

Efficient approximation of Sturm-Liouville and Schrödinger problems

V. Ledoux

Department of Applied Mathematics and Computer Science
Ghent university

Symposium on Numerical approaches of oscillatory functions
January 16-18, 2008

Outline

- 1 Introduction
- 2 Numerical computation of the eigenvalues
- 3 Piecewise Perturbation methods
- 4 Modified integral series methods
- 5 Conclusion

Outline

- 1 Introduction
 - The Sturm-Liouville problem
 - Elementary theory
- 2 Numerical computation of the eigenvalues
- 3 Piecewise Perturbation methods
- 4 Modified integral series methods
- 5 Conclusion

The Sturm-Liouville problem



Jacques Sturm (1803-1855)



Joseph Liouville (1809-1882)

$$-\frac{d}{dx} \left[p(x) \frac{dy(x)}{dx} \right] + q(x)y(x) = Ew(x)y(x)$$

The Sturm-Liouville problem

$$-\frac{d}{dx} \left[p(x) \frac{dy(x)}{dx} \right] + q(x)y(x) = Ew(x)y(x), \quad x \in (a, b)$$

- The value of E is not specified. Finding the values of E for which there exists a solution $y \neq 0$ satisfying some BCs is part of the so-called **Sturm-Liouville problem** (SLP).
- One-dimensional model of many important physical processes (quantum physics, quantum chemistry, geophysical applications, vibration and heat flow problems,...).
- In many applications, the SLP problem describes the oscillation in the physical system.

The Sturm-Liouville problem

$$-\frac{d}{dx} \left[p(x) \frac{dy(x)}{dx} \right] + q(x)y(x) = Ew(x)y(x), \quad x \in (a, b)$$

The Liouville normal form

- When p and w constant: divide by p and take $t = \sqrt{w/p}x$ to obtain

$$-\frac{d^2y(t)}{dt^2} + Q(t)y(t) = Ey(t)$$

with $Q(t) = q(x)/p$.

- = one-dimensional, time-independent form of the **Schrödinger equation**
- Any Sturm-Liouville problem can be put in this form.

The Schrödinger problem



Erwin Schrödinger (1887 - 1961)

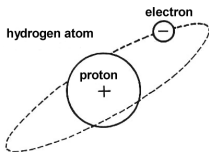
The Schrödinger problem

$$\frac{d^2 y}{dx^2} = [V(x) - E]y(x)$$

- Description of quantum mechanical systems
- $V(x)$: potential function
- E : energy level (eigenvalue)
- y : wavefunction

Some examples

- **Hydrogen atom equation**



The classical one-dimensional equation for quantum modelling of the hydrogen atom:

$$-y'' + \left(\frac{2}{x^2} - \frac{1}{x} \right) y = Ey, \quad x \in (0, +\infty)$$

(Interaction of the proton and the electron)

- **Coffey-Evans equation**

Equation which arises in a model of the coupling between dipoles in polarizable liquids such as liquid crystal displays (LCD):

$$-y'' + \left(-2\beta \cos 2x + \beta^2 \sin^2 2x \right) y = Ey, \quad x \in [-\pi/2, \pi/2]$$

- **Mathieu equation**

The French mathematician E.L. Mathieu investigated this equation in 1868 while seeking a description of the vibrations of an elliptical membrane (drumheads).

$$y'' = [2 \cos(2x) - E]y, \quad y(0) = y(\pi) = 0.$$

The Sturm-Liouville boundary value problem

$$-\frac{d}{dx} \left[p(x) \frac{dy(x)}{dx} \right] + q(x)y(x) = Ew(x)y(x), \quad x \in [a, b]$$

- Boundary conditions (BCs):

$$\begin{cases} a_0y(a) + b_0p(a)y'(a) = 0, & (y' = dy/dx) \\ a_1y(b) + b_1p(b)y'(b) = 0 \end{cases}$$

- **Eigenvalue**: E for which there exists a nonzero solution y satisfying the BCs.
- **Eigenfunction**: corresponding solution y .

