Article

Solutions of Sturm-Liouville Problems

Upeksha Perera [1],[*],[†] and **Christine Böckmann [2]**

[1]   Department of Mathematics, University of Kelaniya, Kelaniya 11600, Sri Lanka
[2]   Institut für Mathematik, Universität Potsdam, 14476 Potsdam, Germany; bockmann@uni-potsdam.de
[*]   Correspondence: upeksha@kln.ac.lk or bodhiyabadug@uni-potsdam.de
[†]   Current address: Institut für Mathematik, Universität Potsdam, 14476 Potsdam, Germany.

check for updates

Abstract: This paper further improves the Lie group method with Magnus expansion proposed in a previous paper by the authors, to solve some types of direct singular Sturm–Liouville problems. Next, a concrete implementation to the inverse Sturm–Liouville problem algorithm proposed by Barcilon (1974) is provided. Furthermore, computational feasibility and applicability of this algorithm to solve inverse Sturm–Liouville problems of higher order (for $n = 2, 4$) are verified successfully. It is observed that the method is successful even in the presence of significant noise, provided that the assumptions of the algorithm are satisfied. In conclusion, this work provides a method that can be adapted successfully for solving a direct (regular/singular) or inverse Sturm–Liouville problem (SLP) of an arbitrary order with arbitrary boundary conditions.

Keywords: Sturm–Liouville problems of higher order; singular Sturm–Liouville problems; inverse Sturm–Liouville problems

MSC: 34L16; 65L09; 34B24

1. Introduction

The inverse Sturm–Liouville theory was originated in 1929 by Ambarzumian [1] and further developed in [2–6].

The $2m$th order, nonsingular, self-adjoint eigenvalue problem (EVP) or Sturm–Liouville problem (SLP) is given by:

$$(-1)^m (p_0(x)y^{(m)})^{(m)} + (-1)^{m-1}(p_1(x)y^{(m-1)})^{(m-1)}$$
$$+ \ldots + (p_{m-2}(x)y'')'' - (p_{m-1}(x)y')' + p_m(x)y = \lambda w(x)y, \qquad a < x < b \tag{1}$$

together with some boundary conditions at a and b, the functions $p_k, (0 \le k \le m)$, and $w(x)$ being continuous on the finite closed interval $[a, b]$, and p_0 having a continuous derivative. In the inverse SLP, the coefficient functions $p_k, (0 \le k \le m)$ need to be reconstructed, given suitable valid spectral data.

For a discussion of analytical methods and numerical methods for inverse SLP, see [7,8], respectively. Iterative methods [9,10], Rayleigh–Ritz method [11], finite difference approximation [12], Quasi-Newton method [13], shooting method [14], interval Newton's method [15], finite-difference method [16], boundary value methods [17–20], Numerov's method [21–23], least-squares functional [24], generalized Rundell–Sacks algorithm [25,26], spectral mappings [27], Lie-group estimation method [28], Broyden method [29,30], decent flow methods [31], modified Numerov's method [32], Newton-type method [33], Fourier–Legendre series [34], and Chebyshev polynomials [35] are of particular importance among the existing methods to solve inverse SLP.

Numerical algorithms to solve the inverse fourth-order Sturm–Liouville problem (FSLP) are proposed in [36–38].

In [39], authors implemented Barcilon [9]'s iterative algorithm to solve inverse SLPs successfully. The present paper extends the results for the case $m = 2$ by considering the general-order inverse SLP algorithm as described in [36].

2. Materials and Methods

Inverse SLP Algorithm for the General Order

Setting $p_0(x), w(x) \equiv 1$ in (1) and taking the domain $[0, 1]$, the EVP reads as:

$$(y^{(m)})^{(m)} - (p_1 y^{(m-1)})^{(m-1)} + \ldots + p_m(x)y = \lambda y, \qquad 0 < x < 1. \tag{2}$$

According to [40], $m + 1$ distinct spectra are required in order to determine $p_1(x), p_2(x), \ldots, p_m(x)$. In other words, $m + 1$ different EVPs with distinct boundary conditions (BC) are needed. In addition, the spectra need to be interlaced, i.e.,

$$\ldots < \lambda_{n-1}^{(m+1)} < \lambda_n^{(1)} < \lambda_n^{(2)} < \lambda_n^{(3)} < \ldots < \lambda_n^{(m+1)} < \lambda_{n+1}^{(1)} < \ldots \tag{3}$$

Assume that m boundary conditions are common to all the given $m + 1$ eigenvalue problems, say, conditions at the right end point $x = 1$.

1. Combine the $m + 1$ eigenvalue problems into an equivalent linear differential equation:

    1.1. Define a vector function $\boldsymbol{\phi}(x, \lambda)$:

    $$\begin{aligned} \boldsymbol{\phi}^T(x, \lambda) &= [\varpi\chi, \varpi'\chi', \ldots, \varpi^{(m-1)}\chi^{(m-1)}] \\ &= [\phi_{(1)}, \phi_{(2)}, \ldots, \phi_{(m)}] \end{aligned}$$

    where $\varpi(x, \lambda)$ and $\chi(x, \lambda)$ are two solutions of Equation (2).

    1.2. Differentiate each $\phi_{(k)}$, $2m$ times.

    1.3. Obtain $m(2m + 1)$ linear equations for $\varpi^{(i)}\chi^{(j)} + \varpi^{(j)}\chi^{(i)}$, where $i \leq j$, $j = 0, 1, \ldots, 2m - 1$.

    1.4. Express $\varpi^{(i)}\chi^{(j)} + \varpi^{(j)}\chi^{(i)}$, $i \leq j$, $j = 0, 1, \ldots, 2m - 1$, in terms of $\phi_{(1)}, \phi_{(2)}, \ldots, \phi_{(m)}$ and their first $2m$ derivatives.

    1.5. Differentiate $\phi_{(i)}^{(2m)}$, $i = 1, 2, \ldots, m$ again, to obtain m linear coupled differential equations of order $2m + 1$, which is in matrix form:

    $$\mathcal{M}\boldsymbol{\phi} = \lambda \mathcal{N} \boldsymbol{\phi} \tag{4}$$

2. Obtain the corresponding boundary conditions for the linear differential equation using the boundary conditions of the $m + 1$ eigenvalue problems:

    2.1. Assume that $\varpi(x, \lambda)$ and $\chi(x, \lambda)$ satisfy the m boundary conditions common to all the given $m + 1$ eigenvalue problems.

    2.2. Then from 1.3, find the $m(3m + 1)/2$ boundary conditions at $x = 1$, and the $m(m + 1)/2$ boundary conditions at $x = 0$ for $\boldsymbol{\phi}(x, \lambda)$.

3. Solve Equation (4) using above boundary conditions, to obtain the solutions $\boldsymbol{\phi}_n$.

4. Solve the adjoint system of equations to the above system, and denote the solutions by $\boldsymbol{\eta}_n$.

5. Find the bi-orthogonal set of functions $\{y_n\}_1^\infty$ to $\{\boldsymbol{\phi}_n\}_1^\infty$, using the relation $y_n(x) = \mathcal{N}^T \boldsymbol{\eta}_n(x)$.

Some remarks on Algorithm 1 follow:

Line (Data:) Infinite eigenvalue (EV) sequence is replaced with the first N EVs. The truncated EV sequences $\{\lambda_n^{(i)}\}_{n=1}^N$, $i = 1, 2, \ldots, m + 1$ are obtained, using the Magnus method [39] to solve Equation (2).

Line (Result:) Output **p**: $m \times n$ matrix of function values, n: the number of subdivisions in the x-axis and m: the number of potentials.

Line (1) We take initial guess $\mathbf{p}^{(0)}$ as $\mathbf{0}_{m \times n}$.

