chosen to ensure that the boundary conditions at $x = a$ (43) was satisfied, so we know that

$$\begin{pmatrix} l_1(y_1) \\ l_2(y_1) \end{pmatrix} = \begin{pmatrix} y_1(a, \lambda) \\ y_1''(a, \lambda) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad (58)$$

$$\begin{pmatrix} l_1(y_2) \\ l_2(y_2) \end{pmatrix} = \begin{pmatrix} y_2(a, \lambda) \\ y_2''(a, \lambda) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad (59)$$

that is, that both $y_1(x, \lambda)$ and $y_2(x, \lambda)$ satisfy the boundary conditions at $x = a$ (43). Also, of course, the space \mathfrak{S} of solutions spanned by these two solutions $\{y_1(x, \lambda), y_2(x, \lambda)\}$ of (1) satisfies the boundary conditions at $x = a$ (43), that is

$$l_1(Ay_1(x, \lambda) + By_2(x, \lambda)) = 0, \quad (60)$$

$$l_2(Ay_1(x, \lambda) + By_2(x, \lambda)) = 0 \quad (61)$$

for all $A, B \in \mathbb{C}$. It remains only to apply the boundary conditions at $x = b$ (44), *i.e.* to require that

$$r_1(y) = y(b, \lambda) = 0, \quad (62)$$

and

$$r_2(y) = y''(b, \lambda) = 0 \quad (63)$$

for boundary conditions at $x = b$ (44). Hence, we find

$$r_1(Ay_1(x, \lambda) + By_2(x, \lambda)) = Ar_1(y_1(x, \lambda)) + Br_1(y_2(x, \lambda)) = 0 \quad (64)$$

and

$$r_2(Ay_1(x, \lambda) + By_2(x, \lambda)) = Ar_2(y_1(x, \lambda)) + Br_2(y_2(x, \lambda)) = 0 \quad (65)$$

as the requirement for solutions in \mathfrak{S} to also satisfy the boundary conditions at $x = b$ (44). But the Equations (64) and (65) can have a nonzero solution for A, B if and only if

$$\begin{vmatrix} r_1(y_1) & r_1(y_2) \\ r_2(y_1) & r_2(y_2) \end{vmatrix} = \begin{vmatrix} y_1(b, \lambda) & y_2(b, \lambda) \\ y_1''(b, \lambda) & y_2''(b, \lambda) \end{vmatrix} = 0 \quad (66)$$

Hence (66) is a N.S.C. condition for λ to be an eigenvalue of the SL problem for Equation (1). The multiplicity of the eigenvalue is defined as the number of linearly independent solutions of (1) which satisfy both boundary conditions at $x = a$ and both boundary conditions at $x = b$. Since the dimension of \mathfrak{S} is two, this can be at most two. For multiplicity one, we must have

$$\text{rank} \begin{pmatrix} r_1(y_1) & r_1(y_2) \\ r_2(y_1) & r_2(y_2) \end{pmatrix} = \text{rank} \begin{pmatrix} y_1(b, \lambda) & y_2(b, \lambda) \\ y_1''(b, \lambda) & y_2''(b, \lambda) \end{pmatrix} = 1, \quad (67)$$

and in this case the solution of the two Equations (64) and (65) will be

$$\begin{pmatrix} A \\ B \end{pmatrix} = \begin{pmatrix} \frac{-r_1(y_2)}{r_1(y_1)} \\ 1 \end{pmatrix}, \text{ if } r_1(y_1) \neq 0 \quad (68)$$

and

$$\begin{pmatrix} A \\ B \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \text{ if } r_1(y_1) = 0. \quad (69)$$