The Sturm-Liouville boundary value problem

$$-\frac{d}{dx} \left[p(x) \frac{dy(x)}{dx} \right] + q(x)y(x) = Ew(x)y(x), \quad x \in [a, b]$$

Example

$$\frac{d^2y}{dx^2} + Ey = 0, \quad 0 \leq x \leq \pi,$$

Boundary value problem: $y(0) = 0, \quad y(\pi) = 0$.

- This equation represents the physics of a vibrating elastic string.
- **Eigenvalues:** $E_k = (k + 1)^2, \quad k = 0, 1, 2, \dots$
- **Eigenfunctions:** $y_k(x) = \sin((k + 1)x), \quad k = 0, 1, 2, \dots$
- k : eigenvalue index

Oscillatory character of the eigenfunctions

Fundamental theorem

For a regular Sturm-Liouville problem

- The eigenvalues E_k are real and simple.
- The E_k can be ordered as an increasing sequence tending to infinity,

$$E_0 < E_1 < E_2 < \dots$$

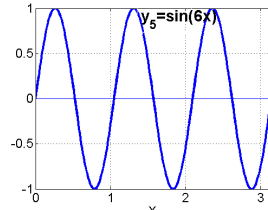
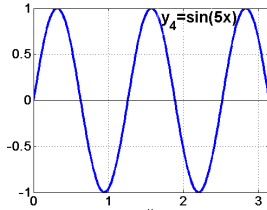
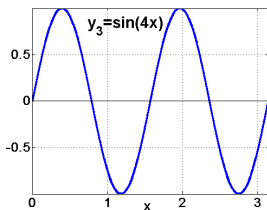
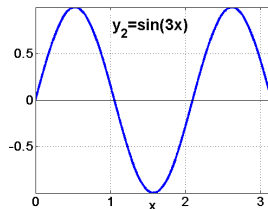
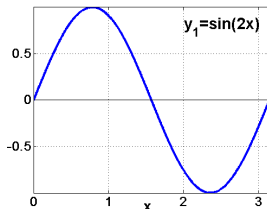
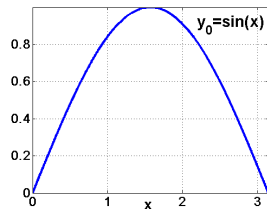
- The eigenfunction $y_k(x)$ corresponding to E_k has **exactly k zeros** on (a, b) .

- A higher eigenfunction is oscillating 'more rapidly' than a lower eigenfunction.
- This oscillatory behaviour of the solutions forces a standard (naive) integrator to take increasingly smaller steps.

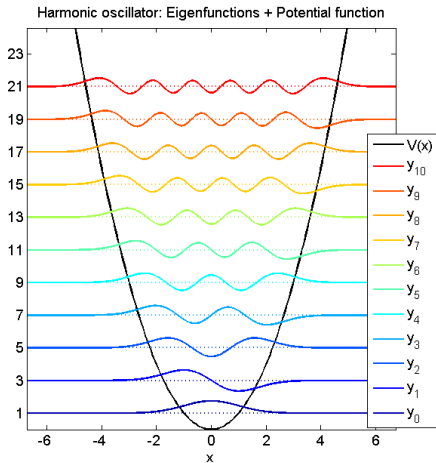
Oscillatory character of the eigenfunctions

$$\frac{d^2y}{dx^2} + Ey = 0, \quad y(0) = 0, \quad y(\pi) = 0.$$

$$E_k = (k + 1)^2, \quad y_k(x) = \sin((k + 1)x), \quad k = 0, 1, 2, \dots$$



Oscillatory character of the eigenfunctions



Schrödinger problem:

Classically allowed region

- $E > V(x)$
- solution is oscillating

Classically forbidden region

- $E < V(x)$
- solution increases/decreases exponentially

Harmonic oscillator

$$y'' = (x^2 - E)y, \quad E_k = 2k + 1$$

Outline

- 1 Introduction
- 2 Numerical computation of the eigenvalues**
 - Some classical methods
 - Shooting methods
- 3 Piecewise Perturbation methods
- 4 Modified integral series methods
- 5 Conclusion

Matrix methods

- Methods based on finite differences/finite elements.
- Typically they lead to matrix eigenvalue problems $\mathbf{A}y = Ey$ or generalized matrix eigenvalue problems $\mathbf{A}y = E\mathbf{B}y$, with a matrix of band structure.