Line (3) `while` loop exit condition is replaced with a maximum number of iterations: $kMAX$ and in each iteration the condition $\sum_{n=1}^{(m+1)N} |\hat{\sigma}_n - \sigma_n^{(k-1)}| < tol$ (*tol*: a fixed tolerance) is checked to exit the loop.

Line (4) $\langle \cdot \rangle$ denotes the integration w.r.t. x from 0 to 1 and $\mathbf{e}^T = [1,0]$. $\mathbf{y}_n$ and $\boldsymbol{\phi}_n$ are kept fixed at $\mathbf{y}_0$ and $\boldsymbol{\phi}_0$, which are the solutions by setting $p_i = 0$, $\forall i$. Solutions $\mathbf{y}_0$ and $\boldsymbol{\phi}_0$ are calculated using Mathematica [41]. Matlab [42] built-in functions: `trapz()` and `griddedInterpolant()` with `pchip` (piecewise cubic Hermite interpolating polynomial) option are used to approximate integrals and to have p as a function, respectively. The latter is required since, p as a function is input to the Magnus method. Again, the eigenvalue sequence $\sigma^{(k)}$ is calculated using $\mathbf{p}^{(k)}$ and the iteration repeats.

You may refer Section 2.6 of [39] for the specific computational details of the Algorithm 1 for the case $m = 1$ and next section for the case $m = 2$.

Algorithm 1: Solving inverse Sturm–Liouville problem (SLP).

Data: $m + 1$ distinct spectra $\{\lambda_n^{(i)}\}_{n=1}^\infty$, $i = 1, 2, \ldots, m + 1$ corresponding to $m + 1$ different EVPs.

Result: the unknown potential functions $\mathbf{p}^T(x) = [p_m(x), p_{m-1}(x), \ldots, p_1(x)]$

1 Coalesce the $m + 1$ sequences of eigenvalues into a single sequence $\{\hat{\sigma}_n\}_1^\infty$ using

$$\hat{\sigma}_{(i-1)(m+1)+j} = \lambda_i^{(j)}, j = 1, 2, \ldots, m + 1, \qquad i = 1, 2, \ldots \tag{5}$$

Set: initial guess $\mathbf{p}^{(0)}$, $k = 1$;

2 Solve: EVPs using $\mathbf{p}^{(0)}$ and combine the solutions to $\{\sigma_n^{(0)}\}$;

3 **while** $\sum_n |\hat{\sigma}_n - \sigma_n^{(k-1)}| \neq 0$ **do**

4 ┃ Set

$$\hat{\mathbf{y}}_n(x) = \frac{\langle \mathbf{e}^T \boldsymbol{\phi}_n \rangle}{\langle \mathbf{y}_n^T \boldsymbol{\phi}_n \rangle} \mathbf{y}_n \tag{6}$$

┃ and

$$\mathbf{p}^{(k)}(x) = \mathbf{p}^{(k-1)}(x) + \sum_{n=1}^\infty (\hat{\sigma}_n - \sigma_n^{(k-1)}) \hat{\mathbf{y}}_n(x) \tag{7}$$

┃ Solve the EVPs using $\mathbf{p}^{(k)}$ and set the solutions to $\{\sigma_n^{(k)}\}$;

5 ┃ $k = k + 1$;

6 **end**

---

3. Inverse FSLP Algorithm

Setting $m = 2$ in Equation (2) we obtain FSLP:

$$\mathcal{L}u \equiv u^{(4)} - (pu')' + qu = \lambda u. \tag{8}$$

Three spectra $\{\lambda_n\}$, $\{\mu_n\}$ and $\{\nu_n\}$ required to reconstruct the unknown coefficients: p and q will be obtained from the following EVPs:

$$\mathcal{L}u_n = \lambda_n u_n, \qquad u_n(0) = u_n''(0) = u_n(1) = u_n''(1) = 0, \tag{9}$$

$$\mathcal{L}v_n = \mu_n v_n, \qquad v_n(0) = v_n'(0) = v_n(1) = v_n''(1) = 0, \tag{10}$$

$$\mathcal{L}w_n = \nu_n w_n, \qquad w_n'(0) = w_n''(0) = w_n(1) = w_n''(1) = 0. \tag{11}$$

Although, Equations (9) and (10) are self-adjoint, Equation (11) is not. Its adjoint equation is:

$$\mathcal{L}\omega_n = \nu_n \omega_n, \qquad \omega_n'(0) = \omega_n'''(0) - p(0)\omega_n'(0) = \omega_n(1) = \omega_n''(1) = 0. \tag{12}$$

The eigenvalues λ_n, μ_n and ν_n (for $p = q = 0$) are given by the solutions of the following equations:

$$\sin(s) = 0, \qquad s^4 = \lambda \tag{13}$$

$$\tan(r) - \tanh(r) = 0, \qquad r^4 = \mu \tag{14}$$

$$\tan(t) + \tanh(t) = 0, \qquad t^4 = \nu \tag{15}$$

Furthermore,

$$t \approx \left(n - \frac{1}{4}\right)\pi, \qquad s = n\pi, \qquad r \approx \left(n + \frac{1}{4}\right)\pi, \qquad n \in \mathbb{N}$$

so that the spectra $\{\lambda_n\}$, $\{\mu_n\}$ and $\{\nu_n\}$ are interlaced, i.e.,

$$0 < \nu_1 < \lambda_1 < \mu_1 < \nu_2 < \lambda_2 < \mu_2 < \ldots$$

Since the eigenvalues are interlaced, they can be coalesced into a single one by defining

$$\left\{\sigma_k, \begin{bmatrix} \phi_k \\ \psi_k \end{bmatrix}\right\}_1^\infty = \begin{cases} \left\{\nu_n, \begin{bmatrix} w_n\omega_n \\ w_n'\omega_n' \end{bmatrix}\right\}_1^\infty & \text{for } k = 3n - 2, \\ \left\{\lambda_n, \begin{bmatrix} u_n^2 \\ u_n'^2 \end{bmatrix}\right\}_1^\infty & \text{for } k = 3n - 1, \qquad n = 1, 2, \ldots . \\ \left\{\mu_n, \begin{bmatrix} v_n^2 \\ v_n'^2 \end{bmatrix}\right\}_1^\infty & \text{for } k = 3n, \end{cases} \tag{16}$$

Denote:

$$\mathbf{q} = \begin{bmatrix} q \\ p \end{bmatrix}, \qquad \boldsymbol{\Phi}_n = \begin{bmatrix} \phi_n \\ \psi_n \end{bmatrix}. \tag{17}$$

Let $\chi(x, \lambda)$ and $\varpi(x, \lambda)$ be two solutions of (8) satisfying the common boundary conditions of Equations (9)–(12), namely, ($u(1) = u''(1) = 0$, etc.)

$$\chi(1, \lambda) = \varpi(1, \lambda) = \chi''(1, \lambda) = \varpi''(1, \lambda) = 0. \tag{18}$$

Guided by (16), define

$$\begin{cases} \phi(x, \lambda) = \varpi(x, \lambda)\chi(x, \lambda) \\ \psi(x, \lambda) = \varpi'(x, \lambda)\chi'(x, \lambda). \end{cases} \tag{19}$$

