Exponentially Convergent Numerical-Analytical Method for Solving Eigenvalue Problems for Singular Differential Operators

V. L. Makarov, D. V. Dragunov, D. V. Bohdan

The paper summarizes the authors' recent work on developing and proving an exponentially convergent numerical-analytical method (the FD-method) for solving Sturm-Liouville problems with a singular Legendre operator and a singular potential. It gives a concise general overview of the FD-method, outlines the proof of its convergence and exponential convergence rate when applied to the particular problem at hand and talks briefly about its software implementation.

Introduction

What follows presents a summary of the article [1]. In the article the authors generalize the results found in [2] and [3], which concern the subject of solving the Sturm-Liouville problem

$$-\frac{d}{dx}\left[(1-x^2)\frac{du(x)}{dx}\right]+q(x)u(x)=\lambda u(x), \quad x\in(-1,1), \tag{1}$$

$$\lim_{x\to\pm1}(1-x^2)\frac{du(x)}{dx}=0. \tag{2}$$

Such problems arise in applications like solving partial differential equations in spherical coordinates using separation of variables, as is done, e.g., with *hydrogen-molecule ion's equation* in [4] (see [4, p. 167–170]).

The authors' article [1] extends the FD-method for solving problem (1), (2) previously developed in [2], [3] to the case when the potential function $q(x)$ is such that

$$\|q\|_{1,\rho}=\int\limits_{-1}^{1}\frac{|q(x)|}{\sqrt{1-x^2}}dx<\infty. \tag{3}$$

The preceding articles consider a more limited case of piecewise continuous functions that are bounded on the closed interval $[-1,1]$ and have no more than a finite number of jump discontinuities.

This extension was prompted by the numerical convergence of the FD-method when applied to problem (1), (2) with the potential $q(x)=|x+1/3|^{1/2}+\ln(|x-1/3|)$, which does not belong to the class $Q^0[-1,1]$, shown in [3].

Under the new conditions on $q(x)$ the problem is enough of a generalization of those considered in [2] and [3] that the proof techniques used therein couldn't be applied. Instead, to obtain the sufficient conditions for convergence a new approach was used based on an inequality for Legendre functions proposed by V. L. Makarov.

The inequality (see Theorem 2) follows from Theorem 1, analogues of which the authors were unable to find. These theorems are the novel and original results in the article.

Overview of the FD-method

Below a solving algorithm for problem (1), (2) is constructed based on the general idea of the FD-method (see [5]).

The authors have proved that the eigenvalues of problem (1), (2) form an increasing sequence $\lambda_0 < \lambda_1 < \ldots < \lambda_n < \ldots$

We are going to look for the eigensolution $u_n(x), \lambda_n$ to eigenvalue problem (1), (2) in the form of a series

$$u_n(x) = \sum_{j=0}^{\infty} u_n^{(j)}(x), \quad \lambda_n = \sum_{j=0}^{\infty} \lambda_n^{(j)}, \tag{4}$$

where the pair $u_n^{(j)}(x), \lambda_n^{(j)}$ can be found as the solution to the following system of recurrence problems:

$$\frac{d}{dx}\left[(1-x^2)\frac{du_n^{(j)}(x)}{dx}\right] + \lambda_n^{(0)} u_n^{(j)}(x) = F_n^{(j)}(x), \tag{5}$$

$$F_n^{(j)}(x) = -\sum_{i=0}^{j-1} \lambda_n^{(j-i)} u_n^{(i)}(x) + q(x) u_n^{(j-1)}(x), \ j = 1, 2, \ldots \tag{6}$$

$$\lim_{x\to\pm 1} (1-x^2)\frac{du_n^{(j)}(x)}{dx} = 0, \quad j = 0, 1, 2, \ldots. \tag{7}$$

Although problem (5), (6), (7) does not possess a unique solution, the most convenient one (from the computational standpoint) can be found recursively through applying the following formulas:

$$u_n^{(0)}(x) = \sqrt{\frac{2n+1}{2}} P_n(x), \quad \lambda_n^{(0)} = n(n+1), \quad n = 0, 1, 2, \ldots. \tag{8}$$

$$u_n^{(j)}(x) = c_n^{(j)} u_n^{(0)}(x) + \int_{-1}^{x} K_n(x,\xi) F_n^{(j)}(\xi) d\xi, \tag{9}$$

$$c_n^j = -\int_{-1}^{1} u_n^{(0)}(x) \int_{-1}^{x} K_n(x,\xi) F_n^{(j)}(\xi) d\xi dx, \tag{10}$$

$$\lambda_n^{(j)} = \int_{-1}^{1} q(x) u_n^{(0)}(x) u_n^{(j-1)}(x) dx, \quad j \in \mathbb{N}, \tag{11}$$

where $K_n(x,\xi) = P_n(x)Q_n(\xi) - Q_n(x)P_n(\xi)$ and $P_n(x), Q_n(x)$ denote Legendre functions of the first and the second kind respectively.