Matrix methods

- Methods based on finite differences/finite elements.
- Typically they lead to matrix eigenvalue problems $\mathbf{A}y = Ey$ or generalized matrix eigenvalue problems $\mathbf{A}y = E\mathbf{B}y$, with a matrix of band structure.

$$y'' = [V(x) - E]y, \quad y(a) = y(b) = 0$$

Numerov for $y'' = f(x, y)$:

$$y_{i-1} - 2y_i + y_{i+1} = \frac{h^2}{12}(f_{i-1} + 10f_i + f_{i+1}), \quad f_i = f(x_i, y_i)$$

$$\mathbf{A}y = E\mathbf{B}y$$

\mathbf{A} and \mathbf{B} are tridiagonal: $\mathbf{A} = \frac{1}{h^2}\mathbf{M} + \mathbf{B}V$, $\mathbf{B} = \mathbf{I} - \frac{1}{12}\mathbf{M}$,

$$\mathbf{M} = \begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & & \ddots & \ddots & \\ & & & -1 & 2 \end{pmatrix}, \quad \mathbf{V} = \begin{pmatrix} V(x_1) & & & & \\ & V(x_2) & & & \\ & & \ddots & & \\ & & & \ddots & \\ & & & & V(x_{n-1}) \end{pmatrix}$$

Matrix methods

- Methods based on finite differences/finite elements.
- Typically they lead to matrix eigenvalue problems $\mathbf{A}y = Ey$ or generalized matrix eigenvalue problems $\mathbf{A}y = E\mathbf{B}y$, with a matrix of band structure.

Pros

- Simple to set up

Cons

- The infinite-dimensional problem is replaced by a matrix problem of dimension $n - 1$.
- The use of a uniform mesh is not always a good idea.
- Main disadvantage is the difficulty in providing high order approximations with uniform error bounds.

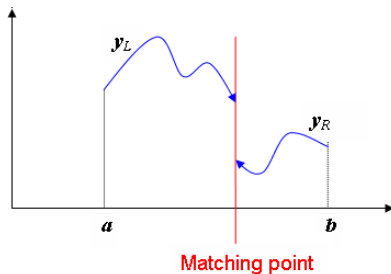
Matrix methods

Table: Errors ($\times 10^3$) obtained with $n = 40$ for $V(x) = e^x$, $y(0) = y(\pi) = 0$.

k	E_k	Numerov	Exp. fitted Numerov
0	4.8966694	0.0028	0.0014
1	10.045190	0.0427	0.0127
2	16.019267	0.2272	0.0424
3	23.266271	0.8837	0.1040
4	32.263707	2.8802	0.1959
5	43.220020	8.0432	0.3132
6	56.181594	19.6872	0.4535
7	71.152998	43.2849	0.6197
8	88.132119	87.2765	0.8115
9	107.11668	164.0239	1.0373
		$O(k^6 h^4)$	$O(k^3 h^4)$

The error in E_k is typically of the form $O(h^p k^q)$

Shooting methods



- Reducing the solution of the boundary value problem (BVP) to the solution of (one or) two initial value problems (IVPs).
- The IVPs are solved for a succession of trial E values which are adjusted till y_L, y'_L and y_R, y'_R match in the matching point x_m .

Shooting methods

The challenge

- uniform approximation over the whole eigenvalue spectrum
- large steps even for high eigenvalues

Shooting methods

The challenge

- uniform approximation over the whole eigenvalue spectrum
- large steps even for high eigenvalues

How to propagate y_L, y'_L and y_R, y'_R ?

- Shooting methods based on a standard initial-value library code? Some typical difficulties occur.
 - ▷ e.g. Runge Kutta (RK) method
 - ▷ based on piecewise polynomial approximation of the exact solution.
 - ▷ step-size restriction when solving for higher eigenvalues or when the potential is particularly large
 - ▷ not suited for computing a large set of eigenvalues

Shooting methods

The challenge

- uniform approximation over the whole eigenvalue spectrum
- large steps even for high eigenvalues

How to propagate y_L, y'_L and y_R, y'_R ?