In this case the eigenfunction, which is unique up to a constant multiple, is

$$y(x, \lambda) = B \left(\frac{-r_1(y_2)}{r_1(y_1)} \cdot y_1(x, \lambda) + y_2(x, \lambda) \right), \text{ if } r_1(y_1) \neq 0 \quad (70)$$

or

$$y(x, \lambda) = Ay_1(x, \lambda), \text{ if } r_1(y_1) = 0. \quad (71)$$

For multiplicity two, we need to require

$$\text{rank} \begin{pmatrix} r_1(y_1) & r_1(y_2) \\ r_2(y_1) & r_2(y_2) \end{pmatrix} = \text{rank} \begin{pmatrix} y_1(b, \lambda) & y_2(b, \lambda) \\ y_1''(b, \lambda) & y_2''(b, \lambda) \end{pmatrix} = 0, \quad (72)$$

and in this case $y_1(x, \lambda)$ and $y_2(x, \lambda)$ would be linearly independent eigenfunctions of (1). ■

More generally, the general case of boundary conditions (3) would give

$$f(\lambda) := \det(B_1U(b, \lambda) + B_2V(b, \lambda)) = 0 \quad (73)$$

as a N.S.C. for λ to be an eigenvalue of (1) with boundary conditions (44) and (3). More generally, the boundary conditions (2) could be handled by changing the initial conditions (57) appropriately.

3. Description of the MG4 Algorithm

We obtained the 4×4 transfer matrix

$$M = \exp(h\tilde{A}) \quad (1)$$

by doing the following steps:

- 1) Calculate the eigenvalues and the eigenvectors of \tilde{A} .
- 2) Diagonalize

$$\tilde{A} = P \cdot D \cdot P^{-1}, \quad (2)$$

where P denotes the matrix of eigenvectors of \tilde{A} , and D denotes the diagonal matrix of the four eigenvalues of \tilde{A} (12).

- 3) Put

$$M = P(\exp(hD))P^{-1}. \quad (3)$$

This gives a 4×4 transfer matrix with 4 cases of eigenvalues of \tilde{A} . (The matrix elements could be reduced to expressions involving sinh, cosh, sin and cos functions). Our MG4 code implements the above transfer matrix $M = \exp(h\tilde{A})$ by doing the matrix multiplication (3) numerically. Here we describe the implementation of the MG4 method for computing the eigenvalues of the SL problem for Equation (1) with the choices of Dirichlet boundary conditions (43) and (44) at the left and right endpoints. To impose the boundary conditions (44) on \mathfrak{I} , we integrate the IVP, (1) and (49), from $x = a$ to $x = b$ using the MG4 method on the 4×2 matrix $Y(x, \lambda)$. Then when $Y(b, \lambda)$ has been computed, the boundary conditions (44),

$$y(b, \lambda) = Ay_1(b, \lambda) + By_2(b, \lambda) = 0 \tag{4}$$

$$y''(b, \lambda) = Ay_1''(b, \lambda) + By_2''(b, \lambda) = 0, \tag{5}$$

will be satisfied for some choices of real constants A and B , not both zero, if and only if

$$f(\lambda) := \det(U(b, \lambda)) = \begin{vmatrix} y_1(b, \lambda) & y_2(b, \lambda) \\ y_1''(b, \lambda) & y_2''(b, \lambda) \end{vmatrix} = 0. \tag{6}$$

The computation is performed using an initial uniform mesh, applying bisection method with initial upper and lower bounds for a given eigenvalue λ_n , and then doubling the number of mesh points by bisecting the mesh to generate a Richardson h^4 -extrapolation table over successively bisected meshes. Then the extrapolated eigenvalue is selected when the eigenvalue extrapolation error satisfies a tolerance test.