By differentiating ϕ and ψ five times each, and eliminating ω, and χ, 9th order linear differential equation for ϕ (or a coupled system of linear differential equations of order 5 for ϕ and ψ) can be obtained. When $p = q = 0$, this reduces to

$$\phi^{(9)}(x) - 12\lambda\phi^{(5)}(x) = 64\lambda^2\phi^{(1)}(x) \tag{20}$$

or

$$\begin{pmatrix} 3D^5 & -10D^3 \\ 0 & D^5 \end{pmatrix}\begin{pmatrix} \phi \\ \psi \end{pmatrix} = \lambda \begin{pmatrix} 8D & 0 \\ 10D^3 & -24D \end{pmatrix}\begin{pmatrix} \phi \\ \psi \end{pmatrix} \tag{21}$$

in the form of the Equation (4), and the boundary conditions read as:

$$\phi = \phi''' = \psi' = 0 \quad \text{at} \quad x = 0, \tag{22}$$
$$\phi = \phi' = \phi'' - 2\psi = \phi''' = \phi'''' - 4\psi'' = \psi' = \psi''' = 0 \quad \text{at} \quad x = 1. \tag{23}$$

The solution of the system of Equations (21)–(23) using Mathematica is:

$$\phi(x) = \frac{\csc(\mu)}{8(2\cosh(\mu) + \cosh(2\mu)(-3\cos(\mu) + \cos(3\mu) - \cos((2+i)\mu) + \cos((1+2i)\mu) + \cosh((2+i)\mu) - \cosh((1+2i)\mu)))}$$

$\times \, (i\cos((1+i)((1+i)x+1)\mu) - i\cos((1+i)((1+i)x\mu+\mu)) + 16\sin(2\mu) - 4\sin(4\mu) - (10+10i)\sin((1+i)\mu)$

$+ (2-2i)\sin((3+i)\mu) - (4+4i)\sin((2+2i)\mu) - (2-2i)\sin((1+3i)\mu) + (2+2i)\sin((3+3i)\mu)$

$- 2i\sin((1+i)((3+i)-x)\mu) - (6+6i)\sin((1+i)(x-2)\mu) + (2+2i)\sin((1+i)(x+2)\mu)$

$+ 6i\sin((1+i)((-1-i)+x)\mu) + 2\sin((1+i)((-3+i)+x)\mu) + 4\sin((1+i)(-2i+x)\mu) + 4i\sin((1+i)(2i+x)\mu)$

$- (2+4i)\sin((1+i)((1+i)+x)\mu) - 2i\sin((1+i)((-1+3i)+x)\mu) + (2-2i)\sin((1+i)((-2-2i)+x)\mu)$

$- 2\sin((1+i)((-1-3i)+x)\mu) + 2i\sin(((1+i)+2x)\mu) - 2i\sin(((1+3i)+2x)\mu) - 6i\sin((2i+(1-i)x)\mu)$

$+ (2+2i)\sin((4i+(1-i)x)\mu) - 2\sin(((2+4i)+(1-i)x)\mu) + 2\sin(2(ix+1)\mu) + 4i\sin((1+i)(ix+2)\mu)$

$+ 8\sin(((1-i)+2ix)\mu) - 2\sin(((3+i)+2ix)\mu) + 2\sin(((1-3i)+2ix)\mu) - 2\sin(((3-3i)+2ix)\mu)$

$+ \sin((1+i)((1+i)x+1)\mu) - (4+2i)\sin(((1+i)x+2)\mu) + 8i\sin(((1+i)-2x)\mu) + 2i\sin(((3+i)-2x)\mu)$

$- 2i\sin(((3+3i)-2x)\mu) - 6\sin((2-(1-i)x)\mu) + (2+2i)\sin((4-(1-i)x)\mu) + 2\sin(2(1-ix)\mu)$

$+ 4\sin((1+i)(2-ix)\mu) - 8\sin(2((1+i)-ix)\mu) + 2\sin(2((2+i)-ix)\mu) + 2\sin(2((1+2i)-ix)\mu)$

$+ 2\sin(((1-i)-2ix)\mu) - 2\sin(((3-i)-2ix)\mu) + 8\sin(((1+i)-2ix)\mu) + 2\sin(((1+3i)-2ix)\mu)$

$- 2\sin(((3+3i)-2ix)\mu) - 6\sin((2-(1+i)x)\mu) + (2-2i)\sin((4-(1+i)x)\mu) - (4-2i)\sin((1-i)x\mu$

$+ 2\mu) + \sin((1+i)((1+i)x\mu+\mu)) - 8i\sin((2+2i)\mu - 2x\mu) + 2i\sin((4+2i)\mu - 2x\mu)$

$+ 2i\sin((2+4i)\mu - 2x\mu) - 16\sinh(2\mu) + 4\sinh(4\mu) + (10+10i)\sinh((1+i)\mu)$

$- (2-2i)\sinh((3+i)\mu) + (4+4i)\sinh((2+2i)\mu) + (2-2i)\sinh((1+3i)\mu) - (2+2i)\sinh((3+3i)\mu)$

$+ (6+6i)\sinh((1+i)(x-2)\mu) - (2+2i)\sinh((1+i)(x+2)\mu) - 2i\sinh((1+i)((-3-i)+x)\mu) + 2i\sinh(2((-2-i)+x)\mu)$

$- 8i\sinh(2((-1-i)+x)\mu) - 2\sinh((1+i)((-3+i)+x)\mu) + 2i\sinh(2((-1-2i)+x)\mu) + 2\sinh((1+i)((-1-3i)+x)\mu)$

$- 2\sinh(2(ix+1)\mu) + 2\sinh(((3+i)+2ix)\mu) - 2\sinh((1+i)((1+i)x+1)\mu) + (4+2i)\sinh(((1+i)x+2)\mu)$

$- 2\sinh(2(1-ix)\mu) + 8\sinh(2((1+i)-ix)\mu) - 2\sinh(2((2+i)-ix)\mu) - 2\sinh(2((1+2i)-ix)\mu)$

$- 8\sinh(((1+i)-2ix)\mu) - 2\sinh(((1+3i)-2ix)\mu) + 2\sinh(((3+3i)-2ix)\mu))$

$$\psi(x) = \frac{\mu^2 \csc(\mu)}{8(2\cosh(\mu) + \cosh(2\mu)(-3\cos(\mu) + \cos(3\mu) - \cos((2+i)\mu) + \cos((1+2i)\mu) + \cosh((2+i)\mu) - \cosh((1+2i)\mu)))}$$

$$\times \; (i\cos((1+i)((1+i)x+1)\mu) - i\cos((1+i)((1+i)x\mu + \mu)) - 16\sin(2\mu) + 4\sin(4\mu) + (10-10i)\sin((1+i)\mu)$$

$$- (2+2i)\sin((3+i)\mu) + (4-4i)\sin((2+2i)\mu) + (2+2i)\sin((1+3i)\mu) - (2-2i)\sin((3+3i)\mu)$$

$$- 2\sin((1+i)((3+i)-x)\mu) - (6-6i)\sin((1+i)(x-2)\mu) + (2-2i)\sin((1+i)(x+2)\mu) + 6\sin((1+i)((-1-i)+x)\mu)$$

$$- 2i\sin((1+i)((-3+i)+x)\mu) - 4i\sin((1+i)(-2i+x)\mu) + 4\sin((1+i)(2i+x)\mu) - (4-2i)\sin((1+i)((1+i)+x)\mu)$$

$$- 2\sin((1+i)((-1+3i)+x)\mu) - (2+2i)\sin((1+i)((-2-2i)+x)\mu) + 2i\sin((1+i)((-1-3i)+x)\mu) - 2i\sin(((1+i)+2x)\mu)$$

$$+ 2i\sin(((1+3i)+2x)\mu) + 6\sin((2i+(1-i)x)\mu) - (2-2i)\sin((4i+(1-i)x)\mu) - 2i\sin(((2+4i)+(1-i)x)\mu)$$

$$+ 2\sin(2(ix+1)\mu) - 4\sin((1+i)(ix+2)\mu) + 8\sin(((1-i)+2ix)\mu) - 2\sin(((3+i)+2ix)\mu) + 2\sin(((1-3i)+2ix)\mu)$$

