Available online at www.sciencedirect.com
ScienceDirect
Journal of Computational and Applied Mathematics 208 (2007) 404–424
JOURNAL OF COMPUTATIONAL AND APPLIED MATHEMATICS
www.elsevier.com/locate/cam

Numerical methods for the eigenvalue determination of second-order ordinary differential equations

Hideaki Ishikawa

Semiconductor Leading Edge Technologies, Inc. (Selete), 16-1 Onogawa, Tsukuba-shi, Ibaraki-ken 305-8569, Japan

Received 20 April 2005; received in revised form 2 July 2006

Abstract

An accurate method for the numerical solution of the eigenvalue problem of second-order ordinary differential equation using the shooting method is presented. The method has three steps. Firstly initial values for the eigenvalue and eigenfunction at both ends are obtained by using the discretized matrix eigenvalue method. Secondly the initial-value problem is solved using new, highly accurate formulas of the linear multistep method. Thirdly the eigenvalue is properly corrected at the matching point. The efficiency of the proposed methods is demonstrated by their applications to bound states for the one-dimensional harmonic oscillator, anharmonic oscillators, the Morse potential, and the modified Pöschl–Teller potential in quantum mechanics.

MSC: 65L05; 65L06; 65L07; 65L15; 81-08

Keywords: Numerical eigenvalue determination; Linear multistep method; Shooting

1. Introduction

Eigenvalue problem in ordinary differential equations is one of the basic approaches in the field of boundary-value problems in mathematics and mathematical physics and its important field of application is quantum mechanics. Solutions of eigenvalue problems in quantum mechanics are obtained analytically with special or other mathematical functions and approximately by using the perturbation, variation or Wentzel–Kramers–Brillouin (WKB) method, but their applicable range is rather restricted for practical problems. In order to overcome these limitations, numerical methods of solution have been developed from early days of quantum mechanics.

One of the methods for the solution of eigenvalue and eigenfunction is the classical matching or shooting method. Though pioneering research using the Numerov or Cowell method for the initial-value problem has provided fruitful insight into atomic structure calculations [23,17], errors in numerical calculations are usually larger than $1.0\mathrm{D}-8$ ($=1.0\times10^{-8}$) for the eigenvalues so that further improvements in accuracy are necessary. Another method of solution in two-point boundary-value problem is the discretized matrix (DM) eigenvalue method [39,13,20,27]. The merits of this method of solution are that the eigenvalues and eigenfunctions are accurate up to high quantum number and highly reliable because *all* eigenvalues are found without missing any. On the downside, the method requires long computing time, especially for tridiagonalization of the matrix.

E-mail address: ishikawa.hideaki@selete.co.jp.

0377-0427/$ - see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.cam.2006.10.035

The main theme of the present paper is to provide an accurate and fast method for solving eigenvalue problems while retaining good properties of the DM eigenvalue method. A powerful candidate is the shooting method, which requires three tasks for implementing it: (i) to get initial-value or guess of the eigenvalue and eigenfunction adequately, (ii) to solve the initial-value problem accurately, and (iii) to suitably correct the eigenvalue at the matching point.

The first task is settled by devising usage of the DM eigenvalue method described later. The second, most important, task needs detailed considerations. A large number of approaches to this task have been reported in books on numerical analysis [24,51], software for numerical calculations [38,40], numerical methods for the ordinary differential [25,32,22], and references cited therein. Numerical methods for the solution of ordinary differential equations are classified into two categories: one-step methods represented by the Runge–Kutta method and multistep methods. In practical problems, since physical quantities for the solutions of ordinary differential equations are given only at discretized points in many cases, the multistep methods that use quantities only at the discretized points are more suitable than the one-step methods which also use quantities at intermediate points between the discretized points. Although a large number of publications on the linear multistep method (LMM) show formulations and typical examples of calculations, the ultimate performance of this computation method has not yet been fully investigated. In particular, classical works on the optimal operator for the special second-order ordinary differential equation, in which the first-order derivative is absent, for accuracy and stability in [9,10,25] are important and their fundamental concept and formulas for a small number of steps have been explained explicitly. However, research on extending their concept to, and deriving formulas for, a large number of steps and evaluating their performance can be enhanced. For example, use of the formula with 4-step number [31,52] stays within the framework of Henrici [25] in quantum mechanical applications. Moreover, a special class of the classical LMM for reducing weak instability, namely, a symmetric multistep method formulated using the test equation for a harmonic oscillator in classical mechanics, has been developed for calculating long-term planetary motion in celestial dynamics [32,33,46,45,18,19,21], but it is necessary to clarify whether the symmetric multistep method is also useful in quantum mechanics. Recently, higher-order exponentially fitted methods for numerical integration of the Schrödinger equation have been extensively developed, see, for example, [50,54], however, further improvements in accuracy are necessary. Thus, whether the LMM provides highly accurate numerical solutions in quantum mechanics, is still an open question. It is a challenging and non-trivial problem to derive formulas with a large step-number explicitly and to get highly accurate solutions to confirm efficacy of the method.

The third task of suitable correction of the eigenvalue at the matching point is carried out based on the correction formula [23,24,47,11] and its generalization. Careful calculations such as suitable choice of matching point, numerical differentiation and integration are necessary for highly accurate eigenvalues and eigenfunctions.

The method mentioned above has generality and wide applicability. We can solve eigenvalue problems for the case the potential is not only given in an analytic form but also in a form of numerical table. Moreover, we can apply our method to problems that the potential should be calculated self-consistently by using the eigenfunctions for regular meshes with arithmetic or geometric progression, which often meets in physical problems such as atomic structure calculations. The applicability of the present method is much wider than that of recent methods adopted in the codes SLEIGN [3], SLEIGN2 [2,1], SLEDGE [41], SL02F [44,37,42], SLTSTPAK [43], SLCPM12 [28–30], and MATSLISE [35,36]; these methods are applied only to the case the potential is given analytically, and the meshes are, in general, not regular and obtained after all calculations are ended. In this paper we show results of the eigenvalue problems in one-dimension in which the potential is regular in the whole interval $(-\infty, \infty)$. A method and results of the eigenvalue problems for the central force-field potential will be published elsewhere, since it is necessary to treat singularities at both ends of the semi-infinite whole interval.

Organization of the paper is as follows. In Section 2, we present the method of calculations. Section 3 is devoted to results and discussion. We also show comparison with other recent methods for the solution of the Schrödinger equation.

2. Numerical methods of calculation

2.1. Initial value or guess of eigenvalue and eigenfunction

We can use the solutions from the DM eigenvalue method for the initial value or guess of eigenvalue and eigenfunction for the shooting method. Since the required accuracy for the initial value or guess of eigenvalue and eigenfunction is not very high, the second-order derivative is discretized by using the lowest, three-point formula, so that the matrix is in

tridiagonal form from the beginning. This enables us to avoid time-consuming tridiagonalization using the Householder transformation [38,40]. We use the standard method for the algebraic (i.e., matrix) eigenvalue problem [38]; that is, the bisection method for eigenvalue computation and the inverse iteration method for eigenvector computation. The whole interval of computation is taken wide enough in order to avoid deterioration in the accuracy of eigenvalues and eigenfunctions [27]. Though the absolute values of relative errors of the eigenvalues are usually of order larger than $1.0\mathrm{D}-5$, the eigenvalues are accurate enough for initial guesses. Since we can suitably choose end-positions of the whole interval for computation so that the absolute values of absolute errors in the eigenfunctions near both ends are of order around $1.0\mathrm{D}-15$ [27], we can get reliable initial guesses for eigenfunctions. With these initial eigenfunctions, we can compute diagonal matrix elements (MEs) of the Hamiltonian, which are, in general, more accurate than the eigenvalues [27], and also use them as initial guesses for the eigenvalues. The whole interval for the DM eigenvalue method is also used for the shooting method described later.

2.2. Numerical method for initial-value problems in ordinary differential equations

2.2.1. Boundary-value problem written as initial-value problems

The Schrödinger equation for a particle in the one-dimensional potential with suitable boundary conditions at both ends is written in the form

$$(\mathrm{d}/\mathrm{d}x)^2 y = [V(x) - \lambda]y \tag{1}$$

for the initial value problems starting from one end-point. Here $y = y(x)$ is the wave function of the coordinate x, $V(x)$ is the potential energy, and λ is the energy or the parameter, where these quantities have been reduced to dimensionless form by suitably choosing the units of the energy and length. Replacing the right-hand side with a continuous and smooth function $f(x, y, \lambda)$ of x, y, and λ that does not contain first-order derivative of y, we have a generalized differential equation

$$(\mathrm{d}/\mathrm{d}x)^2 y = f(x, y, \lambda). \tag{2}$$

Here, absence of the first derivative causes no loss of generality, because in the case of second-order linear ordinary differential equations it is always possible to rewrite the differential equations into a form with the first derivative absent. In the case of initial-value problem we treat the differential equation in the form such that the parameter λ is contained in the function $f(x, y)$

$$(\mathrm{d}/\mathrm{d}x)^2 y = f(x, y). \tag{3}$$

2.2.2. Numerical solution using linear multistep method

The initial-value problem of the second-order ordinary differential equation (3) is solved by using the LMM that evaluates functions only at the discretized points. A general formula for approximating the differential equation (3) in the form of a linear difference equation is given by [25,32,22,10]

$$\begin{aligned}&\alpha_k y_{n+k} + \alpha_{k-1} y_{n+k-1} + \alpha_{k-2} y_{n+k-2} + \cdots + \alpha_0 y_n \\ &\quad = h^2(\beta_k f_{n+k} + \beta_{k-1} f_{n+k-1} + \beta_{k-2} f_{n+k-2} + \cdots + \beta_0 f_n), \quad n = 0, 1, 2, \ldots,\end{aligned} \tag{4}$$

where y_m, $m = 0, 1, 2, \ldots$, is a solution of the difference equation at the equidistant discretized point $x = x_m$ and $f_m = f(x_m, y_m)$; k is an integer and coefficients α_μ and β_μ, $\mu = 0, \ldots, k$, are constants independent of n. In this formulation, y_{n+k} is evaluated by using already known k-point values (k-step method). h is the step-size between discretized points.

We obtained implicit formulas for the linear difference equation (4) explicitly based on the method of constructing the optimal operators, which is an approach based on Dahlquist [9,10] and developed in [25].[1] Coefficients α_μ and β_μ of Eq. (4) are symmetric for the optimal operator such that

$$\alpha_{k-\mu} = \alpha_\mu, \quad \beta_{k-\mu} = \beta_\mu, \ \ \mu = 0, 1, 2, \ldots, k. \tag{5}$$

[1] In Ref. [25, Table 6.8] the minus sign for the coefficient l_8 of t^8 is necessary.