3.1. Description of h^4 -Richardson's Extrapolation

If $q(x), s(x) \in C^\infty[a, b]$, we can assume that if MG4 method is applied we will have for each choice of h :

$$\hat{\lambda}(h) \approx \lambda_{Exact} + \tau_1 h^4 + \tau_2 (h^4)^2 + \dots + \tau_m (h^4)^m \tag{7}$$

for some choices of real constants $\tau_1, \tau_2, \dots, \tau_m$. For each $m = 1, 2, 3, \dots$

$$\hat{\lambda}(h) - \lambda_{Exact} = \tau_1 h^4 + \tau_2 (h^4)^2 + \dots + \tau_m (h^4)^m + O((h^4)^{m+1}) = O(h^4). \tag{8}$$

Putting

$$x_1 = h_0^4 \tag{9}$$

$$x_2 = h_1^4 = \left(\frac{h_0}{2}\right)^4 \tag{10}$$

\vdots

$$x_i = h_{i-1}^4 = \left(\frac{h_0}{2^{i-1}}\right)^4, \tag{11}$$

in Neville's algorithm ([39], 2.1.2.5b, p. 42),

$$T_{ik}(x) = T_{i,k-1}(x) + \frac{T_{i,k-1}(x) - T_{i-1,k-1}(x)}{\frac{x - x_{i-k} - 1}{x - x_i}}, \quad 1 \leq k \leq i, \quad i = 0, 1, 2, \dots, n, \quad (12)$$

we find

$$T_{I-1,J-1}(0) = T_{I-1,J-2}(0) + \frac{T_{I-1,J-2}(0) - T_{I-2,J-2}(0)}{16^{J-1} - 1} \quad (13)$$

where we have taken

$$k = J - 1, \quad i = I - 1, \quad x = 0, \quad x_i = h_{I-1}^4, \quad x_{i-k} = h_{I-J}^4.$$

Applying Neville's algorithm generates the H^4 -Richardson's extrapolation table for the eigenvalue computation. Defining

$$\hat{\lambda}_{IJ} := T_{I-1,J-1}(0) \quad (14)$$

we have from (13) that

$$\hat{\lambda}_{I,J} = \hat{\lambda}_{I,J-1} + \frac{\hat{\lambda}_{I,J-1} - \hat{\lambda}_{I-1,J-1}}{16^{J-1} - 1}, \quad (15)$$

where

$$\begin{aligned} \hat{\lambda}_{11} &= \text{computed value for } h = h_0, \\ \hat{\lambda}_{21} &= \text{computed value for } h = \frac{h_0}{2}, \\ &\vdots \\ \hat{\lambda}_{n1} &= \text{computed value for } h = \frac{h_0}{2^n}. \end{aligned}$$

here the second term in (15) is the extrapolation error. The first column of the extrapolation table, that is, the eigenvalues $\hat{\lambda}_{11}, \hat{\lambda}_{21}, \dots$, are computable quantities. The columns two, three, four, five, \dots , are generated from column one using (15).

3.2. Computing Large Eigenvalues by Using the Invariant-Imbedding Variables

In a manner similar to Greenberg and Marletta in their SLEUTH code (see [14], Section 3.2, pp. 461-462), we applied the change to "invariant-imbedding" variables, $VU^A, UV^A, \det U$ and $\det V$, in our code in order to provide a good stable integration scheme. We generated the "invariant-imbedding" variables by using the 2×2 matrices $U(x, \lambda)$ and $V(x, \lambda)$ which we defined in (50) and (51) for eigenvalue computation of the SL problem.

4. Numerical Results and Discussion

In this section we give some numerical outputs for each of the 5 test problems in Section 1.1, and compare with the comparable SLEUTH outputs. The 5 test problems are squares of 2nd order SL equations. For such problems the choice of Dirichlet boundary conditions for the 2nd order problem, generates, by squaring,

a 4th order SL problem whose eigenvalues are the squares of the 2nd order SL problem, Greenberg and Marletta ([14] pp. 478-481). For example, we consider in section 5 the following 2nd order SL problems