$$- 2\sin(((3-3i)+2ix)\mu) + \sin((1+i)((1+i)x+1)\mu) - (2-4i)\sin(((1+i)x+2)\mu) - 8i\sin(((1+i)-2x)\mu)$$

$$- 2i\sin(((3+i)-2x)\mu) + 2i\sin(((3+3i)-2x)\mu) - 6i\sin((2-(1-i)x)\mu) - (2-2i)\sin((4-(1-i)x)\mu) + 2\sin(2(1-ix)\mu)$$

$$+ 4i\sin((1+i)(2-ix)\mu) - 8\sin(2((1+i)-ix)\mu) + 2\sin(2((2+i)-ix)\mu) + 2\sin(2((1+2i)-ix)\mu) + 2\sin(((1-i)-2ix)\mu)$$

$$- 2\sin(((3-i)-2ix)\mu) + 8\sin(((1+i)-2ix)\mu) + 2\sin(((1+3i)-2ix)\mu) - 2\sin(((3+3i)-2ix)\mu) + 6i\sin((2-(1+i)x)\mu)$$

$$- (2+2i)\sin((4-(1+i)x)\mu) - (2+4i)\sin((1-i)x\mu + 2\mu) + \sin((1+i)((1+i)x\mu + \mu)) + 8i\sin((2+2i)\mu - 2x\mu)$$

$$- 2i\sin((4+2i)\mu - 2x\mu) - 2i\sin((2+4i)\mu - 2x\mu) - 16\sinh(2\mu) + 4\sinh(4\mu) + (10-10i)\sinh((1+i)\mu)$$

$$- (2+2i)\sinh((3+i)\mu) + (4-4i)\sinh((2+2i)\mu) + (2+2i)\sinh((1+3i)\mu) - (2-2i)\sinh((3+3i)\mu) - (6-6i)\sinh((1+i)(x-2)\mu)$$

$$+ (2-2i)\sinh((1+i)(x+2)\mu) + 2\sinh((1+i)((-3-i)+x)\mu) + 2i\sinh(2((-2-i)+x)\mu) - 8i\sinh(2((-1-i)+x)\mu)$$

$$- 2i\sinh((1+i)((-3+i)+x)\mu) + 2i\sinh(2((-1-2i)+x)\mu) + 2i\sinh((1+i)((-1-3i)+x)\mu) + 2\sinh(2(ix+1)\mu)$$

$$- 2\sinh(((3+i)+2ix)\mu) + 2\sinh((1+i)((1+i)x+1)\mu) - (2-4i)\sinh(((1+i)x+2)\mu) + 2\sinh(2(1-ix)\mu)$$

$$- 8\sinh(2((1+i)-ix)\mu) + 2\sinh(2((2+i)-ix)\mu) + 2\sinh(2((1+2i)-ix)\mu) + 8\sinh(((1+i)-2ix)\mu) + 2\sinh(((1+3i)-2ix)\mu)$$

$$- 2\sinh(((3+3i)-2ix)\mu))$$

with $\mu^4 = \lambda$. The adjoint of system of Equations (21)–(23) is

$$\begin{pmatrix} 3D^5 & 0 \\ -10D^3 & D^5 \end{pmatrix} \begin{pmatrix} \eta \\ \zeta \end{pmatrix} = \lambda \begin{pmatrix} 8D & 10D^3 \\ 0 & -24D \end{pmatrix} \begin{pmatrix} \eta \\ \zeta \end{pmatrix} \tag{24}$$

with

$$\eta = \eta'' = \eta''' = \zeta = \zeta' = \zeta'' = \zeta'''' = 0 \quad \text{at} \quad x = 0, \tag{25}$$

$$\zeta = \zeta'' + 2\eta = \zeta'''' - 4\eta'' = 0 \quad \text{at} \quad x = 1. \tag{26}$$

The solutions are

$$\eta(x) = \cos(2\mu x) + 2\cos((1+i)\mu x) + \cosh(2\mu x) + 2\cosh((1+i)\mu x)$$

$$+ \frac{(2-2i)(\sin(\mu) - \sinh(\mu))(\sin(\mu)\sinh(2\mu) - \sin(2\mu)\sinh(\mu))}{(\sin(2\mu) + \sinh(2\mu))(\sin(\mu)\cosh(\mu) - \cos(\mu)\sinh(\mu))}(\sin((1+i)\mu x) + \sinh((1+i)\mu x))$$

$$- \frac{(\cos(2\mu) + \cosh(2\mu) - 2)}{\sin(2\mu) + \sinh(2\mu)}(\sin(2\mu x) + \sinh(2\mu x)) - 6$$

$$\zeta(x) = \frac{1}{\mu^2}\Bigg(-\cos(2\mu x) - 2i\cos((1+i)\mu x) + \cosh(2\mu x) + 2i\cosh((1+i)\mu x)$$

$$+ \frac{(2+2i)(\sin(\mu) - \sinh(\mu))(\sin(\mu)\sinh(2\mu) - \sin(2\mu)\sinh(\mu))}{(\sin(2\mu) + \sinh(2\mu))(\sin(\mu)\cosh(\mu) - \cos(\mu)\sinh(\mu))}(\sinh((1+i)\mu x) - \sin((1+i)\mu x))$$

$$+ \frac{(\cos(2\mu) + \cosh(2\mu) - 2)}{\sin(2\mu) + \sinh(2\mu)}(\sin(2\mu x) - \sinh(2\mu x))\Bigg).$$

The bi-orthogonal set for $\{\boldsymbol{\phi}\}$ given by $\mathbf{y} = \mathcal{N}^T \boldsymbol{\eta}$ is:

$$
\begin{aligned}
y(x) = \; & \frac{24(1+i)\mu}{(\sin(2\mu) + \sinh(2\mu))(\sin(\mu)\cosh(\mu) - \cos(\mu)\sinh(\mu))} \\
& \times \Big(-(2-2i)\sin^2(\mu)\sinh(2\mu)\cos(\mu x)\cosh(\mu x) \\
& + \cos(\mu)\sinh(\mu)\left(-\sin(2\mu)(\sin((1+i)\mu x) - (2-2i)\sin(2\mu x)) + (2-2i)(\cos(2\mu) - 2)\cos(2\mu x) \right. \\
& + \sin(2\mu)(\sinh((1+i)\mu x) - (2-2i)\sinh(2\mu x)) + \sinh(2\mu)((2-2i)\sin(2\mu x) - \sin((1+i)\mu x) \\
& - (2-2i)\sinh(2\mu x) + \sinh((1+i)\mu x)) + (2-2i)\sin^2(\mu)\cos((1+i)\mu x) \\
& + (2-2i)(\cos(2\mu) - 2)\cosh(2\mu x) + (2-2i)\cosh(2\mu)(\cos(2\mu x) \\
& \left. + \cosh(2\mu x)) + (2-2i)\sin^2(\mu)\cosh((1+i)\mu x) \right) \\
& + \sin(\mu)\cosh(\mu)\left(\sin(2\mu)(\sin((1+i)\mu x) - (2-2i)\sin(2\mu x)) + (-2+2i)(\cos(2\mu) - 2)\cos(2\mu x) \right. \\
& + \sinh(2\mu)(-(2-2i)\sin(2\mu x) + \sin((1+i)\mu x) + (2-2i)\sinh(2\mu x) - \sinh((1+i)\mu x)) \\
& - \sin(2\mu)(\sinh((1+i)\mu x) - (2-2i)\sinh(2\mu x)) - (2-2i)(\cos(2\mu) - 2)\cosh(2\mu x) \\
& \left. - (2-2i)\cosh(2\mu)(\cos(2\mu x) + \cosh(2\mu x)) + (4-4i)\sinh^2(\mu)\cos(\mu x)\cosh(\mu x) \right) \\
& + (-2+2i)\sin(2\mu)\sinh^2(\mu)\cos(\mu x)\cosh(\mu x) \Big)
\end{aligned}
$$

$$
\begin{aligned}
z(x) = \; & -\frac{48}{\mu}\Big(\sin(2\mu x) + (-1+i)\sin((1+i)\mu x) + \sinh(2\mu x) - (1-i)\sinh((1+i)\mu x) \\
& + \frac{(\cos(2\mu) + \cosh(2\mu) - 2)}{\sin(2\mu) + \sinh(2\mu)}(\cos(2\mu x) - \cosh(2\mu x)) \\
& + \frac{2i(\sin(\mu) - \sinh(\mu))(\sin(\mu)\sinh(2\mu) - \sin(2\mu)\sinh(\mu))}{(\sin(2\mu) + \sinh(2\mu))(\sin(\mu)\cosh(\mu) - \cos(\mu)\sinh(\mu))}(\cosh((1+i)\mu x) - \cos((1+i)\mu x)) \Big).
\end{aligned}
$$