Justification outline

The following statements are proven in [1].

We establish the result stated in Theorem 1, which is then used to prove the convergence of the FD-method.

Theorem 1 *Suppose that $u_I(\theta)$ and $u_{II}(\theta)$ are a pair of solutions to the differential equation*

$$\frac{d^2u(\theta)}{d\theta^2} + \phi(\theta)u(\theta) = 0, \quad \theta \in (a,b)\,, \tag{12}$$

$$\phi(\theta) \in C^1(a,b), \phi(\theta) > 0, \forall\theta \in (a,b)$$

that satisfy the following condition:

$$W(\theta) = u_I(\theta)u'_{II}(\theta) - u'_I(\theta)u_{II}(\theta) = 1, \quad \forall\theta \in (a,b). \tag{13}$$

If there exists a point $c \in (a,b)$ such that $\phi'(\theta) \leq 0$ $\forall\theta \in (a,c]$ and $\phi'(\theta) \geq 0$ $\forall\theta \in [c,b)$ then

$$\left|v(\theta,\tilde{\theta})\right| \leq \sqrt{2\phi^{-1}(c)}, \forall\theta,\tilde{\theta} \in (a,b), \tag{14}$$

$$v(\theta,\tilde{\theta}) \stackrel{def}{=} u_I(\theta)u_{II}(\tilde{\theta}) - u_I(\tilde{\theta})u_{II}(\theta).$$

If $\phi'(\theta) \leq 0$ or $\phi'(\theta) \geq 0$ $\forall\theta \in (a,b)$ then

$$\left|v(\theta,\tilde{\theta})\right| \leq \max\left\{\sqrt{\phi^{-1}(\theta)}, \sqrt{\phi^{-1}(\tilde{\theta})}\right\}, \forall\theta,\tilde{\theta} \in (a,b). \tag{15}$$

The function $\phi(\theta) = (2\sin(\theta))^{-2} + (\nu + 1/2)^2$ fulfils all the requirements of Theorem 1 with $c = \pi/2$. Therefore, Theorem 1 provides us with the estimation

$$\sqrt{\sin(\theta)\sin(\tilde{\theta})} \cdot \left|P_\nu(\cos(\theta))Q_\nu(\cos(\tilde{\theta}) - P_\nu(\cos(\tilde{\theta}))Q_\nu(\cos(\theta)\right| \leq \sqrt{2\phi^{-1}(\pi/2)} \leq$$

$$\leq \sqrt{\frac{2}{\frac{1}{4} + \left(\nu + \frac{1}{2}\right)^2}},$$

$\forall\theta,\tilde{\theta} \in (0,\pi)$ and the following corollary:

Theorem 2 *For every $\nu \in \mathbb{R}$ it holds true that*

$$\sqrt[4]{(1-x^2)(1-\xi^2)}\,|P_\nu(x)Q_\nu(\xi) - P_\nu(\xi)Q_\nu(x)| \leq \sqrt{\frac{2}{\frac{1}{4} + \left(\nu + \frac{1}{2}\right)^2}}. \tag{16}$$

Theorem 3 *Let* $n_0 = \left[\frac{3\sqrt{2}\pi}{3-2\sqrt{2}}\|q\|_{1,\rho}\right]^{1} + 1$ *and* $\tilde{\alpha}_n = \frac{3\sqrt{2}\pi}{n}\|q\|_{1,\rho}$. *The FD-method described by formulas* (4), (8), (9), (10) *and* (11) *converges to the eigensolution* $(u_n(x); \lambda_n)$ *of problem* (1), (2) *for all* $n > n_0$. *Furthermore, for the* $n > n_0$ *the following estimations of the method's convergence rate hold true:*

$$\left\| u_n(x) - \overset{m}{u}_n(x) \right\|_{\infty, 1/\sqrt{\rho}} \leq \frac{\tilde{\alpha}_n^{m+1}}{(2m+3)\sqrt{\pi(m+2)}(1-\tilde{\alpha}_n)}, \tag{17}$$

$$\left| \lambda_n - \overset{m}{\lambda}_n \right| \leq \|q\|_{1,\rho} \frac{\tilde{\alpha}_n^{m}}{(2m+1)\sqrt{\pi(m+1)}(1-\tilde{\alpha}_n)}, \tag{18}$$

where $\overset{m}{u}_n(x) = \sum\limits_{j=0}^{m} u_n^{(j)}(x)$, $\overset{m}{\lambda}_n = \sum\limits_{j=0}^{m} \lambda_n^{(j)}$.