$$-y'' + \left(\frac{-1}{4x^2}\right)y = \lambda y \quad (1)$$

$$y(1) = 0, y(5) = 0$$

$$-y'' + (x^2 + x^4)y = \lambda y \quad (2)$$

$$y(1) = 0, y(5) = 0$$

$$-y'' + (\cos(x) + 2\cos(2x) + 3\cos(3x))y = \lambda y \quad (3)$$

$$y(0) = 0, y(\pi) = 0$$

$$-y'' + (\beta^2 \sin^2(2x) - 2\beta \cos(2x))y = \lambda y \quad (4)$$

$$y\left(\frac{-\pi}{2}\right) = 0, y\left(\frac{\pi}{2}\right) = 0$$

$$-y'' + \frac{1}{4}\sec^2(x)y = \lambda y \quad (5)$$

$$y(0) = 0, y\left(\frac{\pi}{4}\right) = 0$$

Squaring the self-adjoint operator corresponding to (1)-(5) gives the 4th order self-adjoint operator corresponding to the 4th order problems in **Tables 1-5**, respectively. Accordingly, the eigenvalues of the problems in **Tables 1-5** are the squares of the eigenvalues of the 2nd order SL problems (1)-(5), respectively. **Tables 1-5** give outputs of MG4 and SLEUTH codes on the test problems 1, 2, 3, 4 and 5 respectively, with the choices of Dirichlet boundary conditions. In these tables, we list the SLEUTH and MG4 outputs to 17 digits. The number of these digits which are correct is always a key issue in assessing the performance of a numerical algorithm. Since the exact eigenvalues of these 4th order SL problems are the squares of the exact eigenvalues of the 2nd order SL problems (1)-(5), we computed the eigenvalues of the 2nd order SL problems (1)-(5) and computed their squares to provide a benchmark against which we can compare MG4 and SLEUTH algorithm outputs. The purpose for the following tables is to make comparisons at reasonably high accuracy; so we ran the MG4 and SLEUTH codes with the tolerance parameter $TOL = 10^{-12}$. Outputs of the SLEDGE code (Sturm-Liouville Estimates Determined by Global Error Control) of Pruess and Fulton [6] for the problems (1)-(5) were obtained for the eigenvalue indices listed in **Tables 1-5**, and then their squares were computed to generate the second columns of **Tables 1-5**. Since SLEDGE is known to compute eigenvalues quite reliably to the user-requested accuracy, we used the squares of the SLEDGE eigenvalues as the benchmark for MG4 and SLEUTH codes in **Tables 1-5**. The error characterization E_{sledge} in columns 4 and 6 of **Tables 1-5** was computed as

$$E_{sledge} = \begin{cases} \left| \lambda_n^{(SLEDGE)} - \lambda_n \right|, & \text{if } \lambda_n < 1 \\ \left| \frac{\lambda_n^{(SLEDGE)} - \lambda_n}{\lambda_n^{(SLEDGE)}} \right|, & \text{if } \lambda_n \geq 1, \end{cases} \quad (6)$$

where $\lambda_n^{(SLEDGE)}$ are the SLEDGE-squared eigenvalues and λ_n are eigenvalues obtained by the SLEUTH and MG4 codes. This represents the absolute or relative eigenvalue errors of each code relative to the benchmark values obtained from SLEDGE. The obtained absolute or relative eigenvalue errors of each code are used to measure the performance of each method. In **Table 1**, we see that the error characterization E_{sledge} for the eigenvalues λ_0 , λ_{20} and λ_{100} obtained by the SLEUTH are slightly larger than the error characterization E_{sledge} for the eigenvalues λ_0 , λ_{20} and λ_{100} achieved by the MG4 method.

Table 1. Eigenvalues of Problem 1.

Eval Index	SLEDGE	SLEUTH	E_{sledge} of SLEUTH	MG4	E_{sledge} of MG4
0	0.339260710091670	0.3392607100805489	1.105e-11	0.339260710092205	5.348e-13
20	73973.71134197622	73973.711342122158	1.973e-12	73973.711342028924	7.125e-13
100	39594796.88731044	39594796.886610150	1.769e-11	39594796.887332238	5.506e-13

Table 2. Eigenvalues of Problem 2.