4. Results for Direct Singular and Inverse SLPs

Here, we present some numerical examples of direct and inverse SLPs of orders 2 and 4.

Algorithm 1 was implemented using MATLAB (2014) [42]. The reference solutions of the EVPs are computed using Wolfram Mathematica 11 [41]. Direct problems were solved using the Magnus expansion method [39].

For the numerical calculations, n is the number of subdivisions in the interval $[a, b]$, m is the number of subdivisions in the interval $[\lambda_0, \lambda^*]$; λ^* being the maximum eigenvalue searching, L is the number of multisection steps used to calculate each eigenvalue in the characteristic function and M is the number of inverse algorithm steps.

Error values are stated using max-norm which is defined as $||\mathbf{x}||_\infty := \max_n(|x_n|)$.

4.1. Singular SLPs

Example 1 (Singular SLP: infinite domain)**.** *This example extends the Magnus method, as described in [39] to the infinite domain. Consider the harmonic oscillator problem*

$$
u'' + (\lambda - x^2)u = 0, \quad x \in (-\infty, \infty) \tag{27}
$$

together with Dirichlet and Neumann BCs, having exact eigenvalues $\lambda_k = 2k + 1$, $n = 0, 1, \ldots$ [43].

The domain is truncted to $[-l, l]$, and using the parameters $m = 10$, $n = 100$, $L = 10$ the first three eigenvalues are computed (see Table 1). It can be deduced that the Magnus method has comparable performance to other methods. Moreover, fixed parameters can be changed to obtain better accuracy.

Table 1. Absolute errors for the first three eigenvalues of Equation (27) from Table 1 of Taşeli [43] and using Magnus method.

λ	l	Dirichlet Error		Neumann Error	
		[43]	Magnus Method	[43]	Magnus Method
	4.5	1.70×10^{-8}	1.58×10^{-8}	1.50×10^{-8}	1.68×10^{-8}
1	6.0	2.60×10^{-11}	5.80×10^{-10}	2.00×10^{-12}	5.80×10^{-10}
	7.5	2.00×10^{-14}	2.20×10^{-9}	0.00×10^{0}	2.20×10^{-9}
	4.5	1.88×10^{-6}	1.10×10^{-5}	1.10×10^{-5}	1.19×10^{-5}
5	6.0	3.30×10^{-8}	8.02×10^{-9}	2.73×10^{-9}	8.00×10^{-9}
	7.5	5.50×10^{-11}	3.07×10^{-8}	6.00×10^{-12}	3.07×10^{-8}
	4.5	9.99×10^{-1}	9.85×10^{-4}	9.85×10^{-4}	1.12×10^{-3}
9	6.0	6.01×10^{-6}	3.06×10^{-8}	6.11×10^{-7}	2.54×10^{-8}
	7.5	1.70×10^{-8}	1.08×10^{-7}	2.00×10^{-9}	1.08×10^{-7}

Example 2 (Singular SLP with asymmetric potential). *Consider the Morse potential problem*

$$-u'' - (e^{-\alpha x} - 1)^2 u = \lambda u, \quad x \in (-\infty, \infty) \tag{28}$$

together with Dirichlet and Neumann BCs. When $\lambda \in (0,1)$ this has a finite number of EV: $\lambda_k = \beta_k(2 - \beta_k)$, with $\beta_k = \frac{1}{2}\alpha(2k + 1)$, $k = 0, 1, \ldots, K$, where $K = \left\lfloor \frac{1}{\alpha} - \frac{1}{2} \right\rfloor$ [43].

The domain is truncated to $[-l, l]$, and using the parameters $m = 10$, $n = 100$, $L = 10$ and $\alpha = 0.02$, the first three eigenvalues are computed (see Table 2). Magnus method is more accurate when the truncated interval is small. Furthermore, it can be observed that initial eigenvalues are more accurate than the later ones.

Table 2. Absolute errors for the first three eigenvalues of Equation (28) from Table 2 of Taşeli [43] and using Magnus method.

λ	l	Dirichlet Error		Neumann Error	
		[43]	Magnus Method	[43]	Magnus Method
	40	1.10×10^{-8}	5.60×10^{-12}	1.90×10^{-9}	1.12×10^{-11}
0.0199	50	6.07×10^{-10}	3.17×10^{-11}	6.00×10^{-11}	3.17×10^{-11}
	60	1.15×10^{-13}	9.39×10^{-11}	2.48×10^{-14}	9.39×10^{-11}
	40	3.56×10^{-7}	3.78×10^{-10}	3.90×10^{-8}	3.34×10^{-10}
0.0591	50	2.20×10^{-8}	1.12×10^{-10}	5.97×10^{-9}	1.12×10^{-10}
	60	5.00×10^{-12}	3.37×10^{-10}	6.75×10^{-13}	3.37×10^{-10}
	40	7.69×10^{-7}	1.85×10^{-8}	1.13×10^{-7}	1.93×10^{-8}
0.0975	50	6.65×10^{-7}	3.83×10^{-10}	1.29×10^{-7}	3.82×10^{-10}
	60	2.50×10^{-11}	1.15×10^{-9}	3.18×10^{-12}	1.15×10^{-9}

Example 3 (Singular SLP in half axis). *Consider the hydrogen atom equation*

$$-u'' - \left(\frac{2}{x^2} - \frac{1}{x} \right) u = \lambda u, \quad x \in (0, \infty) \tag{29}$$

together with Dirichlet and Neumann BCs. This has a finite number of eigenvalues lying between $-1 < \lambda < 0$, given by $\lambda_k = -\dfrac{1}{4(k + 2)^2}$, with $k = 0, 1, \ldots$, [44].

The domain is truncated to $[0, 100]$, and using the parameters $m = 100$, $n = 200$, $L = 10$ the first few eigenvalues are computed (see Table 3).

Table 3. Eigenvalues and absolute errors for the first five eigenvalues of Equation (29) using Magnus method.

| k	Magnus Value	Magnus Error
1	$-6.2323550470257 \times 10^{-2}$	1.76×10^{-4}
2	$-2.7717584125111 \times 10^{-2}$	6.02×10^{-5}
3	$-1.5581393052633 \times 10^{-2}$	4.36×10^{-5}
4	$-9.0660619438460 \times 10^{-3}$	9.34×10^{-4}
5	$-1.6285043066920 \times 10^{-3}$	5.32×10^{-3}

4.2. Inverse SLPs

Example 4 (Inverse SLP with a different domain). *This example extends Barcilon's algorithm in [36] by changing the domain $[0, 1]$ to $[0, \pi]$:*

$$u'' + (\lambda - \cos(x))u = 0, \quad x \in (0, \pi)$$

Here, $p(x)$ is symmetric and normalized. In the inverse Algorithm 1 of [39], $w_n^{(0)}(x)$ and $\omega_n^{(0)}(x)$ need to be changed into $\{w_n^{(0)}\}_1^\infty = \{\cos(nx) - \cos(n\pi)\}_1^\infty$, $\{\omega_n^{(0)}\}_1^\infty = \{\sin(nx)\}_1^\infty$ which are the basic solutions in the new domain. The truncated eigenvalue sequences at $N = 6$, for Dirichlet boundary conditions (DDBCs) and Dirichlet– Neumann boundary conditions (DNBCs) are calculated using the Magnus method (with $m = 20$, $n = 20$, $L = 15$) and $M = 10$ inverse algorithm steps are used.