Table 1
Coefficient α_μ and error constant C_{k_i} for the optimal operator in the linear multistep method, where k is the step-number, i is the sequence number and $\mu = 0, 1, \ldots, k$

Case	k	i	α_0 α_k	α_1 α_{k-1}	α_2 α_{k-2}	α_3 α_{k-3}	α_4 α_{k-4}	α_5 α_{k-5}	$-C_{k\_i}$	
(a)	2	1	1	−2					$1/240 = 4.167 \times 10^{-3}$	(A)
(b)	4	1	1	0	−2				$1/1890 = 5.291 \times 10^{-4}$	(A)
(c)		2	1	−1	0				$53/60480 = 8.763 \times 10^{-4}$	(A)
(d)		3	1	−2	2				$19/12096 = 1.571 \times 10^{-3}$	(A)
(e)	6	1	1	0	−1	0			$7/32400 = 2.160 \times 10^{-4}$	(A)
(f)		2	1	1	−1	−2			$29/226800 = 1.279 \times 10^{-4}$	
(g)		3	1	−1	−1	2			$109/226800 = 4.806 \times 10^{-4}$	
(h)		4	1	0	0	−2			$9/44800 = 2.009 \times 10^{-4}$	
(i)		5	1	−1	1	−2			$1259/3628800 = 3.469 \times 10^{-4}$	
(j)		6	1	−2	2	−2			$407/518400 = 7.851 \times 10^{-4}$	(B)
(k)	8	1	1	0	0	0	−2		$2/22275 = 8.979 \times 10^{-5}$	
(l)		2	1	0	−1	0	0		$881/7484400 = 1.177 \times 10^{-4}$	
(m)		3	1	1	0	−1	−2		$23/427680 = 5.378 \times 10^{-5}$	
(n)		4	1	−1	0	1	−2		$47/237600 = 1.978 \times 10^{-4}$	
(o)		5	1	0	1	−2	0		$163/1971200 = 8.269 \times 10^{-5}$	
(p)		6	1	−1	1	−2	2		$361/1971200 = 1.831 \times 10^{-4}$	
(q)		7	1	−2	3	−4	4		$150401/479001600 = 3.140 \times 10^{-4}$	
(r)		8	4	−5	4	−8	10		$493/1971200 = 2.501 \times 10^{-4}$	
(s)	10	1	1	0	−1	0	0	0	$3011003/40864824000 = 7.368 \times 10^{-5}$	
(t)		2	1	0	0	0	−1	0	$2036393/40864824000 = 4.983 \times 10^{-5}$	
(u)		3	1	1	1	0	−2	−2	$29299/1277025750 = 2.294 \times 10^{-5}$	
(v)		4	1	−1	1	0	−2	2	$105739/1277025750 = 8.280 \times 10^{-5}$	
(w)		5	1	−1	2	−3	3	−4	$74359/978432000 = 7.600 \times 10^{-5}$	
(x)		6	1	−2	4	−6	7	−8	$335626517/2615348736000 = 1.283 \times 10^{-4}$	
(y)		7	1	−2	3	−4	5	−6	$1803479/10762752000 = 1.675 \times 10^{-4}$	
(z)		8	4	−4	9	−12	12	−18	$729679/10762752000 = 6.780 \times 10^{-5}$	

α_μ is symmetric, i.e., $\alpha_{k-\mu} = \alpha_\mu$.
(A) Henrici [25].
(B) Lambert and Watson [33].

The truncation error for the optimal operator is $C_k t^{k+2}$, where t is related to the step-size by $h = 2\ln\{t/2 + [1 + (t/2)^2]^{1/2}\}$ or $t = [2(\cosh h - 1)]^{1/2}$, with the error constant C_k whose sign is negative. The coefficients α_μ, β_μ, and C_{k_i} up to $k = 10$ are listed[2] in Tables 1 and 2 for the roots listed in Table 3, of the first characteristic polynomial ρ (ζ) corresponding to the left-hand side of Eq. (4) [25]. The values of α_μ, β_μ, and C_{k_i} shown explicitly here, but not available in the literature, should be useful for practical purposes. Since we obtained numerous new formulas in addition to the known formulas, we classify them, without using authors' names, as a formula k_i, where k is the step-number and i is the sequence number.

2.3. Shooting method

Next, a method of correcting the eigenvalue at the matching point is described below. Let us consider a differential equation that is a slight generalization of Eq. (1)

$$(\mathrm{d}/\mathrm{d}x)^2 y = [V(x) - \lambda W(x)]y, \tag{6}$$

where $W(x)$ is a positive function in the whole interval. Consider the solution y' of the differential equation for the parameter $\lambda' = \lambda + \Delta\lambda$ with a deviation $\Delta\lambda$

$$(\mathrm{d}/\mathrm{d}x)^2 y' = [V(x) - \lambda' W(x)]y'. \tag{7}$$

[2] We also obtained implicit formulas up to $k = 14$ that will be useful for arithmetic more precise than double-precision arithmetic.

Table 2
Coefficient β_μ for the optimal operator in the linear multistep method, where k is the step-number, i is the sequence number and $\mu = 0, 1, \ldots, k$

Case	k	i	Denominator	β_0 β_k	β_1 β_{k-1}	β_2 β_{k-2}	β_3 β_{k-3}	β_4 β_{k-4}	β_5 β_{k-5}	
(a)	2	1	12	1	10					(A)
(b)	4	1	15	1	16	26				
(c)		2	240	17	232	222				
(d)		3	120	9	104	14				(B)
(e)	6	1	945	59	1032	1641	2096			
(f)		2	315	19	369	873	1258			
(g)		3	945	61	957	663	418			
(h)		4	2240/3	47	810	1377	2252			
(i)		5	10080	661	10062	9243	20548			
(j)		6	20160	1375	18378	−207	21388			(C)
(k)	8	1	14175/4	209	3968	6032	12416	11450		
(l)		2	18900	1111	21232	30568	46864	27250		
(m)		3	9450	541	11302	25948	50434	50350		
(n)		4	28350	1721	29582	18668	47354	32150		
(o)		5	22400/3	443	8316	13484	32132	25650		
(p)		6	44800/3	913	15476	11524	35212	8150		
(q)		7	604800	38401	574132	−44732	1352524	−211850		
(r)		8	44800/3	3679	60748	30652	111796	−10550		
(s)	10	1	311850	17405	360132	457229	909296	187718	1126040	
(t)		2	623700	34901	718276	966117	2458608	1537974	3537048	
(u)		3	155925/4	2131	47627	104739	266736	293598	441438	
(v)		4	155925/4	2243	41909	22483	120592	43966	161314	
(w)		5	985600/3	19003	351302	224879	1245096	205398	1822244	
(x)		6	39916800	2384065	39396722	−7185891	145207416	−48518622	216434220	
(y)		7	1971200/3	39089	651602	−182451	1853176	−911102	3012972	
(z)		8	1971200/3	152129	2808178	1854621	10627704	2943522	16450092	

β_μ is symmetric, i.e., $\beta_{k-\mu} = \beta_\mu$.
(A)Henrici [25].
(B) Ixaru and Rizea [31] and Vanden Berghe et al. [52].
(C) Lambert and Watson [33].

Integrating both sides of [$y' \times$ Eq. (6)–$y \times$ Eq. (7)] from both ends to the matching point and setting the summations of $\Delta[(\mathrm{d}y/\mathrm{d}x)/y]$ up to first order at both ends of the matching point to be equal, we obtain the equation for $\Delta\lambda$

$$\Delta\lambda = \frac{[(\mathrm{d}y/\mathrm{d}x)/y]|_{x=x_{m-}} - [(\mathrm{d}y/\mathrm{d}x)/y]|_{x=x_{m+}}}{\int_{-\infty}^{x_{m-}} \mathrm{d}x W y y'/[y(x=x_{m-})]^2 + \int_{x_{m+}}^{\infty} \mathrm{d}x W y y'/[y(x=x_{m+})]^2}. \tag{8}$$

Here x_{m-} denotes the value at x_m calculated by using the solution of differential equation that starts from $-\infty$, while x_{m+} is the value at x_m calculated by using the solution that starts from $+\infty$. If the product yy' in the integrand in the denominator is approximated to y^2 by neglecting $y\Delta y$ and W is replaced by 1, Eq. (8) reduces to Ridley's formula [23,24,47,11]

$$\Delta\lambda = \frac{[(\mathrm{d}y/\mathrm{d}x)/y]|_{x=x_{m-}} - [(\mathrm{d}y/\mathrm{d}x)/y]|_{x=x_{m+}}}{\int_{-\infty}^{x_{m-}} \mathrm{d}x [y]^2/[y(x=x_{m-})]^2 + \int_{x_{m+}}^{\infty} \mathrm{d}x [y]^2 [y(x=x_{m+})]^2}. \tag{9}$$

To be useful, Eqs. (8) and (9) require the following: the matching point should not be placed near the zero point or node of eigenfunction in order to avoid division by zero and consequent deterioration of accuracy. The matching point also should not be placed near the extremum of eigenfunction in order to avoid loss of significant digits that occurs from the difference of the first derivatives (normalized by the eigenfunction) in the numerator. Thus the matching point should be placed intermediate between the zero and extremum points. Among the candidates for such points, the point near the center of the whole interval is best for the matching point, because the errors accumulate in the course of solving the initial-value problem so that the accumulated errors starting from the left limit and those from the right limit are

Table 3
Roots of the first characteristic polynomial $\rho(\zeta)$ satisfying the stability condition, where k is the step-number, i is the sequence number