Eval Index	SLEDGE	SLEUTH	E_{sledge} of SLEUTH	MG4	E_{sledge} of MG4
0	236.0251207053473	236.02512061495375	3.830e-10	236.02512070536613	7.984e-14
50	3155257.7441797131	3155257.7441750434	1.480e-12	3155257.7441805508	2.655e-13
100	41735725.88393135	41735725.884191565	6.235e-12	41735725.883945607	3.415e-13

Table 3. Eigenvalues of Problem 3.

Eval Index	SLEDGE	SLEUTH	E_{sledge} of SLEUTH	MG4	E_{sledge} of MG4
0	0.2786088184066070	0.2786088184207267	1.412e-11	0.27860881843425472	2.765e-11
50	6765204.5033669453	6765204.5033744955	1.116e-12	6765204.5033704005	5.107e-13
100	104060404.5008649	104060404.50095831	8.977e-13	104060404.50091156	4.485e-13

Table 4. Eigenvalues of Problem 4.

Eval Index	SLEDGE	SLEUTH	E_{sledge} of SLEUTH	MG4	E_{sledge} of MG4
2	4871.3813098296687	4871.3813098537794	4.950e-12	4871.3813098332166	7.283e-13
50	7028539.5468045129	7028539.5468126526	1.158e-12	7028539.5467997193	6.820e-13
100	105083729.4441578	105083729.44410391	5.128e-13	105083729.44420157	4.166e-13

Table 5. Eigenvalues of Problem 5.

Eval Index	SLEDGE	SLEUTH	E_{sledge} of SLEUTH	MG4	E_{sledge} of MG4
0	265.7655513700918	265.76555137422213	1.554e-11	265.76555137021074	4.474e-13
8	1680440.528481040	1680440.5284831792	1.273e-12	1680440.5284811275	5.210e-14
30	236431164.13291141	236431164.15818894	1.069e-10	236431164.13306919	6.673e-13
100	26639566561.99389	26639566568.073307	2.282e-10	26639566561.998882	2.661e-13

In **Table 2**, we see that the error characterization E_{sledge} for the eigenvalues λ_0 , λ_{50} and λ_{100} obtained by the SLEUTH are slightly larger than the error characterization E_{sledge} for the eigenvalues λ_0 , λ_{50} and λ_{100} achieved by the MG4 method. In **Table 3**, we see that the error characterization E_{sledge} for the eigenvalues λ_0 and λ_{100} achieved by the MG4 method are quite comparable to the error characterization E_{sledge} for the eigenvalues λ_0 and λ_{100} obtained by the SLEUTH. But, the error characterization E_{sledge} for the eigenvalue λ_{50} obtained by the SLEUTH is slightly larger than the error characterization E_{sledge} for the eigenvalue λ_{50} achieved by the MG4 method. In **Table 4**, we see that the error characterization E_{sledge} for the eigenvalue λ_{100} achieved by the MG4 method is quite comparable to the error characterization E_{sledge} for the eigenvalue λ_{100} obtained by the SLEUTH. But, the error characterization E_{sledge} for the eigenvalues λ_2 and λ_{50} obtained by the SLEUTH are slightly larger than the error characterization E_{sledge} for the eigenvalues λ_2 and λ_{50} achieved by the MG4 method. In **Table 5**, we see that the error characterization E_{sledge} for the eigenvalues λ_0 , λ_8 , λ_{30} and λ_{100} obtained by the SLEUTH are slightly larger than the error characterization E_{sledge} for the eigenvalues λ_0 , λ_8 , λ_{30} and λ_{100} achieved by the MG4 method.

Remark 5.1 The machine precision, obtained from the FORTRAN routine EPSILON, on the desktop computer (with Pentium 4 processors) used to obtain the following outputs of the SLEUTH and SLEDGE codes was $\text{MACHEPS} = 2.22\text{D-16}$.