- Special techniques are needed which adequately take into account the oscillatory character of the solution.
- **Coefficient Approximation**: the given diff. eq. is replaced (piecwisely) by a diff. eq. which can be solved exactly.

Approximation of
the **solution**
(classical methods)



Approximation of
the **equation**
(coefficient approximation)

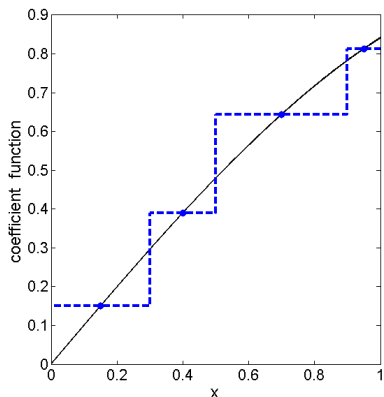
Coefficient approximation methods

$$-(p(x)y'(x))' + q(x)y(x) = Ew(x)y(x)$$



$$-(\tilde{p}(x)y'(x))' + \tilde{q}(x)y(x) = E\tilde{w}(x)y(x)$$

p, q, w are replaced
piecewisely by low
degree polynomials



Pruess methods

- piecewise constant midpoint approximation of the coefficient functions
- Fortran codes:
 - ▶ SLEDGE (Pruess/Fulton)
 - ▶ SL02F (Marletta/Pryce)

Coefficient approximation methods

Pruess method

- Let p , q and w have constant values p_i , q_i , w_i in the interval (x_{i-1}, x_i) :

$$-(p_i(x)y'(x))' + q_i(x)y(x) = Ew_i(x)y(x)$$

- The solution over $[x_{i-1}, x_i]$ is then advanced by the relation

$$\begin{pmatrix} y(x_i) \\ p(x_i)y'(x_i) \end{pmatrix} = \begin{pmatrix} \xi(Z_i) & h_i\eta_0(Z_i) \\ Z_i\eta_0(Z_i)/h_i & \xi(Z_i) \end{pmatrix} \begin{pmatrix} y(x_{i-1}) \\ p(x_{i-1})y'(x_{i-1}) \end{pmatrix}$$

with $Z_i = [(q_i - Ew_i)/p_i]h_i^2$:

$$\xi(Z) = \begin{cases} \cos(|Z|^{1/2}) & Z \leq 0 \\ \cosh(Z^{1/2}) & Z > 0 \end{cases}, \quad \eta_0(Z) = \begin{cases} \sin(|Z|^{1/2})/|Z|^{1/2} & Z < 0 \\ \sinh(Z^{1/2})/Z^{1/2} & Z > 0 \\ 1 & Z = 0 \end{cases}$$

Coefficient approximation methods

Advantages

w.r.t. matrix methods:

- 1 The approximating problem has, like the original, an infinite spectrum.
- 2 The accuracy is maintained or even improves when k increases.

w.r.t. shooting methods based on a standard initial-value solver:

- 1 The mesh is fixed and the coefficients are evaluated before the start of the shooting process. (= big speed advantage).
- 2 relatively unaffected by instability (no stepsize restrictions when solving for large eigenvalues).

Coefficient approximation methods

Advantages

w.r.t. matrix methods:

- 1 The approximating problem has, like the original, an infinite spectrum.
- 2 The accuracy is maintained or even improves when k increases.

w.r.t. shooting methods based on a standard initial-value solver:

- 1 The mesh is fixed and the coefficients are evaluated before the start of the shooting process. (= big speed advantage).
- 2 relatively unaffected by instability (no stepsize restrictions when solving for large eigenvalues).

Drawback

Pruess methods are of low order

Coefficient approximation methods

How to construct higher order coefficient approximation methods?

- 1 Piecewise Perturbation Methods (Neumann series methods)
- 2 Magnus series methods

Coefficient approximation methods

How to construct higher order coefficient approximation methods?