Case	k	i	Roots	
(a)	2	1	1∗	(A)
(b)	4	1	1∗, −1∗	(A)
(c)		2	1∗, exp[i(2π/3)], exp[−i(2π/3)]	(A)
(d)		3	1∗, i, −i	(A)
(e)	6	1	1∗, −1∗, i, −i	(A)
(f)		2	1∗, −1∗, exp[i(2π/3)], exp[−i(2π/3)]	
(g)		3	1∗, −1∗, exp[i(π/3)], exp[−i(π/3)]	
(h)		4	1∗, exp[i(2π/3)]∗, exp[−i(2π/3)]∗	(B)
(i)		5	1∗, exp[i(2π/3)], exp[−i(2π/3)], i, −i	
(j)		6	1∗, exp[i(2π/3)], exp[−i(2π/3)], exp[i(π/3)], exp[−i(π/3)]	(C)
(k)	8	1	1∗, −1∗, i∗, −i∗	(D)
(l)		2	1∗, −1∗, exp[i(2π/3)], exp[−i(2π/3)], exp[i(π/3)], exp[−i(π/3)]	
(m)		3	1∗, −1∗, exp[i(2π/3)], exp[−i(2π/3)], i, −i	
(n)		4	1∗, −1∗, i, −i, exp[i(π/3)], exp[−i(π/3)]	
(o)		5	1∗, exp[i(2π/3)]∗, exp[−i(2π/3)]∗, i, −i	
(p)		6	1∗, exp[i(2π/3)]∗, exp[−i(2π/3)]∗, exp[i(π/3)], exp[−i(π/3)]	
(q)		7	1∗, exp[i(2π/3)], exp[−i(2π/3)], i, −i, exp[i(π/3)], exp[−i(π/3)]	
(r)		8	1∗, exp[i(2π/3)]∗, exp[−i(2π/3)]∗, exp{i[Arc cos(5/8)]}, exp{−i[Arc cos(5/8)]}	(B)
(s)	10	1	1∗, −1∗, exp[i(3π/4)], exp[−i(3π/4)], i, −i, exp[i(π/4)], exp[−i(π/4)]	
(t)		2	1∗, −1∗, exp[i(2π/3)], exp[−i(2π/3)], i, −i, exp[i(π/3)], exp[−i(π/3)]	
(u)		3	1∗, −1∗, exp[i(2π/3)], exp[−i(2π/3)], i∗, −i∗	
(v)		4	1∗, −1∗, i∗, −i∗, exp[i(π/3)], exp[−i(π/3)]	
(w)		5	1∗, exp[i(2π/3)]∗, exp[−i(2π/3)]∗, i, −i, exp[i(π/3)], exp[−i(π/3)]	
(x)		6	1∗, exp[i(2π/3)], exp[−i(2π/3)], i∗, −i∗, exp[i(π/3)], exp[−i(π/3)]	
(y)		7	1∗, exp[i(2π/3)]∗, exp[−i(2π/3)]∗, exp[i(π/3)]∗, exp[−i(π/3)]∗	
(z)		8	1∗, exp[i(2π/3)]∗, exp[−i(2π/3)]∗, exp{i[Arc cos(1/4)]}∗, exp{−i[Arc cos(1/4)]}∗	(B)

The asterisk means a double root, while no asterisk means a simple root
(A) Henrici [25].
(B) Fukushima [18].
(C) Lambert and Watson [33].
(D) Hairer and Lubich [21].

both small and well-balanced near the center of the whole interval. A classical turning point that is the minimum or maximum of $V(x) - \lambda_v = 0$ in Eq. (1) [34,16] has often been used for the matching point because it is necessary to count node numbers between the minimum and maximum of the classical turning point in order to check whether the eigenvalue lies in the suitable range or not, due to uncertain initial value for the eigenvalue. In our method, however, the initial values for eigenvalue and eigenfunction are obtained in the correct range by using the DM eigenvalue method. Since the node count is unnecessary, we can place the matching point anywhere so that we have the advantage of placing it near the center of the whole interval.

The numerical derivative in Eqs. (8) and (9) is computed by using the highly accurate numerical derivatives in [27]. The numerical integration in Eqs. (8) and (9) is carried out by using the highly accurate central-difference integration formula. In [27] we presented formulas for up to $n = 8$, which are further developed and used for highly accurate integration for a function $f(x)$ with $n = 10$ as given in Eq. (10)

$$\int_{x_{i-1}}^{x_{i+1}} f(x)\,\mathrm{d}x = (h/7484400)[263 f_{i-5} - 4148 f_{i-4} + 33879 f_{i-3} - 216624 f_{i-2} + 3116178 f_{i-1} + 9109704 f_i + 3116178 f_{i+1} - 216624 f_{i+2} + 33879 f_{i+3} - 4148 f_{i+4} + 263 f_{i+5}] + \mathrm{O}(h^{13}). \tag{10}$$

The integration formula is also used for evaluating the diagonal Hamiltonian MEs λ_v for the quantum number v

$$\lambda_v = \langle v| - (\mathrm{d}/\mathrm{d}x)^2 + V(x)|v\rangle / \langle v|v\rangle, \tag{11}$$

where the MEs of the operator $A = A(x, \mathrm{d}/\mathrm{d}x, (\mathrm{d}/\mathrm{d}x)^2)$ is

$$\langle v|A|v'\rangle = \int_{-\infty}^{\infty} \mathrm{d}x y_v(x) A y_{v'}(x). \tag{12}$$

With the formulas developed thus far, calculations for correcting the eigenvalue $\lambda' = \lambda + \Delta\lambda$ are carried out until the absolute value of $\Delta\lambda$ falls below a specified error limit.

2.4. Comparison with other recent methods

We compare our classical method with other recent methods for the solution of the Schrödinger equation. First category is the SLEIGN [3] and SLEIGN2 [2,1], SLEDGE [41], SL02F [44,37,42], and SLTSTPAK [43] codes. In the latest code SLEIGN2 [1] the Schrödinger equation is treated as a special form of the Strum–Liouville (SL) equation. The second-order SL equation is written into two first-order nonlinear differential equations for phase θ and amplitude ρ by using the Prüfer transformation. The eigenvalue is evaluated by using the shooting method for the nonlinear equations for θ and its derivative $\partial\theta/\partial\lambda$. The initial-value problem for θ is solved by using the Runge–Kutta–Fehlberg method with step-size adjustment determined by extrapolation procedure of error estimation [49]. The eigenfunction is evaluated by solving the nonlinear equation for ρ and by applying the Prüfer transformation. The SLEIGN2 code is not satisfactory for practical purposes in the following: the coefficient functions of the SL equation should be given analytically. The nonlinear differential equations for θ and ρ are more computationally complicated than the Schrödinger equation, because they contain the functions of $\cos\theta$ and $\sin\theta$. Computing time for the initial-value problem using the Runge–Kutta–Fehlberg method with step-size adjustment is longer than that using the LMM. In short, directly solving the Schrödinger equation by using the LMM is much faster than the method adopted in the SLEIGN2 code. The eigenvalues converge slowly and sometimes to erroneous ones. The normalization integral for the eigenfunction is not accurate enough. Moreover, tools for evaluating MEs are not provided. In the SLEDGE code [41] the coefficient functions in the SL equation are approximated by piecewise constants and the resulting equation is solved analytically by using trigonometric and exponential functions. Since errors in the eigenvalue is of order $\mathrm{O}(h^2)$, where h is the maximum mesh size, for the SL normal form and $\mathrm{O}(h)$ for the general SL form, h should be small enough for getting highly accurate eigenvalues. The mesh points are added by repeated bisection at the midpoint of the interval, thus determined after all calculations are ended. In the SL02F code [44,37,42] the scaled Prüfer transformation is applied to the SL equation, the coefficient functions are approximated by piecewise constants, then the resulting nonlinear differential equations for θ and ρ are solved analytically in terms of trigonometric and hyperbolic functions, in order to avoid stiffness of the differential equations. Computing time with the SL02F code is of the same order or one order shorter than that with the SLEIGN2 code [37,28–30]. The codes SLEDGE [41], SL02F [44,37,42], SLEIGN [3] and SLEIGN2 [2] are all included in the SLTSTPAK code [43]. Later, we compare our numerical results for the potentials in the analytic form with those obtained by using the latest code SLEIGN2 [1], where single-precision arithmetic is replaced to double-precision one, that is a representative of the first category.

The second category is the Constant reference potential Perturbation Method (CPM). In the CPM the initial-value problem for the Schrödinger equation is solved by using the so-called propagation matrix algorithm. The whole interval is divided into small intervals in which the potential is decomposed to a constant and a deviation from the constant. The constant potential and the deviation are taken as reference and perturbation potentials, respectively. The propagators, the MEs of the propagation matrix, are constructed by the perturbation series truncated at the maximum order Q. For practical purpose, the potential is approximated by a truncated series of shifted Legendre polynomials with maximum degree N. The CPM of order {12,10} [28–30], {14,12}, {16,14}, and {18,16} [35,36] are reported. The number of terms in the perturbation series increases very rapidly for higher-order perturbations. The derivation of formulas by hand calculation is a formidable task thus symbolic computation using, for example, MATHEMATICA, MAPLE, etc., is indispensable. The resulting formulas are very complicated thus hard to implement, and their computational cost is high. In the CPM it is necessary to give potential analytically because the coefficients of the expansion in terms of the shifted Legendre polynomials are calculated by using the Gauss–Legendre integration formula. This restricts applicability of the method to the case the potential is given only in the form of numerical table, which often meets in physical problems such as atomic structure calculations. The users must give end-positions of the whole interval, but there is no guidance how to set them. If the end-positions are not given appropriately, calculations often hang up or give incorrect eigenvalues without hints for improvement. How to choose the

end-positions in our method shown in Section 2.1 can also be used for such cases. Tools for calculating MEs in the CPM are not provided. Later we compare our numerical results with those obtained by using the code SLCPM12 [30] because we do not have MATLAB facility with Symbolic Math Toolbox necessary for MATSLISE [35,36]. Our method is much simpler and easier to implement than the CPM, but compares favorably with CPM on accuracy and computing time.

We use a PC with a 266 MHz processor and 128 MB memory, and the compiler DIGITAL Visual Fortran Version 5.0, for computation.