5. Conclusion

In this paper we have presented the MG4 algorithm of Iserles *et al.* [33] [40], for eigenvalue computation of regular 4th order Sturm-Liouville problems. Applying the change to “Invariant-Imbedding” variables in a manner similar to Greenberg and Marletta in their SLEUTH code [14] provides good stabilization for the MG4 algorithm, and this resulted in its capability for achieving high accuracy, very often on the order of machine precision. The MG4 algorithm appears to be competitive to the SLEUTH code.

Acknowledgements

The author would like to thank Al Baha University for their financial and moral support. The author would also like to express his sincere gratitude to the following people:

- 1) Professor Charles Fulton and Dr. Steven Pruess for supplying a FORTRAN 90 version of the SLEDGE code.
- 2) Professor M. Marletta for supplying a FORTRAN 77 version of the SLEUTH code based on use of the BLAS underlying the LAPACK software, and for suggesting a modification to it which allowed us to run SLEUTH with high accuracy requests.

Conflicts of Interest

The author declares no conflicts of interest regarding the publication of this paper.

References

- [1] Alalyani, A. (2019) Numerical Methods for Eigenvalue Computation of Fourth-Order Self-Adjoint Ordinary Differential Operators. PhD Dissertation, Florida Institute of Technology, Melbourne.
- [2] Everitt, W.N. (1957) The Sturm-Liouville Problem for Fourth Order Differential Equations. *The Quarterly Journal of Mathematics*, **8**, 146-160. <https://doi.org/10.1093/qmath/8.1.146>
- [3] Fulton, C. (1989) The Bessel-Squared Equation in the Lim-2, Lim-3, Im-4 Cases. *The Quarterly Journal of Mathematics*, **40**, 423-456. <https://doi.org/10.1093/qmath/40.4.423>
- [4] Bailey, P., Gordon, M. and Shampine, L. (1978) Automatic Solution of the Sturm-Liouville Problem. *ACM Transactions on Mathematical Software*, **4**, 193-208. <https://doi.org/10.1145/355791.355792>
- [5] Bailey, P.B., Everitt, W.N. and Zettl, A. (2001) The SLEIGN2 Sturm-Liouville Code. *ACM Transactions on Mathematical Software*, **21**, 143-192. <https://doi.org/10.1145/383738.383739>
- [6] Pruess, S. and Fulton, C. (1993) Mathematical Software for Sturm-Liouville Problems. *ACM Transactions on Mathematical Software*, **19**, 360-376. <https://doi.org/10.1145/155743.155791>
- [7] Fulton, C. and Pruess, S. (1998) The Computation of Spectral Density Functions for Singular Sturm-Liouville Problems Involving Simple Continuous Spectra. *ACM Transactions on Mathematical Software*, **34**, 107-129. <https://doi.org/10.1145/285861.285867>
- [8] Pryce, J.D. (1993) Numerical Solution of Sturm-Liouville Problems. Oxford University Press, Oxford.
- [9] Pryce, J.D. and Marletta, M. (1992) Automatic Solution of Sturm-Liouville Problems Using the Pruess Method. *Journal of Computational and Applied Mathematics*, **39**, 57-78. [https://doi.org/10.1016/0377-0427\(92\)90222-J](https://doi.org/10.1016/0377-0427(92)90222-J)
- [10] Pryce, J. (1986) Error Control of Phase-Function Shooting Methods for Sturm-Liouville Problems. *IMA Journal of Numerical Analysis*, **6**, 103-123. <https://doi.org/10.1093/imanum/6.1.103>
- [11] Ledoux, V., Van Daele, M. and Vanden Berghe, G. (2004) CP Methods of Higher Order for Sturm-Liouville and Schrödinger Equations. *Computer Physics Communications*, **162**, 153-165. <https://doi.org/10.1016/j.cpc.2004.07.001>
- [12] Ledoux, V., Van Daele, M. and Vanden Berghe, G. (2005) MATSLISE: A Matlab Package for the Numerical Solution of Sturm-Liouville and Schrödinger Equations. *ACM Transactions on Mathematical Software*, **31**, 532-554. <https://doi.org/10.1145/1114268.1114273>
- [13] Ixaru, L. (2000) CP Methods for the Schrödinger Equation. *Journal of Computational and Applied Mathematics*, **125**, 347-357. [https://doi.org/10.1016/S0377-0427\(00\)00478-7](https://doi.org/10.1016/S0377-0427(00)00478-7)
- [14] Greenberg, L. and Marletta, M. (1997) Algorithm 775: The Code SLEUTH for Solving Fourth-Order Sturm-Liouville Problems. *ACM Transactions on Mathematical Software*, **23**, 453-493. <https://doi.org/10.1145/279232.279231>
- [15] Greenberg, L. and Marletta, M. (1995) Oscillation Theory and Numerical Solution of Fourth-Order Sturm-Liouville Problems. *IMA Journal of Numerical Analysis*, **15**, 319-356. <https://doi.org/10.1093/imanum/15.3.319>