- 1 Piecewise Perturbation Methods (Neumann series methods)
- 2 Magnus series methods

We consider the problem in its Liouville normal form $y'' = (V(x) - E)y$
or

$$Y'(x) = \begin{pmatrix} 0 & 1 \\ V(x) - E & 0 \end{pmatrix} Y(x), \quad Y(x)^T = [y(x), y'(x)]$$

- The eigenvalues are determined using a shooting procedure
- Approximations to Y_L and Y_R are obtained using **1**

2

Outline

- 1 Introduction
- 2 Numerical computation of the eigenvalues
- 3 Piecewise Perturbation methods**
 - Constant Perturbation Method (CPM)
 - Shooting for eigenvalues
- 4 Modified integral series methods
- 5 Conclusion

Piecewise Perturbation methods

Mathematical physics \rightarrow perturbation approximation

- The given diff. eq. is replaced (piecewisely) by reference diff. eq. which can be solved exactly
- The deviation of the solution of the reference eq. from the solution of the original eq. is estimated by means of the perturbation theory (gives us some **correction terms**)

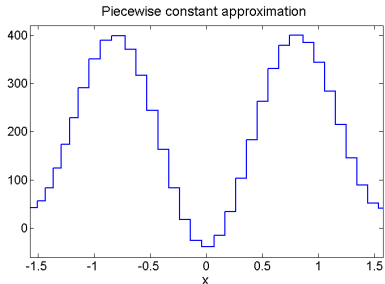
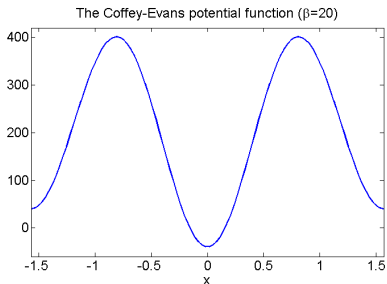
Piecewise Perturbation methods

Mathematical physics \rightarrow perturbation approximation

- The given diff. eq. is replaced (piecewisely) by reference diff. eq. which can be solved exactly
- The deviation of the solution of the reference eq. from the solution of the original eq. is estimated by means of the perturbation theory (gives us some **correction terms**)

- **CPM**: potential function of the reference eq. is piecewise constant
 - ▶ L.Gr. Ixaru, H. De Meyer, G.Vanden Berghe, JCAM 88 (1997).
 - ▶ L.Gr. Ixaru, H. De Meyer, G. Vanden Berghe, CPC 118 (1999).
 - ▶ V. L., M.Van Daele, G.Vanden Berghe, CPC 162 (2004).
 - ▶ ...
- **LPM**: potential function of the reference eq. is piecewise linear
 - ▶ V. L., M. Rizea, L. Ixaru, G. Vanden Berghe, M. Van Daele, CPC 175 (2006).
 - ▶ L. Ixaru, CPC 177 (2007).

Solution of the Schrödinger problem using CPM



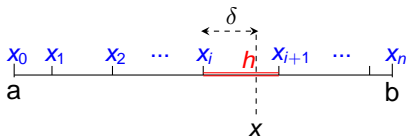
The initial value problem

$$\begin{bmatrix} y(x) \\ y'(x) \end{bmatrix}' = \begin{bmatrix} 0 & 1 \\ V(x) - E & 0 \end{bmatrix} \begin{bmatrix} y(x) \\ y'(x) \end{bmatrix}, \quad y(a) = \alpha, y'(a) = \beta$$

- E : constant
- Aim: construct a CPM for the calculation (propagation) of y, y' .