3. Results and discussion

3.1. Harmonic oscillator

As a typical application of the new methods of calculation, we take the harmonic oscillator, because the analytical properties of the solution are well known [39,34,16]. The Schrödinger equation for the potential $V(x) = x^2$ in the dimensionless form[3] is written as

$$(\mathrm{d}/\mathrm{d}x)^2 y = (x^2 - \lambda)y. \tag{13}$$

The solution with the quantum number v is given in terms of the Hermite polynomial $H_v(x)$

$$\lambda_v = 2v + 1, \quad v = 0, 1, 2, \ldots, \tag{14}$$

$$y_v(x) = (1/\pi^{1/2} 2^v v!)^{1/2} \exp[-(1/2)x^2] H_v(x). \tag{15}$$

Fig. 1 shows the magnitude of absolute (not relative) errors in the numerical solutions for the initial-value problem using the LMM with exact eigenvalue λ_v and exact eigenfunction y_v for the initial value, for the typical cases of each step-number. The quantum state is taken from $v = 0$ to 7. The ordinary differential equation was solved starting from $x = 10$, where the absolute value of the wave function is sufficiently small, to $x = 0$ with the step-size $h = \frac{1}{64}$. In the formula 2_1, the absolute error of the numerical solution of the LMM is less than 1.0D − 5 as shown in Fig. 1(a). For $v > 0$ the absolute error is small with downward cusps for small x because the solution passes near the zeros (nodes) of the wave function. For higher v values, the absolute errors at large x become larger because the solutions themselves contain polynomials thus widening the range with finite magnitude of the wave function for high v. The absolute error for the formula 4_1 is the largest for $v = 0$ and $x = 0$ with the magnitude 2.4D − 8 and the error difference is small among the formulas with the same step-number $k = 4$. In the 6-step formulas represented by the formula 6_2, the absolute errors are the largest for $v = 0$ and $x = 0$, and the largest errors are of order 1.0D − 10. In the formulas of 8-step, the largest absolute errors are of order 1.0D − 12. In the 10-step formulas, the absolute errors are often zero (i.e., undetectable) in the double precision arithmetic, which is shown by vertical lines in Fig. 1(v), and the largest errors are of order 1.0D − 14 for the formulas 10_4 and 10_5. We can conclude that increasing the step-number of the LMM formula raises accuracy of the numerical solutions.

We also calculated the absolute errors in the numerical solutions using the explicit formulas [46,45] of the LMM and obtained results similar to those using the implicit formulas. In the formula SY8, SY8A, and SY8B for $k = 8$, the largest absolute error is of order 1.0D − 9 which is intermediate between that of $k = 4$ and 6 of the implicit formula. The largest absolute error in the explicit formula SY10 for $k = 10$ is of order 1.0D − 11, intermediate between $k = 6$ and 8 of the implicit formula. The largest absolute errors in the explicit formula SY12 for $k = 12$ and SY14 for $k = 14$ are of order 1.0D-12 and 1.0D-14, respectively, close to those in the implicit formula for $k = 8$ and 10, respectively. The implicit formula is more accurate than the explicit one for the same k, which is consistent with the situation that fewer evaluations of f_μ are required in the explicit case ($\beta_n = \beta_0 = 0$) than in the implicit case ($\beta_n = \beta_0 \neq 0$). Since the explicit formulas with large step-number provide accurate results, they also provide useful tools for the initial-value problems. Since we have many choices of the implicit and explicit formulas for the accurate numerical solutions of the initial-value problems, the higher-order LMM method provides useful tools.

Fig. 2 shows convergence of the eigenvalue in the shooting method using the formula 10_4 for $h = \frac{1}{32}$. Fig. 2(a) shows the case of eigenvalue correction according to Eq. (8). In the case of using the eigenvalue obtained via the DM

[3] In [27] $\alpha = (\hbar/m\omega)^{1/2}$ above Eq. (25) should be corrected to $\alpha = (m\omega/\hbar)^{1/2}$.

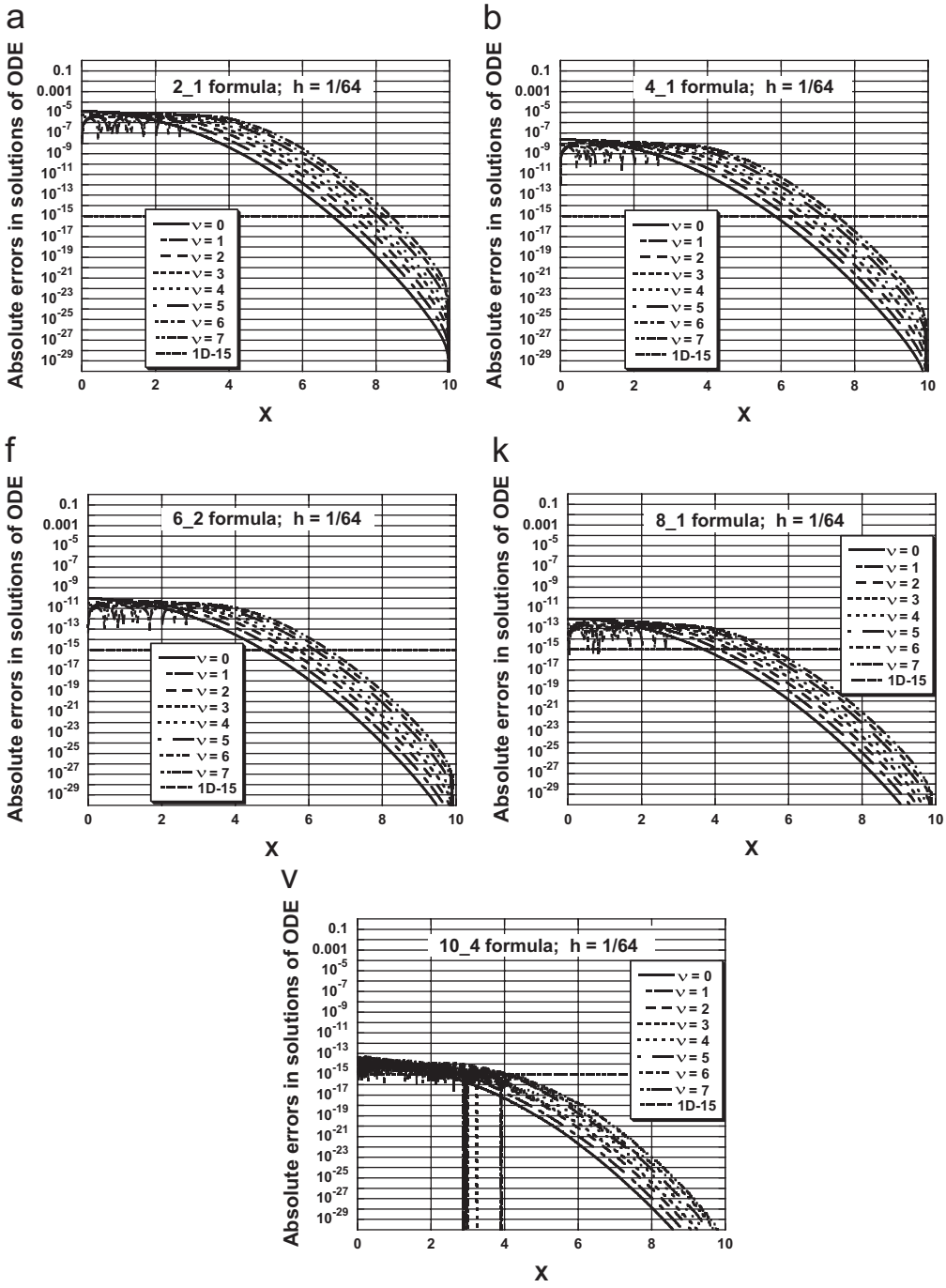

Fig. 1. Absolute errors in the numerical solutions of the initial-value problems for the ordinary differential equations (ODEs) using the linear multistep method (LMM) for the harmonic oscillator. The ordinate is the absolute value of the errors and the abscissa is coordinate x. The quantum state is taken from $v = 0$ to 7. Exact eigenvalue and exact eigenfunction are taken for the initial values. The ODE is solved from $x = 10$ to $x = 0$ with the step-size $h = \frac{1}{64}$ using the implicit formulas (a) 2_1, (b) 4_1, (f) 6_2, (k) 8_1, (v) 10_4.

eigenvalue method for the initial guess, the relative errors are initially of order 1D-4, which decrease to the order of 1D-13 in three to four iterations. In the case of using the diagonal Hamiltonian ME for the initial guess, the relative errors are initially of more than three orders of magnitude lower than those for DM, then reduced to the order of 1D-13

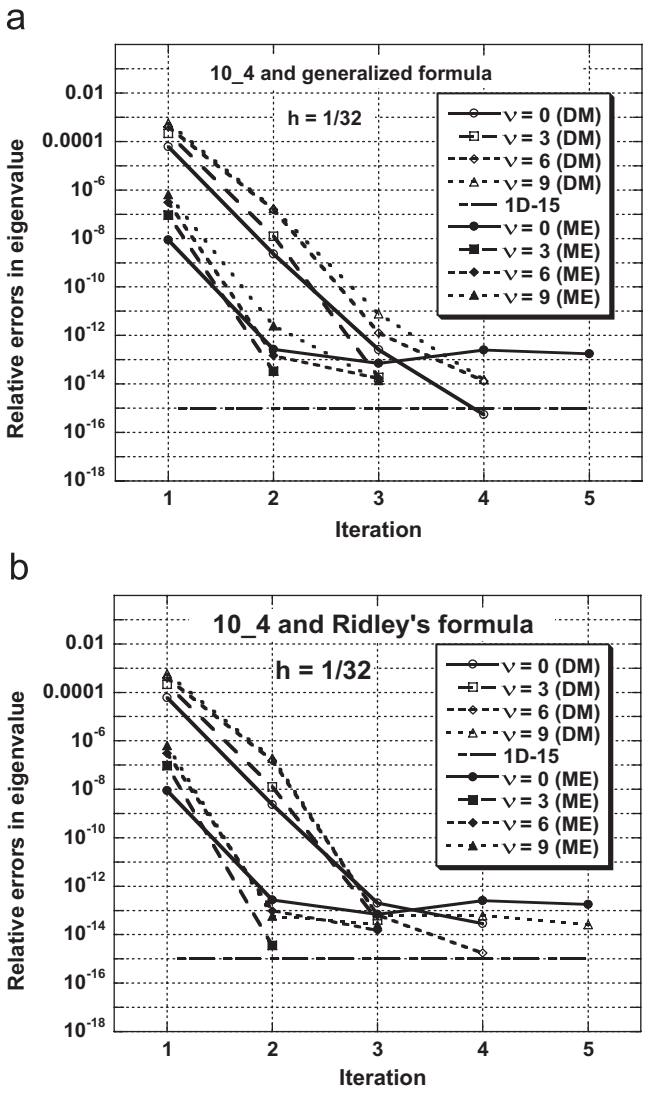

Fig. 2. Convergence of the eigenvalues in the shooting method. Absolute value is taken for the errors. The quantum state is shown for $v = 0$, 3, 6, and 9. DM denotes that the initial values are obtained by using the discretized matrix eigenvalue method. ME denotes that the initial eigenvalues are the diagonal Hamiltonian matrix elements. The initial-value problem for the ordinary differential equation is solved using the formula 10_4 for $h = \frac{1}{32}$. Correcting the eigenvalue by using (a) the generalized formula (8), (b) Ridley's formula (9).

in two more iterations. Similar results are also obtained for the case of eigenvalue correction according to Ridley's formula (9) as shown in Fig. 2(b). Hereafter, we use Ridley's formula (9) rather than Eq. (8), because there is little difference in the convergence of the two formulas but the former is simpler than the latter.