Software implementation

The software implementation was written in Python 2.7 using the libraries NumPy, SciPy, mpmath and matplotlib. The use of the NumPy library has allowed us to have floating-point variables with up to quadruple precision[2]. We faced a technical problem when trying to compute the values of Legendre Q_n function for an argument that's sufficiently close to ± 1 using SciPy's `lqmn` to circumvent which we had to resort to calling the corresponding function `legenq` of the mpmath library. This process involved converting the argument of `legenq` from the data type `numpy.longdouble` to `mpf` and back again with sufficient precision.

In the description of the algorithm we use the *tanh rule* and *Stenger's formula* in order to approximate integration in formulas (9), (10), (11):

$$\int\limits_a^b f(x)dx = \int\limits_{-\infty}^{+\infty} f\left(\frac{a+be^t}{1+e^t}\right)\frac{(b-a)dt}{(e^{-t/2}+e^{t/2})^2} \approx$$

$$\approx h_{sinc} \sum_{i=-K}^{K} f\left(\frac{a+be^{ih_{sinc}}}{1+e^{ih_{sinc}}}\right)\frac{b-a}{(e^{-ih_{sinc}/2}+e^{ih_{sinc}/2})^2},$$

$$\int\limits_a^{z_j} f(x)dx \approx h_{sinc} \sum_{i=-K}^{K} \delta_{j-i}^{(-1)} f\left(\frac{a+be^{ih_{sinc}}}{1+e^{ih_{sinc}}}\right)\frac{b-a}{(e^{-ih_{sinc}/2}+e^{ih_{sinc}/2})^2}$$

where $\delta_i^{(-1)} = \frac{1}{2} + \int\limits_0^i \frac{\sin(\pi t)}{\pi t}dt$, $i = -2K \ldots 2K$, $h_{sinc} = \sqrt{\frac{2\pi}{K}}$.

[1] Here $[\cdot]$ denotes the integer part of a real number.

[2] If the code called upon by SciPy and NumPy is compiled for the `x86_64` architecture. For reasons to do the GCC compiler the same `numpy.longdouble` type we use results in 80-bit precision on 32-bit processors.

Algorithm 1: Main program

Data: n — the number of the eigenvalue we want to find, m — the order of the FD-method (the number of steps taken), K, h_{sinc}, z_i, μ_i, $\delta_i^{(-1)}$

Result: $\overset{m}{\lambda}_n$, $\overset{m}{\eta}_n$, $\overset{m}{u}_n(x)$, $\frac{d\overset{m}{u}_n}{dx}(x)$, $\left\{\left\|u_n^{(i)}(x)\right\|\right\}_{i=0}^{m}$

begin

- // We initialize L as a one-dimensional array of $2K+1$ zeros and F, U and DU as two-dimensional arrays of $2K+1$ by $2K+1$ zeros.
- $L := zeros(-K \ldots K)$;
- $F, U, DU := zeros(-K \ldots K, -K \ldots K)$;
- $L[0] = n(n+1)$;
- **for** $i := -K \ldots K$ **do**
  - $U[0][i] = P_n(x)$;
  - $DU[0][i] = dP_n(x)$;
- **end**
- **for** $d := 1, 2 \ldots m$ **do**
  - // Compute the correction for the eigenvalue
  - $L[d] := A^{-2}$IntAB$(U[0], U[d-1], q)$;
  - // Compute F
  - **for** $i := -K \ldots K$ **do**
    - $F[d][i] := U[d-1][i]\, q(z_i)$;
    - **for** $j := 0 \ldots d-1$ **do**
      - $F[d][i] := F[d][i] - L[d-j]\, U[j][i]$;
    - **end**
  - **end**
  - // Compute the correction for the eigenfunction
  - **for** $i := -K \ldots K$ **do**
    - $U[d][i] := Q_n(z_i)$IntAZ$(i; F[d], P_n) - P_n(z_i)$IntAZ$(i; F[d], Q_n)$;
    - $DU[d][i] :=$ $dQ_n(z_i)$IntAZ$(i; F[d], P_n) - dP_n(z_i)$IntAZ$(i; F[d], Q_n)$;
  - **end**
  - // Orthogonality
  - $I = A^{-2}$IntAB$(U[d], U[0])$;
  - **for** $i := -K \ldots K$ **do**
    - $U[d][i] := U[d][i] - I\, U[0][i]$;
    - $DU[d][i] := DU[d][i] - I\, DU[0][i]$;
  - **end**
  - // Compute the residual
  - CompRes;
- **end**
- $\overset{m}{\lambda}_n := \sum_{i=0}^{m} L[i]$;