Figure 1 shows the reconstructed and exact potential, and the log absolute errors in the reconstructed potential, respectively. From Figure 1a, it is obvious that the potential is converging towards the exact one. The max-norm of difference between the reconstructed p and the actual p is $\approx 3.35 \times 10^{-3}$ and the max-norm of difference between the reconstructed eigenvalues and the actual ones is $\approx 1.62 \times 10^{-3}$.
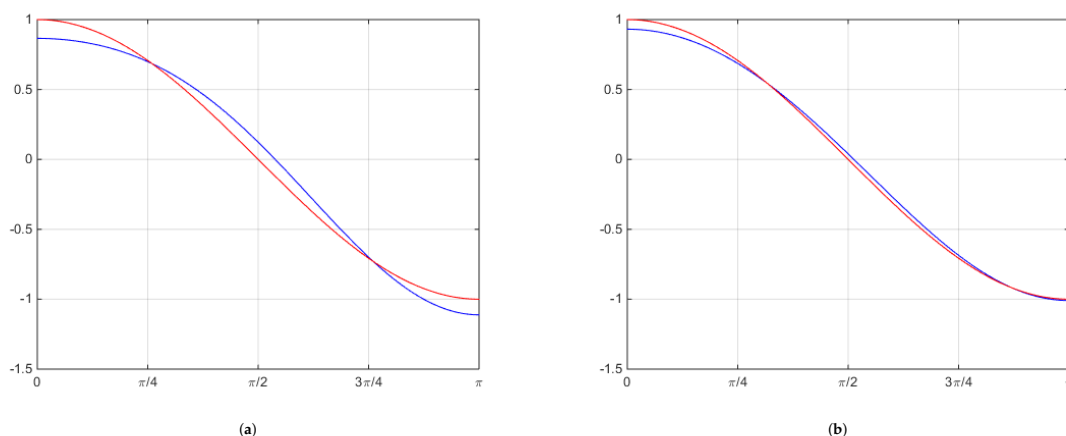

(a)

(b)
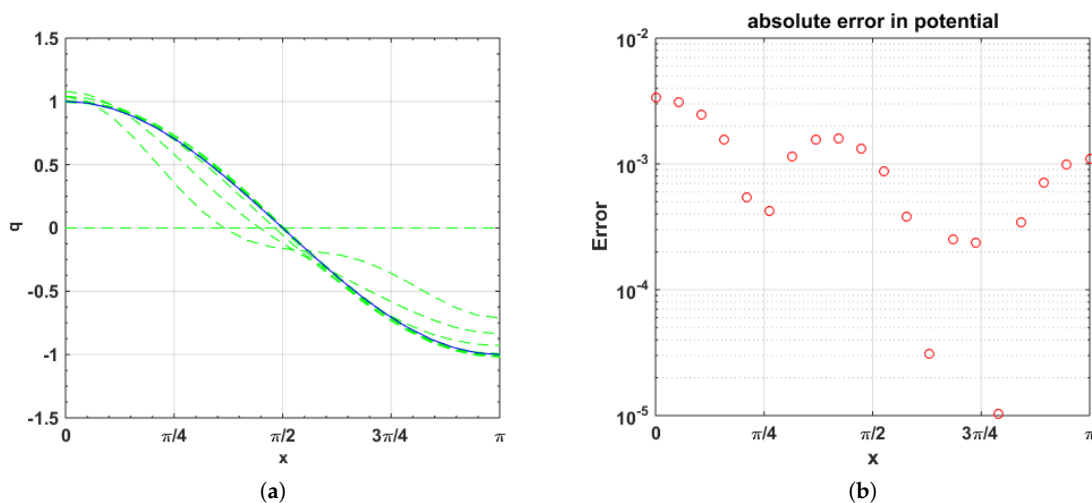
Figure 1. (**a**) Iteratively reconstructed potential (green dashed line) and exact potential (blue solid line) $p(x) = \cos(x)$ (**b**) Log absolute error in the reconstructed p.

Figure 2 shows reconstructed p starting with perturbed eigenvalues

$$\sigma_k^\delta = \sigma_k + \delta\,\sigma_k \cdot rand(size(\sigma_k)), \quad k = 1, \dots, 2N$$

where δ is the noise level and $N = 5$, $m = 20$, $n = 100$, $L = 5$, $M = 6$.

(a)　　　　　　　　　　　　　　　　　　　(b)

Figure 2. Reconstructed potential (blue) and exact potential (red) $p(x) = \cos(x)$ (**a**) $\delta = 0.1$ (**b**) $\delta = 0.05$.

It is obvious that reconstruction is possible even in the presence of a significant noise.

Example 5 (Inverse SLP with a non-smooth potential)**.** *This example extends Barcilon's algorithm in [36] by reconstructing a non-smooth potential:*

$$u'' + (\lambda - p(x))u = 0, \quad x \in (0,2)$$

with $p(x) = |1 - x| - 0.5$: symmetric and normalized, but non-smooth. The truncated eigenvalue sequences at $N = 4$, for DDBCs and DNBCs are calculated using the Magnus method (with $m = 10$, $n = 100$, $L = 5$), and used in the reconstruction of the potential.

Figure 3 shows the reconstructed and exact potential, and the log absolute errors in the reconstructed potential, respectively, using one inverse algorithm step. From Figure 3a, it is obvious that the potential is converging towards the exact one. The max-norm of difference between the reconstructed p and the actual p is $\approx 1.13 \times 10^{-1}$ and the max-norm of difference between the reconstructed eigenvalues and the actual ones is $\approx 3.15 \times 10^{1}$. As anticipated, the error is maximum at the point of non-differentiability and at the boundaries.
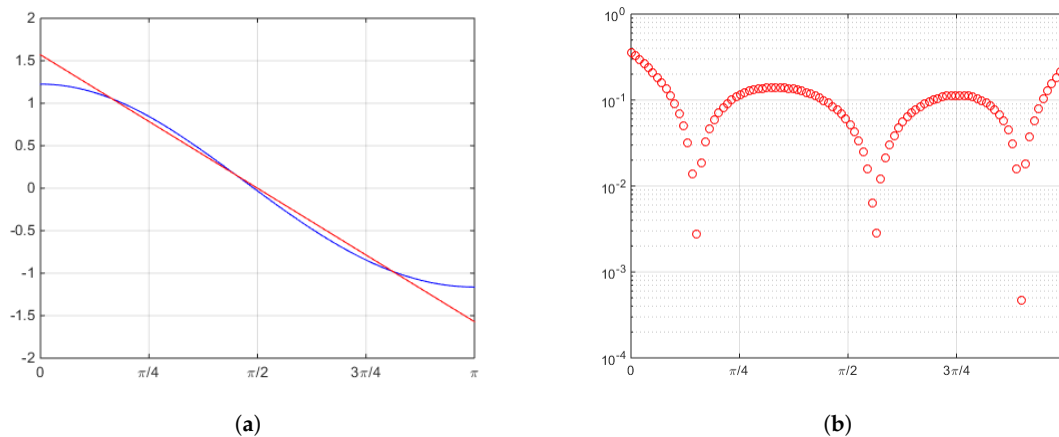