We now investigate the dependence of the errors in the converged eigenvalues, diagonal Hamiltonian MEs and eigenfunctions on the step-number of the formulas. Fig. 3(a) shows that the relative error in the eigenvalues decreases with increasing step-number for small step-number and becomes constant for large step-number. The relative error in the eigenvalues increases for high v. On the other hand, the relative error in the diagonal Hamiltonian MEs is almost constant as a function of step-number and of order 1D-15 for $v = 0$, which is small even for small step-number. Fig. 3(b) shows the largest absolute error in the eigenfunctions in the shooting method against the step-number of the formulas of the LMM for $h = \frac{1}{32}$ and various v. The absolute error in the eigenfunctions decreases as the step-number increases for small step-number and saturates for large step-number. The absolute error in the eigenfunctions also increases for high v.

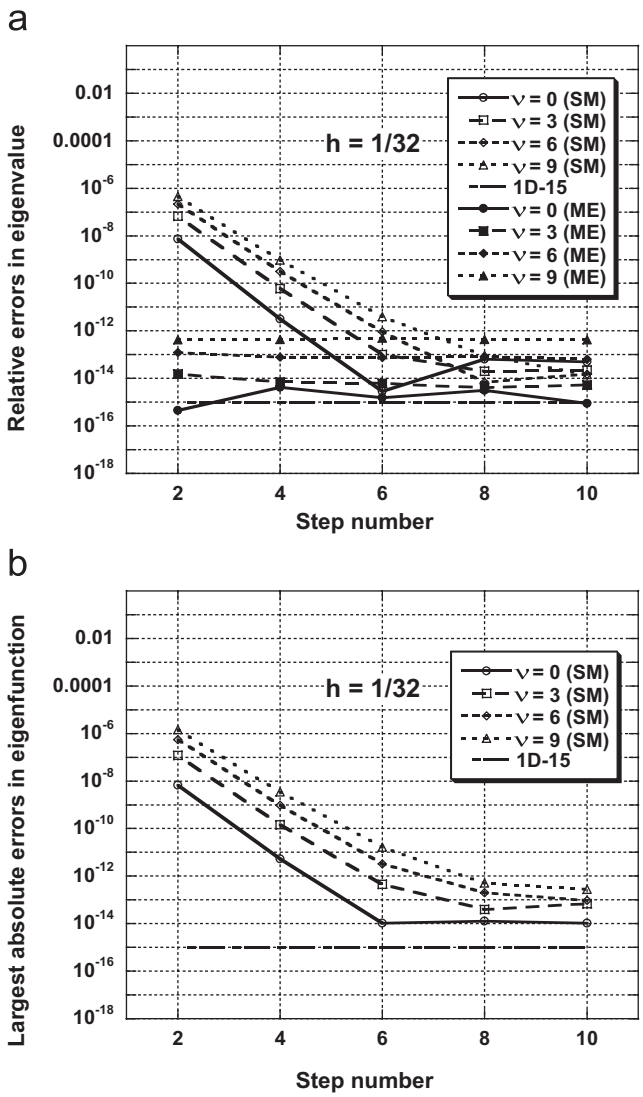

Fig. 3. Dependence of the errors in the eigenvalue and eigenfunction on the step-number in the shooting method. The formula with 2-step is represented by that with 2_1, 4-step by 4_1, 6-step by 6_2, 8-step by 8_1, and 10-step by 10_4. The quantum state is shown for $v = 0$, 3, 6, and 9. (a) The ordinate is the relative error in the eigenvalue and the diagonal Hamiltonian matrix element, the abscissa is the step-number of the formulas with $h = \frac{1}{32}$, (b) the ordinate is the largest absolute error in the eigenfunction and the abscissa is the step-number of the formulas with $h = \frac{1}{32}$.

Table 4 summarizes the results for the harmonic oscillator in the shooting method, over the whole interval $(-10.0, 10.0)$ for the formula 10_4 with $h = \frac{1}{32}$. The accuracy of the eigenvalue in the second column and that of the ME λ_v in the third column range from 13 to 15 digits. The largest absolute value of the absolute errors in the normalized wave functions ranges from 7.2D-15 to 2.8D-13 in the fifth column. The accuracy of the eigenvalue in the sixth column by using the SLEIGN2 with the error tolerance 1.0D-14 ranges from 12 to 14 digits, one digit less accurate than that in the present method. CPU-time only for eigenvalue evaluation with SLEIGN2 is 183 times, namely, of two orders of magnitude, longer than that for evaluating eigenvalue, eigenfunction, and MEs in the present method. The accuracy of the eigenvalue in the seventh column by using the SLCPM12 for the same whole interval in the present method ranges from 12 to 14 digits, two digits less accurate than that in the present method. CPU-time for evaluating eigenvalue and eigenfunction with SLCPM12 is of the same order of magnitude for evaluating eigenvalue, eigenfunction, and MEs in the present method. In concluding this subsection, it is noted that the shooting method provides accurate results comparable to ones shown in Ref. [27, Table 7] obtained by using the DM eigenvalue method.

Table 4
Comparison of exact and computational results using the shooting method presented here and other methods for the harmonic oscillator with $V(x) = x^2$

Quantum no. v	Eigenvalue	λ_v	Exact	LAVAEWF	Other methods SLEIGN2 (a)	SLCPM12 (b)
0	9.99999999999950D − 01	9.99999999999999D − 01	1.0	1.055D − 14	9.99999999999693D − 01	1.00000000008220D + 00
1	2.99999999999993D + 00	3.00000000000000D + 00	3.0	7.161D − 15	3.00000000000071D + 00	3.00000000005223D + 00
2	5.00000000000005D + 00	4.99999999999998D + 00	5.0	1.749D − 14	5.00000000000113D + 00	4.99999999998599D + 00
3	7.00000000000015D + 00	6.99999999999996D + 00	7.0	6.745D − 14	6.99999999999918D + 00	7.00000000005347D + 00
4	9.00000000000004D + 00	8.99999999999984D + 00	9.0	5.640D − 14	9.00000000000215D + 00	8.99999999997320D + 00
5	1.10000000000000D + 01	1.09999999999996D + 01	11.0	8.732D − 14	1.10000000000054D + 01	1.09999999999966D + 01
6	1.29999999999998D + 01	1.29999999999992D + 01	13.0	9.556D − 14	1.30000000000005D + 01	1.30000000000202D + 01
7	1.50000000000001D + 01	1.49999999999979D + 01	15.0	5.054D − 14	1.50000000000002D + 01	1.49999999999268D + 01
8	1.70000000000004D + 01	1.69999999999955D + 01	17.0	2.259D − 13	1.70000000000022D + 01	1.70000000000204D + 01
9	1.90000000000003D + 01	1.89999999999919D + 01	19.0	2.769D − 13	1.89999999999939D + 01	1.89999999999413D + 01
	CPU-time (seconds)	0.2604 (c)			47.84 (d)	0.2003 (e)

The formula 10_4 with $h = \frac{1}{32}$ is used and v is the quantum number. Matrix elements of the wave function $\lambda_v = \langle v| - (\mathrm{d}/\mathrm{d}x)^2 + V(x)|v\rangle/\langle v|v\rangle$; LAVAEWF is the largest absolute value of the absolute (not relative) error in the wave functions. The underline indicates erroneous digits.
(a) The whole interval is $(-\infty, +\infty)$ and the boundary conditions are limit points.
(b) The whole interval is the same one used in the present method.
(c) CPU-time for eigenvalues, eigenfunctions, and matrix elements.
(d) CPU-time for eigenvalues.
(e) CPU-time for eigenvalues and eigenfunctions.

3.2. Anharmonic oscillator

The second application example is an anharmonic oscillator with $V(x) = \mu x^2 + \lambda x^4$, where μ and λ are constants [13,4–7,48,14,27]. The eigenvalue for the bound state has been frequently investigated in a wide range of physics applications and accurate eigenvalues have been numerically obtained using other methods of solution. In order to illustrate performance of our method, we take three typical cases for $(\mu, \lambda) = (0.0, 1.0)$, $(1.0, 1.0)$, and $(-1.0, 1.0)$ in Table 5,

where the formula 10_4 is used over the whole interval $(-5.5, 5.5)$ with $h = \frac{1}{54}$. The eigenvalues of the shooting method and the MEs λ_v are of 13- to 15-digit accuracy, which is comparable to high accuracy obtained by using the DM eigenvalue method [27]. The accuracy of the present work is also comparable to the best values ever reported [4–7,48,14] in double-precision arithmetic operation. For the double-minimum case $(\mu, \lambda) = (-1.0, 1.0)$ we show 10 states that probably have 13-digit or higher accuracy, which means the digits that the eigenvalue (fourth column) and MEs (fifth column) coincide are correct and more digits are substantially correct. Similar results are also obtained by using the DM eigenvalue method [27]. The accuracy of the eigenvalue using the SLEIGN2 (seventh column) ranges from 12 to 14 digits and that by using the SLCPM12 (eighth column) ranges from 10 to 12 digits, one and three digits less accurate than those in the present method. CPU-time using the SLEIGN2 is of two orders of magnitude longer than that in the present method and CPU-time using the SLCPM12 is of the same order as that in the present method.

The third example is an anharmonic oscillator with $V(x) = x^2 + \lambda x^2/(1 + gx^2)$, where λ and g are constants [13,15,53,12,26,27].[4] The eigenvalue for the bound state has been investigated in detail and exact eigenvalues were obtained for special combinations of λ, g, and the quantum number v. We take four typical cases for (λ, g, v) indicated by the notation (*) in Table 6. The eigenvalues of the shooting method are of 13- to 14-digit accuracy and the MEs λ_v are of 15-digit accuracy with the formula 10_4, over the whole interval $(-10.0, 10.0)$ and $h = \frac{1}{32}$; the accuracy is comparable to the best values ever reported [13,15,53,12,26,27] in double-precision arithmetic operation. For another typical case $\lambda = g = 1.0$, we show 10 states whose eigenvalues and MEs coincide with those in [12,26] within the accuracy referred to therein. The accuracy of the eigenvalue using the SLEIGN2 (seventh column) ranges from 11 to 14 digits and that using the SLCPM12 (eighth column) ranges from 10 to 13 digits. CPU-time using the SLEIGN2 is of two orders of magnitude longer than that in the present method and CPU-time using the SLCPM12 is of the same order as that in the present method.