Solution of the Schrödinger problem using CPM

current interval $[x_i, x_{i+1}]$ with steplength h , introduce $\delta = x - x_i \in [0, h]$



Propagators

$u(\delta)$ and $v(\delta)$ are the solutions of

$$y''(\delta) = (V(x_i + \delta) - E) y(\delta)$$

with the initial values

$$u(0) = 1, u'(0) = 0 \quad \text{and} \quad v(0) = 0, v'(0) = 1.$$

Solution of the Schrödinger problem using CPM

Propagators

- Forward propagation

$$\begin{bmatrix} y(x_{i+1}) \\ y'(x_{i+1}) \end{bmatrix} = \begin{bmatrix} u(h) & v(h) \\ u'(h) & v'(h) \end{bmatrix} \begin{bmatrix} y(x_i) \\ y'(x_i) \end{bmatrix}$$

- Backward propagation

$$\begin{bmatrix} y(x_i) \\ y'(x_i) \end{bmatrix} = \begin{bmatrix} v'(h) & -v(h) \\ -u'(h) & u(h) \end{bmatrix} \begin{bmatrix} y(x_{i+1}) \\ y'(x_{i+1}) \end{bmatrix}$$

- Knowledge of u , v , u' and v' is sufficient to advance the solution in both directions.

Solution of the Schrödinger problem using CPM

How to compute the propagators?

$u(\delta)$ and $v(\delta)$ are constructed via perturbation:

$$y''(\delta) = (V(x_i + \delta) - E) y(\delta) \quad (1)$$

is approximated over $[x_i, x_{i+1}]$ by

$$y''(\delta) = (\bar{V} - E) y(\delta) \quad (2)$$

- The 2 indep. solutions of (2) have known analytic forms: $u_0(\delta)$ and $v_0(\delta)$.

Solution of the Schrödinger problem using CPM

How to compute the propagators?

$u(\delta)$ and $v(\delta)$ are constructed via perturbation:

$$y''(\delta) = (V(x_i + \delta) - E) y(\delta) \quad (1)$$

is approximated over $[x_i, x_{i+1}]$ by

$$y''(\delta) = (\bar{V} - E) y(\delta) \quad (2)$$

- The 2 indep. solutions of (2) have known analytic forms: $u_0(\delta)$ and $v_0(\delta)$.
- The desired $u(\delta)$ and $v(\delta)$ of (1) are generated by adding corrections to $u_0(\delta)$ and $v_0(\delta)$:

$$\begin{aligned}
 u(\delta) &= u_0(\delta) + u_1(\delta) + u_2(\delta) + \dots \leftarrow \Delta V(x_i + \delta) = V(x_i + \delta) - \bar{V} \\
 v(\delta) &= v_0(\delta) + v_1(\delta) + v_2(\delta) + \dots \text{correction terms}
 \end{aligned}$$

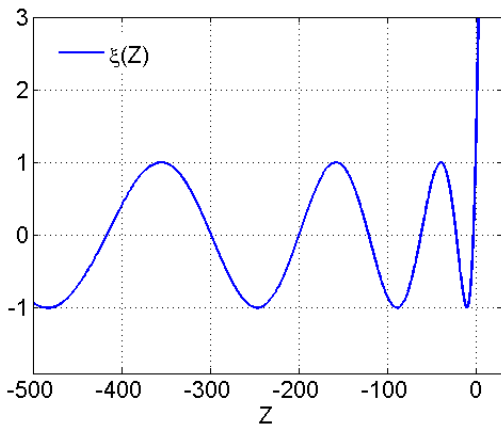
Solution of the Schrödinger problem using CPM

Reference propagators

u_0 solution of $y''(\delta) = (\bar{V} - E) y(\delta)$ with $u_0(0) = 1$, $u_0'(0) = 0$.

$$u_0(Z(\delta)) = \xi(Z) = \begin{cases} \cos(|Z|^{1/2}) & \text{if } Z \leq 0 \\ \cosh(Z^{1/2}) & \text{if } Z > 0 \end{cases}$$

with $Z(\delta) = (\bar{V} - E) \delta^2$.