(a)　　　　　　　　　　　　　　　　　　　(b)
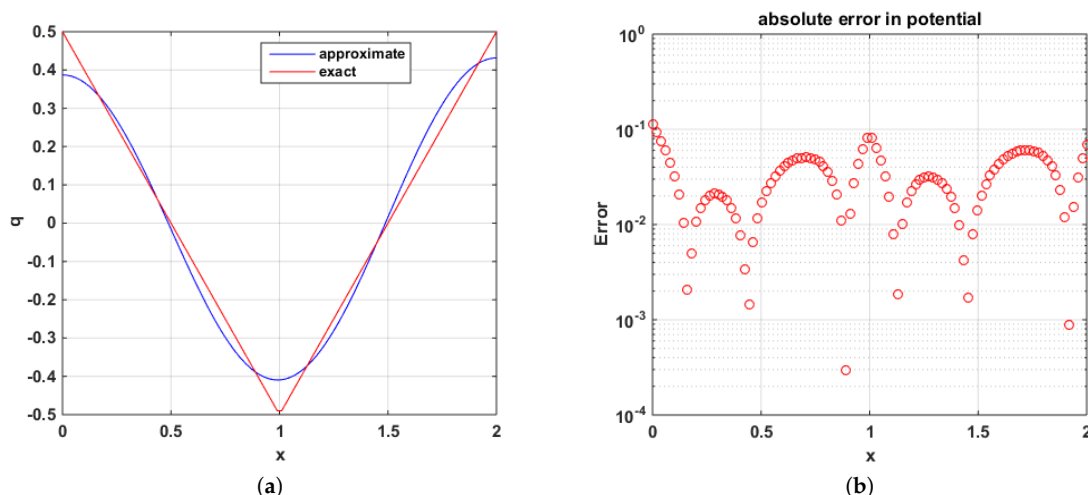
Figure 3. (**a**) Reconstructed potential (blue) and exact potential (red) $p(x) = |1 - x| - 0.5$ (**b**) Log absolute error in the reconstructed p.

Example 6 (FSLP)**.** *Consider*

$$y^{(4)} - (p_1 y')' + p_2 y = \lambda y, \qquad 0 < x < \pi$$

with $p_1 \equiv 0$ and $p_2(x) = \pi/2 - x$. Using the three spectra corresponding to the sets of BCs: $\{y(0) = y''(0) = y(1) = y''(1) = 0\}$, $\{y(0) = y'(0) = y(1) = y''(1) = 0\}$ and $\{y'(0) = y''(0) = y(1) = y''(1) = 0\}$ two potential functions p_1, p_2 are reconstructed starting with zero initial guesses and $N = 4, m = 20, n = 100, L = 5, M = 6$ (Figure 4). The max-norm of difference between the reconstructed p_2 and actual p_2 is $\approx 3.5 \times 10^{-1}$ and the max-norm of difference between the reconstructed eigenvalues and the actual ones is $\approx 2.1 \times 10^{-1}$.
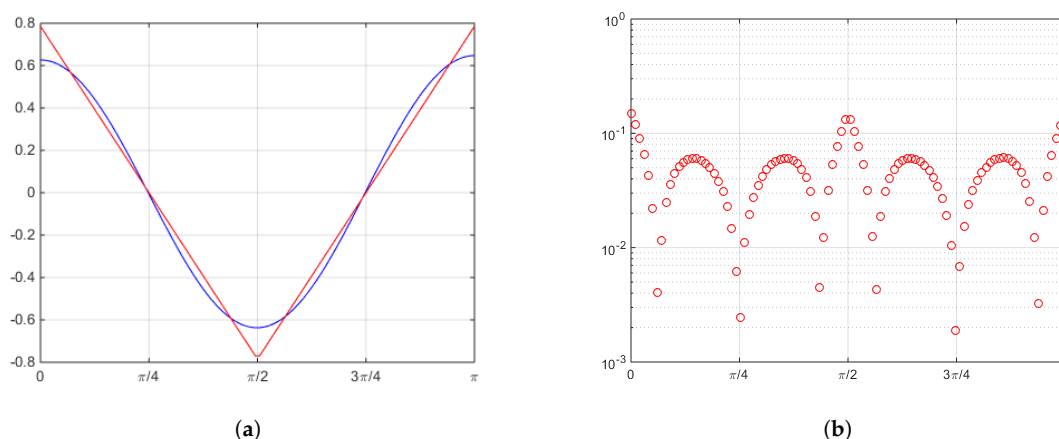

(a)

(b)

Figure 4. (**a**) Reconstructed potential (blue) and exact potential (red) $p_2(x) = \pi/2 - x$ (**b**) Log absolute error in the reconstructed p_2.

Example 7 (FSLP). *Consider*

$$y^{(4)} - (p_1 y')' + p_2 y = \lambda y, \qquad 0 < x < \pi$$

with $p_1(x) = |x - \pi/2| - \pi/4$ and $p_2 \equiv 0$. Using the three spectra corresponding to the sets of BCs: $y(0) = y''(0) = y(1) = y''(1) = 0$, $y(0) = y'(0) = y(1) = y''(1) = 0$ and $y'(0) = y''(0) = y(1) = y''(1) = 0$ two potential functions p_1, p_2 are reconstructed starting with zero initial guesses and $N = 4, m = 20, n = 100, L = 15, M = 6$ (Figure 5). The max-norm of difference between the reconstructed p_1 and the actual p_1 is $\approx 1.4 \times 10^{-1}$ and the max-norm of difference between the reconstructed eigenvalues and the actual ones is $\approx 7.5 \times 10^{-1}$.

(a)

(b)

Figure 5. (**a**) Reconstructed potential (blue) and exact potential (red) $p_1(x) = |x - \pi/2| - \pi/4$ (**b**) Log absolute error in the reconstructed p_1.

5. Discussion

In the first few examples, the Magnus method [39] is further extended to solve some direct singular SLPs on infinite domain, half-axis and for an asymmetric potential. Later, Barcilon's algorithm is extended to solve inverse SLPs in a non–unit domain and for a non–smooth potential. The first two

examples, which extend the Magnus method in [39] to infinite interval, illustrates it's applicability to singular SLPs effectively. Furthermore, the present method does not require transforming the singular SLP to a regular one or re-scaling the eigenvalues. It effectively finds the required eigenvalues, just by truncating the interval to a finite one.

Next, the inverse SLP is solved with smooth and non-smooth potential, even in the presence of noise. It is observed that the method is successful even in the presence of significant noise, provided that the assumptions of the algorithm are satisfied.

The last few examples solve the inverse FSLP using Barcilon's algorithm with the initial knowledge of three spectra. A simplified FSLP is solved keeping one of the unknown potential functions zero.

6. Conclusions

More than 40 years after Barcilon's paper [36], this paper gives a concrete implementation of the inverse SLP algorithm proposed therein, to our knowledge for the first time. Furthermore, computational feasibility and applicability of this algorithm for solving inverse SLPs of higher order is verified successfully in this paper (for $n = 2$ and $n = 4$).

In future work, it is possible to extend the results to solve even higher order SLPs.

In conclusion, this work provides a method that can be adapted successfully for solving a direct (regular/singular) or inverse SLP of an arbitrary order with arbitrary BCs.

References

1.　Ambarzumian, V. Über eine frage der eigenwerttheorie. *Z. Phys. A Hadron. Nucl.* **1929**, *53*, 690–695.

2.　Tikhonov, A. On the uniqueness of the solution of the problem of electromagnetic sounding. *Dokl. Akad. Nauk SSSR* **1949**, *69*, 797–800.