3.3. Morse potential and modified Pöschl–Teller potential

The fourth example is the nonlinear, Morse potential $V(x) = V_0(e^{-2x} - 2e^{-x})$ [34,16,27]. The eigenvalue for the bound state with the quantum number v is given by

$$\lambda_v = -V_0[1 - (v + 0.5)/V_0^{1/2}]^2, \tag{16}$$

where $v = 0, 1, 2, \ldots$ with $v < V_0^{1/2} - 0.5$. We show typical cases for V_0 in Table 7, where the formula is 10_2 and $h = \frac{1}{32}$, the whole interval is $(-4.1875, 35.8125)$ for $V_0 = 1.0$, $(-3.78125, 36.21875)$ for 2.25, $(-3.28125, 36.71875)$ for 6.25, and $(-2.96875, 32.03125)$ for 12.25. The eigenvalues and the MEs of the Hamiltonian are of 13- to 15-digit accuracy. The accuracy of the eigenvalue using the SLEIGN2 (seventh column) ranges from 11 to 15 digits, in general, but in one case the eigenvalue is completely incorrect. The accuracy of the eigenvalue using the SLCPM12 (eighth column) ranges from 10 to 11 digits.

Another form of the Morse potential [39,20,8,27] is $U(x) = D\{1 - \exp[-\alpha(x - x_0)]\}^2$ with $D = \omega_e^2/4\omega_e x_e$, $\alpha = (k\omega_e x_e)^{1/2}$ and $k = 1$, with the theoretical eigenvalues

$$E_v = \omega_e(v + 1/2) - \omega_e x_e(v + 1/2)^2. \tag{17}$$

[4] In [27] $E_0 = \hbar^2/(2m\alpha^2)$ in Sections 4.2 and 4.3 should be corrected to $E_0 = \hbar^2\alpha^2/(2m)$.

Table 5
Comparison of computational results using the shooting method proposed here and other methods

μ	λ	Quantum no. v	Eigenvalue	λ_v	Other methods	SLEIGN2 (a)	SLCPM12 (b)
0.0	1.0	0	1.06036209048460D + 00	1.06036209048418D + 00	1.06036209048418D + 00 (c) (d) (e) (f)	1.06036209048414D + 00	1.06036209050641D + 00
		1	3.79967302980145D + 00	3.79967302980139D + 00	3.79967302980140D + 00 (c) (d) (e)	3.79967302980455D + 00	3.79967302986723D + 00
		2	7.45569793798596D + 00	7.45569793798675D + 00	7.45569793798674D + 00 (c) (d)	7.45569793799075D + 00	7.45569793809411D + 00
		3	1.16447455113780D + 01	1.16447455113781D + 01	1.16447455113782D + 01 (c) (d)	1.16447455113818D + 01	1.16447455115116D + 01
		4	1.62618260188505D + 01	1.62618260188502D + 01	1.62618260188502D + 01 (c)	1.62618260188832D + 01	1.62618260189925D + 01
		5	2.12383729182367D + 01	2.12383729182360D + 01	2.12383729182360D + 01 (c)	2.12383729182668D + 01	2.12383729183597D + 01
		6	2.65284711836812D + 01	2.65284711836823D + 01	2.65284711836825D + 01 (c)	2.65284711837266D + 01	2.65284711837769D + 01
		7	3.20985977109684D + 01	3.20985977109676D + 01	3.20985977109683D + 01 (c)	3.20985977110307D + 01	3.20985977110299D + 01
		8	3.79230010270345D + 01	3.79230010270326D + 01	3.79230010270340D + 01 (c)	3.79230010270918D + 01	3.79230010271058D + 01
		9	4.39811580972870D + 01	4.39811580972881D + 01	4.39811580972897D + 01 (c)	4.39811580973387D + 01	4.39811580973776D + 01
			CPU-time (seconds)	0.2604 (i)		38.54 (j)	0.2103 (k)
1.0	1.0	0	1.39235164153002D + 00	1.39235164153030D + 00	1.39235164153029D + 00 (c) (g) (e) (f)	1.39235164152991D + 00	1.39235164156722D + 00
		1	4.64881270421232D + 00	4.64881270421208D + 00	4.64881270421208D + 00 (c) (g) (e) (f)	4.64881270421257D + 00	4.64881270428911D + 00
		2	8.65504995775976D + 00	8.65504995775928D + 00	8.65504995775931D + 00 (c) (g) (e)	8.65504995776341D + 00	8.65504995785713D + 00
		3	1.31568038980503D + 01	1.31568038980499D + 01	1.31568038980499D + 01 (c) (g) (e)	1.31568038980525D + 01	1.31568038981880D + 01
		4	1.80575574363038D + 01	1.80575574363032D + 01	1.80575574363033D + 01 (c) (g)	1.80575574363240D + 01	1.80575574363965D + 01
		5	2.32974414512221D + 01	2.32974414512231D + 01	2.32974414512232D + 01 (c) (g)	2.32974414512384D + 01	2.32974414513178D + 01
		6	2.88353384595038D + 01	2.88353384595041D + 01	2.88353384595042D + 01 (c) (g)	2.88353384595387D + 01	2.88353384596911D + 01
		7	3.46408483211120D + 01	3.46408483211104D + 01	3.46408483211113D + 01 (c) (g)	3.46408483211941D + 01	3.46408483211669D + 01
		8	4.06903860821064D + 01	4.06903860821058D + 01	4.06903860821064D + 01 (c) (g)	4.06903860821893D + 01	4.06903860820778D + 01
		9	4.69650095056758D + 01	4.69650095056711D + 01	4.69650095056755D + 01 (c) (g)	4.69650095057242D + 01	4.69650095057974D + 01
			CPU-time (seconds)	0.2804 (i)		38.67 (j)	0.1803 (k)

Table 5 (Continued).

μ	λ	Quantum no. v	Eigenvalue	λ_v	Other methods	SLEIGN2 (a)	SLCPM12 (b)
−1.0	1.0	0	6.57653005180482D − 01	6.57653005180713D − 01	6.57653005180715D − 01 (c) (h)	6.57653005181378D − 01	6.57653005201552D − 01
		1	2.83453620211905D + 00	2.83453620211930D + 00	2.83453620211930D + 00 (c) (h)	2.83453620211969D + 00	2.83453620217874D + 00
		2	6.16390125696215D + 00	6.16390125696306D + 00		6.16390125696751D + 00	6.16390125705664D + 00
		3	1.00386461207120D + 01	1.00386461207116D + 01		1.00386461207144D + 01	1.00386461208293D + 01
		4	1.43724065046776D + 01	1.43724065046778D + 01		1.43724065046946D + 01	1.43724065048110D + 01
		5	1.90857146850243D + 01	1.90857146850241D + 01		1.90857146850873D + 01	1.90857146851190D + 01
		6	2.41280754927827D + 01	2.41280754927822D + 01		2.41280754928402D + 01	2.41280754928263D + 01
		7	2.94628559142007D + 01	2.94628559142014D + 01		2.94628559142646D + 01	2.94628559143256D + 01
		8	3.50621490310776D + 01	3.50621490310764D + 01		3.50621490311218D + 01	3.50621490312430D + 01
		9	4.09038562718227D + 01	4.09038562718247D + 01		4.09038562718722D + 01	4.09038562718831D + 01
			CPU-time (seconds)	0.2904 (i)		37.60 (j)	0.1803 (k)

μ and λ are the parameters in the potential function $V(x) = \mu x^2 + \lambda x^4$ and v is the quantum number. $\lambda_v = \langle v| - (\mathrm{d}/\mathrm{d}x)^2 + V(x)|v\rangle / \langle v|v\rangle$ are the matrix elements for the wave function. The formula 10_4 is used with $h = \frac{1}{64}$. The underline indicates erroneous digits.
(a) The whole interval is $(-\infty, +\infty)$ and the boundary conditions are limit points.
(b) The whole interval is the same one used in the present method.
(c) Banerjee et al. [5].
(d) Fernández et al. [14].
(e) Schiffrer and Stanzial [48].
(f) Biswas et al. [7].
(g) Banerjee [4].
(h)Basla et al. [6].
(i) CPU-time for eigenvalues, eigenfunctions, and matrix elements.
(j) CPU-time for eigenvalues.
(k) CPU-time for eigenvalues and eigenfunctions.

Table 6
Comparison of computational results using the shooting method proposed here and other methods

λ	g	Quantum no. v	Eigenvalue	λ_v	Other methods	SLEIGN2 (a)	SLCPM12 (b)
−042	0.1	0	7.99999999999889D − 01	7.99999999999999D − 01	0.8 (*) (c)	7.99999999999912D − 01	8.00000000034215D − 01
		1	2.45569858511913D + 00	2.45569858511910D + 00	2.455698585119 (d)	2.45569858512021D + 00	2.45569858519653D + 00
		2	4.19789589344418D + 00	4.19789589344427D + 00	4.197895893444 (d)	4.19789589344622D + 00	4.19789589347335D + 00
		3	5.99139883718982D + 00	5.99139883718982D + 00	5.991398837190 (d)	5.99139883718923D + 00	5.99139883717249D + 00
		4	7.82009765426843D + 00	7.82009765426847D + 00	7.820097654268 (d)	7.82009765427081D + 00	7.82009765430676D + 00
		5	9.67453731290618D + 00	9.67453731290616D + 00	9.674537312906 (d)	9.67453731291134D + 00	9.67453731292853D + 00
−046	0.1	1	2.39999999999993D + 00	2.40000000000000D + 00	2.4 (*) (e)	2.40000000000113D + 00	2.40000000007987D + 00
−0495357508034270	0.1	2	4.04642491965734D + 00	4.04642491965729D + 00	4.04642491965730 (*) (f)	4.04642491965905D + 00	4.04642491969360D + 00
−0527762515838433	0.1	3	5.72237484161566D + 00	5.72237484161565D + 00	5.72237484161567 (*) (g)	5.72237484161543D + 00	5.72237484158956D + 00
10	1.0	0	1.23235072340597D + 00	1.23235072340606D + 00	1.23235072340606 (h)	1.23235072340549D + 00	1.23235072341027D + 00
		1	3.50738834890524D + 00	3.50738834890528D + 00	3.507388348905 (d)	3.50738834890525D + 00	3.50738834892731D + 00
		2	5.58977893373718D + 00	5.58977893373715D + 00	5.589778933736 (d)	5.58977893373757D + 00	5.58977893377206D + 00
		3	7.64820124171947D + 00	7.64820124171933D + 00	7.648201241723 (d)	7.64820124171862D + 00	7.64820124173403D + 00
		4	9.68404201523016D + 00	9.68404201523001D + 00	9.68404201523017 (h)	9.68404201523200D + 00	9.68404201522972D + 00
		5	1.17122374702085D + 01	1.17122374702079D + 01		1.17122374702125D + 01	1.17122374702247D + 01
		6	1.37332410121094D + 01	1.37332410121083D + 01		1.37332410121109D + 01	1.37332410121294D + 01
		7	1.57506387971465D + 01	1.57506387971441D + 01		1.57506387971466D + 01	1.57506387971502D + 01
		8	1.77647791014220D + 01	1.77647791014168D + 01		1.77647791014239D + 01	1.77647791014221D + 01
		9	1.97768948716950D + 01	1.97768948716864D + 01		1.97768948716886D + 01	1.97768948717083D + 01
			CPU-time (seconds)	0.2604 (i)		48.42 (j)	0.2704 (k)