end

We use the following auxiliary notation:

$$z_i = \frac{a + be^{h_{sinc}i}}{1 + e^{h_{sinc}i}},\ \mu_i = \frac{b-a}{(e^{-ih_{sinc}/2} + e^{ih_{sinc}/2})^2}$$

and

$$A^{-1} = A^{-1}(n) = \sqrt{2/(2n+1)}.$$

In order to measure how close an obtained approximation is to the exact solution we use the *residual* functional

$$\overset{m}{\eta}_n = \left[\int\limits_{-1}^{1} \left[(1-x^2)\frac{d\overset{m}{u}_n(x)}{dx} + \int\limits_{-1}^{x}\left(\overset{m}{\lambda}_n - q(\xi)\right)\overset{m}{u}_n(\xi)d\xi\right]^2 dx\right]^{\frac{1}{2}}.$$

For the sake of simplicity the details related to the subdivision of the interval (a, b) into subintervals are omitted.

The values of $\delta_i^{(-1)}$ are precomputed. By "$F[i]$" we mean the values $F[i][-K], F[i][-K+1], \ldots, F[i][K]$ taken as a one-dimensional array. The function $\mathtt{IntAB}(f_1, \ldots, f_n)$ calculates the definite integral of the product of its arguments over (a, b); $\mathtt{IntAZ}(i; f_1, \ldots, f_n)$ does the same over (a, z_i).

Conclusions

The article [1] lays out the structure of and provides a theoretical justification for the FD-method as applied to solving the Sturm-Liouville problem (1), (2). In Theorem 3 convergence is proven for the case when $q(x)$ satisfies condition (3) and estimates for the convergence rate are given explicitly.

The algorithm is implemented in software as a library (a Python module). The implementation can be integrated into larger systems or used as is in applied sciences. It doesn't require the user to understand much of its internal workings. The source code for the library along with example Python code that uses it can be obtained from GitHub at `https://github.com/imathsoft/legendrefdnum`.

References

[1] V. Makarov, D. Dragunov, and D. Bohdan, "Exponentially convergent numerical-analytical method for solving eigenvalue problems for singular differential operators," 2013. Preprint: arXiv:1309.5795. http://arxiv.org/abs/1309.5795.

[2] V. L. Makarov and Y. V. Klimenko, "Application of the functional-discrete method to the solution of the Sturm-Liouville problem with coefficients of a special form," *Ukraïn. Mat. Zh.*, vol. 59, no. 8, pp. 1140–1147, 2007.

[3] V. L. Makarov, D. V. Dragunov, and Y. V. Klimenko, "The fd-method for solving sturm-liouville problems with special singular differential operator," *Math. Comp.*, vol. 82, no. 282, pp. 953–973, 2013.

[4] S. Y. Slavyanov and W. Lay, *Special functions*. Oxford Mathematical Monographs, Oxford: Oxford University Press, 2000. A unified theory based on singularities, With a foreword by Alfred Seeger, Oxford Science Publications.

[5] V. Makarov and N. Rossokhata, "A review of functional-discrete technique for eigenvalue problems," *J. Numer. Appl. Math.*, no. 97, pp. 79–102, 2009. Available at http://www.nbuv.gov.ua/portal/natural/jopm/2009_1/index.htm.

Authors

Volodymyr L. Makarov — Professor, academician of the NAS of Ukraine, Department of numerical mathematics, Institute of Mathematics of the NAS of Ukraine, Kyiv, Ukraine; E-mail: *makarov@imath.kiev.ua*

Denys V. Dragunov — Candidate of physical and mathematical sciences, developer at 3Shape Ukraine, Kyiv, Ukraine; E-mail: *dragunovdenis@gmail.com*

Danyil V. Bohdan — the 2nd year postgraduate student, Department of numerical mathematics, Institute of Mathematics of the NAS of Ukraine, Kyiv, Ukraine; E-mail: *danyil.bohdan@gmail.com*