3.　Levinson, N. The Inverse Sturm-Liouville Problem. *Mat. Tidsskr. B* **1949**, *1949*, 25–30.

4.　Marčenko, V. *Sturm-Liouville Operators and Their Applications*; Birkhauser Verlag: Basel, Switzerland, 1986.

5.　Levitan, B. *Inverse Sturm-Liouville Problems*; VNU Science Press: Utrecht, The Netherlands, 1987.

6.　Kravchenko, V.V. *Direct and Inverse Sturm-Liouville Problems: A Method of Solution*; Springer Nature: Berlin/Heidelberg, Germany, 2020.

7.　McLaughlin, J.R. Analytical methods for recovering coefficients in differential equations from spectral data. *SIAM Rev.* **1986**, *28*, 53–72. [CrossRef]

8.　Freiling, G.; Yurko, V.A. *Inverse Sturm-Liouville Problems and Their Applications*; NOVA Science Publishers: New York, NY, USA, 2001.

9.　Barcilon, V. Iterative solution of the inverse Sturm-Liouville problem. *J. Math. Phys.* **1974**, *15*, 429–436. [CrossRef]

10.　Sacks, P.E. An iterative method for the inverse Dirichlet problem. *Inverse Probl.* **1988**, *4*, 1055–1069. [CrossRef]

11.　Hald, O.H. The inverse Sturm-Liouville problem and the Rayleigh-Ritz method. *Math. Comput.* **1978**, *32*, 687–705. [CrossRef]

12.　Paine, J. A numerical method for the inverse Sturm-Liouville problem. *SIAM J. Sci. Stat. Comput.* **1984**, *5*, 149–156. [CrossRef]

13.　Rundell, W.; Sacks, P.E. Reconstruction techniques for classical inverse Sturm-Liouville problems. *Math. Comput.* **1992**, *58*, 161–183. [CrossRef]

14.　Lowe, B.D.; Pilant, M.; Rundell, W. The recovery of potentials from finite spectral data. *SIAM J. Math. Anal.* **1992**, *23*, 482–504. [CrossRef]

15.　Neher, M. Enclosing solutions of an inverse Sturm-Liouville problem with finite data. *Computing* **1994**, *53*, 379–395. [CrossRef]

16. Fabiano, R.H.; Knobel, R.; Lowe, B.D. A finite-difference algorithm for an inverse Sturm-Liouville problem. *IMA J. Numer. Anal.* **1995**, *15*, 75–88. [CrossRef]

17. Ghelardoni, P. Approximations of Sturm-Liouville eigenvalues using boundary value methods. *Appl. Numer. Math.* **1997**, *23*, 311–325. [CrossRef]

18. Aceto, L.; Ghelardoni, P.; Magherini, C. BVMs for Sturm-Liouville eigenvalue estimates with general boundary conditions. *J. Numer. Anal. Ind. Appl. Math* **2009**, *4*, 113–127.

19. Ghelardoni, P.; Magherini, C. BVMs for computing Sturm-Liouville symmetric potentials. *Appl. Math. Comput.* **2010**, *217*, 3032–3045. [CrossRef]

20. Aceto, L.; Ghelardoni, P.; Magherini, C. Boundary value methods for the reconstruction of Sturm-Liouville potentials. *Appl. Math. Comput.* **2012**, *219*, 2960–2974. [CrossRef]

21. Andrew, A.L. Numerov's method for inverse Sturm-Liouville problems. *Inverse Probl.* **2005**, *21*, 223–238. [CrossRef]

22. Andrew, A.L. Numerical solution of inverse Sturm-Liouville problems. *ANZIAM J.* **2004**, *45*, 326–337. [CrossRef]

23. Andrew, A.L. Computing Sturm-Liouville potentials from two spectra. *Inverse Probl.* **2006**, *22*, 2069–2081. [CrossRef]

24. Röhrl, N. A least-squares functional for solving inverse Sturm–Liouville problems. *Inverse Probl.* **2005**, *21*, 2009–2017. [CrossRef]

25. Rafler, M.; Böckmann, C. Reconstruction method for inverse Sturm–Liouville problems with discontinuous potentials. *Inverse Probl.* **2007**, *23*, 933–946. [CrossRef]

26. Efremova, L.; Freiling, G. Numerical solution of inverse spectral problems for Sturm-Liouville operators with discontinuous potentials. *Open Math.* **2013**, *11*, 2044–2051. [CrossRef]

27. Ignatiev, M.; Yurko, V. Numerical Methods for Solving Inverse Sturm-Liouville Problems. *Results Math.* **2008**, *52*, 63–74. [CrossRef]

28. Liu, C.S. Solving an inverse Sturm-Liouville problem by a Lie-group method. *Bound. Value Probl.* **2008**, *2008*, 749865. [CrossRef]

29. Böckmann, C.; Kammanee, A. Broyden method for inverse non-symmetric Sturm-Liouville problems. *BIT Numer. Math.* **2011**, *51*, 513–528. [CrossRef]

30. Böckmann, C.; Rattana, A. An inverse fourth order sturm-liouville problem. *AIP Conf. Proc.* **2015**, *1648*, 850030. [CrossRef]

31. Gao, Q. Decent flow methods for inverse Sturm–Liouville problem. *Appl. Math. Model.* **2012**, *36*, 4452–4465. [CrossRef]

32. Gao, Q.; Cheng, X.; Huang, Z. Modified Numerov's method for inverse Sturm–Liouville problems. *J. Comput. Appl. Math.* **2013**, *253*, 181–199. [CrossRef]

33. Drignei, M.C. A Newton-type method for solving an inverse Sturm-Liouville problem. *Inverse Probl. Sci. Eng.* **2014**, *23*, 851–883. [CrossRef]

34. Kravchenko, V.V. On a method for solving the inverse Sturm-Liouville problem. *J. Inverse Ill Posed Probl.* **2018**. [CrossRef]

35. Neamaty, A.; Akbarpoor, S.; Yilmaz, E. Solving Symmetric Inverse Sturm-Liouville Problem Using Chebyshev Polynomials. *Mediterr. J. Math.* **2019**, *16*, 74. [CrossRef]

36. Barcilon, V. On the solution of inverse eigenvalue problems of high orders. *Geophys. J. Int.* **1974**, *39*, 143–154. [CrossRef]

37. McLaughlin, J.R. An Inverse Eigenvalue Problem of Order Four. *SIAM J. Math. Anal.* **1976**, *7*, 646–661. [CrossRef]

38. McLaughlin, J.R. An Inverse Eigenvalue Problem of Order Four—An Infinite Case. *SIAM J. Math. Anal.* **1978**, *9*, 395–413. [CrossRef]

39. Perera, U.; Böckmann, C. Solutions of Direct and Inverse Even-Order Sturm-Liouville Problems Using Magnus Expansion. *Mathematics* **2019**, *7*, 544. [CrossRef]

40. Barcilon, V. On the uniqueness of inverse eigenvalue problems. *Geophys. J. R. Astron. Soc.* **1974**, *38*, 287–298. [CrossRef]

41. *Mathematica*; Wolfram Research, Inc.: Champaign, IL, USA, 2016.

42. *Matlab*; The MathWorks, Inc.: Natick, MA, USA, 2014.

43. Taşeli, H. Accurate numerical bounds for the spectral points of singular Sturm–Liouville problems over $-\infty < x < \infty$. *J. Comput. Appl. Math.* **2000**, *115*, 535–546. [CrossRef]

44. Marletta, M.; Pryce, J.D. Automatic solution of Sturm-Liouville problems using the Pruess method. *J. Comput. Appl. Math.* **1992**, *39*, 57–78. [CrossRef]

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.