λ and g are the parameters in the potential function $V(x) = x^2 + \lambda x^2/(1 + gx^2)$ and v is the quantum number. $\lambda_v = \langle v| - (\mathrm{d}/\mathrm{d}x)^2 + V(x)|v\rangle/\langle v|v\rangle$ are the matrix elements for the wave function. The formula 10_4 is used with $h = \frac{1}{32}$. The underline indicates erroneous digits.
(a) The whole interval is $(-\infty, +\infty)$ and the boundary conditions are limit points.
(b) The whole interval is the same one used in the present method.
(*) Exact eigenvalue.
(c) Fack and Vanden Berghe [13] and Flessas [15].
(d) Fack et al. [12].
(e) Fack and Vanden Berghe [13] and Varma [53].
(f) The result is calculated by using expressions in Fack and Vanden Berghe [13] and Flessas [15].
(g) The result is calculated by using expressions in Fack and Vanden Berghe [13] and Varma [53].
(h) Hodgson [26].
(i) CPU-time for eigenvalues, eigenfunctions, and matrix elements.
(j) CPU-time for eigenvalues.
(k) CPU-time for eigenvalues and eigenfunctions.

Table 7
Comparison of computational results using the shooting method proposed here and other methods for the Morse potential (A) $V(x) = V[\exp(-2(x) - 2\exp(-x)]$ with the depth V_0 and (B) $V(x) = D\{1 - \exp[-\alpha(x - x_0)]\}^2$

	Depth	Quantum no.	Eigenvalue	λ_v	Exact	Other methods	
	V_0	v				SLEIGN2 (a)	SLCPM12 (b)
(A)	1.0	0	−2.50000000000006D − 01	−2.50000000000000D − 01	−0.25	−2.50000000005548D − 01	−2.49999999998742D − 01
	2.25	0	−1.00000000000001D + 00	−1.00000000000001D + 00	−1.0	−1.00000000000033D + 00	−9.99999999998645D − 01
	6.25	0	−4.00000000000000D + 00	−4.00000000000005D + 00	−4.0	−4.00000000000161D + 00	−3.99999999998515D + 00
		1	−9.99999999999990D − 01	−1.00000000000010D + 00	−1.0	−1.00000000000013D + 00	−9.99999999986406D − 01
	12.25	0	−9.00000000000000D + 00	−9.00000000000032D + 00	−9.0	−7.30458442146399D + 00	−8.99999999995840D + 00
		1	−3.99999999999998D + 00	−4.00000000000113D + 00	−4.0	−4.00000000000000D + 00	−3.99999999995699D + 00
		2	−1.00000000000002D + 00	−1.00000000000131D + 00	−1.0	−9.99999999999620D − 01	−9.99999999976265D − 01
(B)		0	2.40899680000009D + 01	2.40899679999999D + 01	2.40899680000000D + 01 (c)	2.40899680000001D + 01	2.40899680008534D + 01
		1	7.0830720000009D + 01	7.08030719999997D + 01	7.08030720000000D + 01 (c)	1.16093666999078D + 02	7.08030720009380D + 01
		2	1.15560400000002D + 02	1.15560399999999D + 02	1.15560400000000D + 02 (c)	1.15560400000237D + 02	1.15560400000930D + 02
		3	1.58361951999999D + 02	1.58361951999995D + 02	1.58361952000000D + 02 (c)	5.53012557878005D + 02	1.58361952000597D + 02
		4	1.99207728000000D + 02	1.99207727999986D + 02	1.99207728000000D + 02 (c)	1.99207727999271D + 02	1.99207727999972D + 02
		5	2.38097728000000D + 02	2.38097727999968D + 02	2.38097728000000D + 02 (c)	2.63651664645498D + 02	2.38097727999898D + 02
		6	2.75031952000001D + 02	2.75031951999935D + 02	2.75031952000000D + 02 (c)	4.60420259313651D + 02	2.75031952000240D + 02
		7	3.10010400000001D + 02	3.10010399999882D + 02	3.10010400000000D + 02 (c)	3.10010399999835D + 02	3.10010400000375D + 02
		8	3.43033072000004D + 02	3.43033071999813D + 02	3.43033072000000D + 02 (c)	3.66547725306152D + 02	3.43033072000553D + 02
		9	3.74099968000001D + 02	3.74099967999700D + 02	3.74099968000000D + 02 (c)	5.66936348747298D + 02	3.74099968000107D + 02
		10	4.03211087999998D + 02	4.03211087999593D + 02	4.03211088000000D + 02 (c)	4.60200638994147D + 02	4.03211087999541D + 02
			CPU-time (seconds)	0.3505 (d)		161.7 (e)	0.1903 (f)

v is the quantum number. $\lambda_v = \langle v| - (\mathrm{d}/\mathrm{d}x)^2 + V(x)|v\rangle/\langle v|v\rangle$ are the matrix elements for the wave function. The formula 10_2 is used with $h = 1/32$ for (A) and 10_4 is used for (B). The underline indicates erroneous digits.
(a) The whole interval is $(-\infty, +\infty)$ and the boundary conditions are limit points.
(b) The whole interval is the same one used in the present method.
(c) Dagher and Kobeissi [8].
(d) CPU-time for eigenvalues, eigenfunctions, and matrix elements.
(e) CPU-time for eigenvalues.
(f) CPU-time for eigenvalues and eigenfunctions.

Table 8
Comparison of computational results using the shooting method proposed here and other methods for the modified Pöschl–Teller potential $V(x) = -V/\cosh(x)$ with th depth V_0

Depth	Quantum no.	Eigenvalue	λ_v	Exact	Other methods	
V_0	v				SLEIGN2 (a)	SLCPM12 (b)
1.0	0	−3.81966011250115D − 01	−3.81966011250102D − 01	−3.81966011250105D − 01	−3.81966011250499D − 01	−3.81966011271461D − 01
2.0	0	−9.99999999999989D − 01	−0.00000000000000D + 00	−1.0	−1.00000000000000D + 00	−9.99999999997747D − 01
6.0	0	−3.99999999999989D + 00	−4.00000000000002D + 00	−4.0	−3.99999999999983D + 00	−4.00000000000045D + 00
	1	−9.99999999999945D − 01	−1.00000000000004D + 00	−1.0	−1.00000000000061D + 00	−1.00000000011497D + 00
12.0	0	−9.00000000000012D + 00	−9.00000000000015D + 00	−9.0	−8.99999999999752D + 00	−9.00000000000925D + 00
	1	−3.99999999999995D + 00	−4.00000000000055D + 00	−4.0	−4.00000000000052D + 00	−4.00000000002033D + 00
	2	−1.00000000000058D + 00	−1.00000000000064D + 00	−1.0	−1.00000000000292D + 00	−9.99999999992384D − 01
		CPU-time (seconds)	0.5207 (c)		1.592 (d)	0.1202 (e)

v is the quantum number. $\lambda_v = \langle v| - (\mathrm{d}/\mathrm{d}x)^2 + V(x)|v\rangle/\langle v|v\rangle$ are the matrix elements for the wave function. The formula 10_4 is used with $h = 1/32$. The underline indicates erroneous digits.
(a) Since the SLEIGN2 fails computation for the whole interval $(-\infty, +\infty)$ with the limit points due to the overflow of $\cosh(x)$, the finite intervals used in our calculations are also used for computations with the limit points.
(b) Since the SLCPM12 gives incorrect, spurious eigenvalues for both too wide and too narrow intervals, the interval is taken $(-24.0, 24.0)$ for $V_0 = 1.0$ and $(-20.0, 20.0)$ for $V_0 = 2.0$, 6.0, and 12.0.
(c) CPU-time for eigenvalues, eigenfunctions and matrix elements.
(d) CPU-time for eigenvalues.
(e) CPU-time for eigenvalues and eigenfunctions.

The eigenvalues and the MEs of the Hamiltonian for the case $x_0 = 2.40873$, $\omega_e = 48.66888$, and $\omega_e x_e = 0.977888$ [20,8], are of 13- to 15-digit accuracy for the formula 10_4, the whole interval (0.8540425, 6.3227925) and $h = \frac{1}{128}$, which is of comparable accuracy to that using the DM eigenvalue method [27]. The eigenvalues using the SLEIGN2 (seventh column) are incorrect in many cases, but in some cases the accuracy of the eigenvalue ranges from 12 to 14 digits. The accuracy of the eigenvalue using the SLCPM12 (eighth column) ranges from 11 to 13 digits. CPU-time using the SLEIGN2 is of two orders of magnitude longer than that in the present method and CPU-time using the SLCPM12 is of the same order as that in the present method.

The fifth example is the modified Pöschl–Teller potential, the symmetric nonlinear potential hole of the form $V(x) = -V_0/\cosh^2(x)$, where V_0 is a constant [34,16,27]. The eigenvalue for the bound state with the quantum number v is given by

$$\lambda_v = -[-(1 + 2v) + (1 + 4V_0)^{1/2}]^2/4, \tag{18}$$

where $v = 0, 1, 2, \ldots$ with $v < [-1 + (1 + 4V_0)^{1/2}]/2$, and the wave function is given in [34,16]. We show typical cases for V_0 in Table 8, where the formula is 10_4, $h = \frac{1}{32}$, and the whole interval is $(-27.5, 27.5)$ for $V_0 = 1.0$, and is $(-24.0, 24.0)$ for $V_0 = 2.0$, 6.0 and 12.0. The eigenvalues and MEs of the Hamiltonian are of 13- to 15-digit accuracy comparable to the results obtained by the DM eigenvalue method [27]. Since the SLEIGN2 fails computation for the default boundary conditions due to the overflow of $\cosh(x)$, the finite intervals used in our calculations are also used for computations. The accuracy of the eigenvalue using the SLEIGN2 (sixth column) ranges from 12 to 15 digits, while CPU-time using the SLEIGN2 is of three times longer than that in the present method. Since the SLCPM12 gives incorrect, spurious eigenvalues for both too wide and too narrow intervals, we take the whole interval in the present method as initial guess, and change the end points by trial and error. The accuracy of the eigenvalue by using the SLCPM12 (seventh column) ranges from 10 to 13 digits and CPU-time using the SLCPM12 is of the same order as that in the present method.