- [16] Greenberg, L. (1991) An Oscillation Method for Fourth-Order, Self-Adjoint, Two-Point Boundary Value Problems with Nonlinear Eigenvalues. *SIAM Journal on Mathematical Analysis*, **22**, 1021-1042. <https://doi.org/10.1137/0522067>
- [17] Chanane, B. (2010) Accurate Solutions of Fourth Order Sturm-Liouville Problems. *Journal of Computational and Applied Mathematics*, **234**, 3064-3071. <https://doi.org/10.1016/j.cam.2010.04.023>
- [18] Chanane, B. (1998) Eigenvalues of Fourth Order Sturm-Liouville Problems Using Fliess Series. *Journal of Computational and Applied Mathematics*, **96**, 91-97. [https://doi.org/10.1016/S0377-0427\(98\)00086-7](https://doi.org/10.1016/S0377-0427(98)00086-7)
- [19] Chanane, B. (2002) Fliess Series Approach to the Computation of the Eigenvalues of Fourth-Order Sturm-Liouville Problems. *Applied Mathematics Letters*, **15**, 459-563. [https://doi.org/10.1016/S0893-9659\(01\)00159-8](https://doi.org/10.1016/S0893-9659(01)00159-8)
- [20] Chanane, B. (1998) Eigenvalues of Sturm-Liouville Problems Using Fliess Series. *Applicable Analysis*, **69**, 233-238. <https://doi.org/10.1080/00036819808840659>
- [21] Mihaila, B. and Mihaila, I. (2002) Numerical Approximations Using Chebyshev Polynomial Expansions: El-gendi's Method Revisited. *Journal of Physics A: Mathematical and General*, **35**, 731-746. <https://doi.org/10.1088/0305-4470/35/3/317>
- [22] El-Gamel, M. and Sameeh, M. (2012) An Efficient Technique for Finding the Eigenvalues of Fourth-Order Sturm-Liouville Problems. *Applied Mathematics*, **3**, 920-925. <https://doi.org/10.4236/am.2012.38137>
- [23] El-Gamel, M. and Abd El-hady, M. (2013) Two Very Accurate and Efficient Methods for Computing Eigenvalues of Sturm-Liouville Problems. *Applied Mathematical Modeling*, **37**, 5039-5046. <https://doi.org/10.1016/j.apm.2012.10.019>
- [24] Fox, L. (1962) Chebyshev Methods for Ordinary Differential Equations. *The Computer Journal*, **4**, 318-331. <https://doi.org/10.1093/comjnl/4.4.318>
- [25] Saleh Taher, A.H., Malek, A. and Momeni-Masuleh, S.H. (2013) Chebyshev Differential Matrices for Efficient Computation of the Eigenvalues of Fourth-Order Sturm-Liouville Problems. *Applied Mathematical Modeling*, **37**, 4634-4642. <https://doi.org/10.1016/j.apm.2012.09.062>
- [26] Ycel, U. (2015) Numerical Approximations of Sturm-Liouville Eigenvalues Using Chebyshev Polynomial Expansions Method. *Cogent Mathematics*, **2**, Article ID: 1045223. <https://doi.org/10.1080/23311835.2015.1045223>
- [27] Ycel, U. and Boubaker, K. (2012) Differential Quadrature Method (DQM) and Boubaker Polynomials Expansion Scheme (BPES) for Efficient Computation of the Eigenvalues of Fourth-Order Sturm-Liouville Problems. *Applied Mathematical Modelling*, **36**, 158-167. <https://doi.org/10.1016/j.apm.2011.05.030>
- [28] Khmelnytskaya, K., Kravchenko, V. and Baldenebro-Obeso, J. (2012) Spectral Parameter Power Series for Fourth-Order Sturm-Liouville Problems. *Applied Mathematics and Computation*, **219**, 3610-3524. <https://doi.org/10.1016/j.amc.2012.09.055>
- [29] Kravchenko, V. and Porter, R. (2010) Spectral Parameter Power Series for Sturm-Liouville Problems. *Mathematical Methods in the Applied Sciences*, **33**, 459-468. <https://doi.org/10.1002/mma.1205>
- [30] Kravchenko, V. and Porter, R. (2015) Eigenvalue Problems, Spectral Parameter Power Series, and Modern Applications. *Mathematical Methods in the Applied Sciences*, **38**, 1945-1969. <https://doi.org/10.1002/mma.3213>
- [31] Fulton, C. and Krall, A. (1983) Self-Adjoint 4th Order Boundary Value Problem in the Lim-4 Case. *Proceedings of Symposium on Ordinary Differential Equations and*