4. Conclusions

We have developed the methods for accurate numerical solution of the eigenvalue problem in quantum mechanics in one-dimension. We have presented the new, accurate, and stable formulas for solving the initial-value problem of the second-order ordinary differential equation and demonstrated their performance. We also obtained the highly accurate eigenvalues and eigenfunctions by using the shooting method. In some cases our results are the best ever reported in the literature. The accurate results are comparable to those we obtained before by using the DM eigenvalue method. Since the method developed here is simple, we hope that it serves as a powerful tool across a wide range of research domains.

Acknowledgments

The author thanks Professor Emeritus Shigeyuki Aono at Kanazawa University, Professor Hirotoshi Ono at Shohoku College, and Dr Toshiaki Iitaka at RIKEN (The Institute of Physical and Chemical Research) for their helpful comments on the manuscript. The author also thanks the one of referees for fruitful comments and suggestions for revising the manuscript.

References

[1] P.B. Bailey, W.N. Everitt, A. Zettl, Algorithm 810: the SLEIGN2 Strum–Liouville code, ACM Trans. Math. Software 27 (2001) 143–192 (See also SLEIGN2 homepage ⟨http://www.math.niu.edu/∼zettl/SL2/⟩).
[2] P.B. Bailey, B.S. Garbow, H.G. Kaper, A. Zettl, Eigenvalue and eigenfunction computations for Strum–Liouville problems, ACM Trans. Math. Software 17 (1991) 491–499.
[3] P.B. Bailey, M.K. Gordon, L.F. Shampine, Automatic solution of the Strum–Liouville problem, ACM Trans. Math. Software 4 (1978) 193–208.
[4] K. Banerjee, General anharmonic oscillators, Proc. Roy. Soc. London A 364 (1978) 265–275.
[5] K. Banerjee, S.P. Bhatnagar, V. Choudhry, S.S. Kanwal, The anharmonic oscillator, Proc. Roy. Soc. London A 360 (1978) 575–586.
[6] R. Basla, M. Plo, J.G. Esteve, A.F. Pacheco, Simple procedure to compute accurate energy levels of a double-well anharmonic oscillator, Phys. Rev. D 28 (1983) 1945–1948.

[7] S.N. Biswas, K. Datta, R.P. Saxena, P.K. Srivastava, V.S. Varma, Eigenvalues of λx^{2m} anharmonic oscillators, J. Math. Phys. 14 (1973) 1190–1195.
[8] M. Dagher, H. Kobeissi, Vibrational eigenvalues for all levels for the Lennard-Jones potential, J. Comput. Chem. 9 (1988) 647–649.
[9] G. Dahlquist, Convergence and stability in the numerical integration of ordinary differential equations, Math. Scand. 4 (1956) 33–53.
[10] G. Dahlquist, Stability and error bounds in the numerical integration of ordinary differential equations, Trans. Roy. Inst. Technol. Stockholm 130 (1959).
[11] A.S. Douglas, On the Strum–Liouville equation with two-point boundary conditions, Proc. Cambridge Philos. Soc. 52 (1956) 636–639.
[12] V. Fack, H. De Meyer, G. Vanden Berghe, Dynamical-group approach to the $x^2 + \lambda x^2/(1 + gx^2)$ potential, J. Math. Phys. 27 (1986) 1340–1343.
[13] V. Fack, G. Vanden Berghe, A finite difference approach for the calculation of perturbed oscillator energies, J. Phys. A 18 (1985) 3355–3363.
[14] F.M. Fernández, A.M. Mesón, E.A. Castro, A simple iterative solution of the Schrödinger equation in matrix representation form, J. Phys. A 18 (1985) 1389–1398.
[15] G.P. Flessas, On the Schrödinger equation for the $x^2 + \lambda x^2/(1 + gx^2)$ interaction, Phys. Lett. A 83 (1981) 121–122.
[16] S. Flügge, Practical Quantum Mechanics, Springer, Berlin, 1971.
[17] C. Froese Fischer, The Hartree–Fock Method for Atoms, Wiley, New York, 1977.
[18] T. Fukushima, Symmetric Multistep Methods Revisited, in: 30th Symposium on Celestial Mechanics, 1998, pp. 229–247.
[19] T. Fukushima, Symmetric multistep methods revisited: II. Numerical experiments, in: 173rd Colloquium of the International Astronomical Union, 1999, pp. 309–314.
[20] G.C. Groenenboom, H.M. Buck, Solving the discretized time-independent Schrödinger equation with the Lanczos procedure, J. Chem. Phys. 92 (1990) 4374–4379.
[21] E. Hairer, C. Lubich, Symmetric multistep methods over long times, Numer. Math. 97 (2004) 699–723.
[22] E. Hairer, S.P. Nørsett, G. Wanner, Solving Ordinary Differential Equations I. Nonstiff Problems, second ed., Springer, Berlin, 1993.
[23] D.R. Hartree, The Calculation of Atomic Structures, Wiley, New York, 1957.
[24] D.R. Hartree, Numerical Analysis, second ed., Oxford University Press, Oxford, 1958.
[25] P. Henrici, Discrete Variable Methods in Ordinary Differential Equations, Wiley, New York, 1962.
[26] R.J.W. Hodgson, High-precision calculation of the eigenvalues for the $x^2 + \lambda x^2/(1 + gx^2)$ potential, J. Phys. A 21 (1988) 1563–1570.
[27] H. Ishikawa, An accurate method for numerical calculations in quantum mechanics, J. Phys. A 35 (2002) 4453–4476 ⟨http://stacks.iop.org/JPhysA/35/4453⟩.
[28] L.Gr. Ixaru, CP methods for the Schrödinger equation, J. Comput. Appl. Math. 125 (2000) 347–357.
[29] L.Gr. Ixaru, H. De Meyer, G. Vanden Berghe, CP methods for the Schrödinger equation revisited, J. Comput. Appl. Math. 88 (1997) 289–314.
[30] L.Gr. Ixaru, H. De Meyer, G. Vanden Berghe, SLCPM12—a program for solving regular Strum–Liouville problems, Comput. Phys. Comm. 118 (1999) 259–277.
[31] L.Gr. Ixaru, M. Rizea, Comparison of some four-step methods for the numerical solution of the Schrödinger equation, Comput. Phys. Comm. 38 (1985) 329–337.
[32] J.D. Lambert, Computational Methods in Ordinary Differential Equations, Wiley, London, 1973.
[33] J.D. Lambert, I.A. Watson, Symmetric multistep methods for periodic initial value problems, J. Inst. Math. Appl. 18 (1976) 189–202.
[34] L.D. Landau, E.M. Lifshitz, Quantum Mechanics, third ed., Elsevier, Amsterdam, 1963.
[35] V. Ledoux, M. Van Daele, G. Vanden Berghe, CP methods of higher order for Strum–Liouville and Schrödinger equations, Comput. Phys. Comm. 162 (2004) 151–165.
[36] V. Ledoux, M. Van Daele, G. Vanden Berghe, MATSLISE: A MATLAB package for the numerical solution of Strum–Liouville and Schrödinger equations, ACM Trans. Math. Software 31 (2005) 532–554 (See also ⟨http://users.ugent.be/∼vledoux/MATSLISE/⟩).
[37] M. Marletta, J.D. Pryce, Automatic solution of Strum–Liouville problems using the Pruess method, J. Comput. Appl. Math. 39 (1992) 57–78.
[38] M. Mori, Programming for Numerical Calculations with FORTRAN77 (in Japanese, FORTRAN77 Suti Keisan Puroguramingu), Iwanami, Tokyo, 1987.
[39] L. Pauling, E.B. Wilson, . Jr, Introduction to Quantum Mechanics, Dover, New York, 1985.
[40] W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery, Numerical Recipes: The Art of Scientific Computing, second ed., Cambridge University Press, Cambridge, 1992.
[41] S. Pruess, C.T. Fulton, Mathematical software for Strum–Liouville problems, ACM Trans. Math. Software 19 (1993) 360–376.
[42] J.D. Pryce, Numerical Solution of Strum–Liouville Problems, Oxford University Press, Oxford, 1993.
[43] J.D. Pryce, A test package for Strum–Liouville solvers, ACM Trans. Math. Software 25 (1999) 21–57.
[44] J.D. Pryce, M. Marletta, A new multi-purpose software package for Schrödinger and Strum-Liouville computations, Comput. Phys. Comm. 62 (1991) 42–52.
[45] G.D. Quinlan, Resonances and instabilities in symmetric multistep methods, Preprint (astro-ph/9901136), 1999.
[46] G.D. Quinlan, S. Tremaine, Symmetric multistep methods for the numerical integration of planetary orbits, Astron. J. 100 (1990) 1694–1700.
[47] E.C. Ridley, The self-consistent field for Mo^+, Proc. Cambridge Philios. Soc. 51 (1955) 702–706.
[48] G. Schiffrer, D. Stanzial, Improved calculations for anharmonic oscillator using the gradient method, Nuovo Cimento B 90 (1985) 74–84.
[49] L.F. Shampine, H.A. Watts, Global error estimation for ordinary differential equations, ACM Trans. Math. Software 2 (1976) 172–186.
[50] T.E. Simos, A family of trigonometrically-fitted symmetric methods for the efficient solution of the Schrödinger equation and related problems, J. Math. Chem. 34 (2003) 39–58.
[51] J. Stoer, R. Bulirsch, Introduction to Numerical Analysis, third ed., Springer, New York, 2002.

[52] G. Vanden Berghe, V. Fack, H.E. De Meyer, Numerical methods for solving radial Schrödinger equations, J. Comput. Appl. Math. 28 (1989) 391–401.
[53] V.S. Varma, On the $x^2 + \lambda x^2/1 + gx^2$) interaction, J. Phys. A 14 (1981) L489–L492.
[54] J. Vigo-Aguiar, T.E. Simos, Family of twelve steps exponential fitting symmetric multistep methods for the numerical solution of the Schrödinger equation, J. Math. Chem. 32 (2002) 257–270.