Operators, Dundee, Lecture Notes in Mathematics 1032, 240-256.

<https://doi.org/10.1007/BFb0076800>

- [32] Atkinson, F. (1964) *Discrete and Continuous Boundary Problems*. Academic Press, New York. <https://doi.org/10.1063/1.3051875>
- [33] Iserles, A., Marthinsen, A. and Nrsett, S.P. (1999) On the Implementation of the Method of Magnus Series for Linear Differential Equations. *BIT Numerical Mathematics*, **39**, 281-304. <https://doi.org/10.1023/A:1022393913721>
- [34] Greenberg, L. and Marletta, M. (1997) The Validity of Richardson Extrapolation in Coefficient Approximation Methods for Fourth Order and Higher Order Self-Adjoint ODE Eigenvalue Problems. Technical Report, Department of Mathematics and Computer Science, University of Leicester, Leicester.
- [35] Greenberg, L. (1991) A Prüfer Method for Calculating Eigenvalues of Self-Adjoint Systems of Ordinary Differential Equations: Parts 1 and 2. Technical Report TR91-24, University of Maryland at College Park, College Park.
- [36] Gallier, J. (2005) *Manifolds, Lie Groups, Lie Algebras, Riemannian Manifolds, with Applications to Computer Vision and Robotics [Lecture Notes]*. Department of Computer and Information Science, University of Pennsylvania, Philadelphia.
- [37] Magnus, W. (1954) On the Exponential Solution of Differential Equations for a Linear Operator. *Communications on Pure and Applied Mathematics*, **7**, 649-673. <https://doi.org/10.1002/cpa.3160070404>
- [38] Iserles, A. and Nrsett, S.P. (1999) On the Solution of Linear Differential Equations in Lie Groups. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, **357**, 983-1019. <https://doi.org/10.1098/rsta.1999.0362>
- [39] Stoer, J. and Bulirsch, R. (1993) *Introduction to Numerical Analysis*. Second Edition, Springer-Verlag, New York. <https://doi.org/10.1007/978-1-4757-2272-7>
- [40] Iserles, A., Munthe-Kaas, H.Z., Norsett, S.P. and Zanna, A. (2005) Lie-Group Methods. *Acta Numerica*, **9**, 1-148. <https://doi.org/10.1017/S0962492900002154>