Solution of the Schrödinger problem using CPM

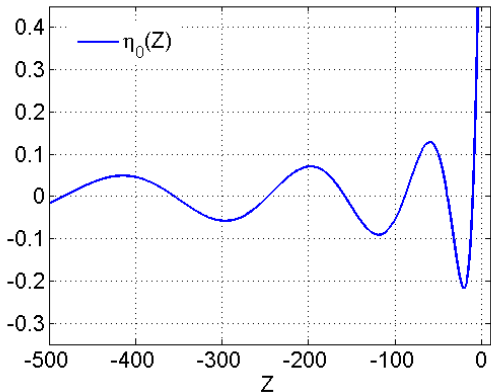
Reference propagators

v_0 solution of $y''(\delta) = (\bar{V} - E) y(\delta)$ with $v_0(0) = 0$, $v_0'(0) = 1$.

$$v_0(Z(\delta))/\delta = \eta_0(Z)$$

$$= \begin{cases} \sin(|Z|^{1/2})/|Z|^{1/2} & \text{if } Z < 0 \\ 1 & \text{if } Z = 0 \\ \sinh(Z^{1/2})/Z^{1/2} & \text{if } Z > 0 \end{cases}$$

$$\text{with } Z(\delta) = (\bar{V} - E) \delta^2.$$



Solution of the Schrödinger problem using CPM

$$\begin{bmatrix} y(x_{i+1}) \\ y'(x_{i+1}) \end{bmatrix} = \begin{bmatrix} u(h) & v(h) \\ u'(h) & v'(h) \end{bmatrix} \begin{bmatrix} y(x_i) \\ y'(x_i) \end{bmatrix}$$

$$\begin{aligned} \begin{bmatrix} u(h) & v(h) \\ u'(h) & v'(h) \end{bmatrix} &= \begin{bmatrix} u_0(h) & v_0(h) \\ u'_0(h) & v'_0(h) \end{bmatrix} + \begin{bmatrix} u_1(h) & v_1(h) \\ u'_1(h) & v'_1(h) \end{bmatrix} + \begin{bmatrix} u_2(h) & v_2(h) \\ u'_2(h) & v'_2(h) \end{bmatrix} + \dots \\ &= \begin{bmatrix} \xi(Z) & \eta_0(Z)/h \\ Z\eta_0(Z)/h & \xi(Z) \end{bmatrix} + \begin{bmatrix} u_1(h) & v_1(h) \\ u'_1(h) & v'_1(h) \end{bmatrix} + \begin{bmatrix} u_2(h) & v_2(h) \\ u'_2(h) & v'_2(h) \end{bmatrix} + \dots \end{aligned}$$

Solution of the Schrödinger problem using CPM

$$\begin{bmatrix} y(x_{i+1}) \\ y'(x_{i+1}) \end{bmatrix} = \begin{bmatrix} u(h) & v(h) \\ u'(h) & v'(h) \end{bmatrix} \begin{bmatrix} y(x_i) \\ y'(x_i) \end{bmatrix}$$

$$\begin{aligned} \begin{bmatrix} u(h) & v(h) \\ u'(h) & v'(h) \end{bmatrix} &= \begin{bmatrix} u_0(h) & v_0(h) \\ u'_0(h) & v'_0(h) \end{bmatrix} + \begin{bmatrix} u_1(h) & v_1(h) \\ u'_1(h) & v'_1(h) \end{bmatrix} + \begin{bmatrix} u_2(h) & v_2(h) \\ u'_2(h) & v'_2(h) \end{bmatrix} + \dots \\ &= \underbrace{\begin{bmatrix} \xi(Z) & \eta_0(Z)/h \\ Z\eta_0(Z)/h & \xi(Z) \end{bmatrix}}_{\text{Pruess Method}} + \underbrace{\begin{bmatrix} u_1(h) & v_1(h) \\ u'_1(h) & v'_1(h) \end{bmatrix} + \begin{bmatrix} u_2(h) & v_2(h) \\ u'_2(h) & v'_2(h) \end{bmatrix}}_{\text{Higher order schemes}} + \dots \end{aligned}$$

- The accuracy increases with the number of correction terms included.

- How to compute correction $P_q = \begin{bmatrix} u_q(h) & v_q(h) \\ u'_q(h) & v'_q(h) \end{bmatrix}$?

Solution of the Schrödinger problem using CPM

Calculation of the corrections

$$P_q = \begin{bmatrix} u_q(h) & v_q(h) \\ u'_q(h) & v'_q(h) \end{bmatrix}$$

- The correction matrix P_q ($q=1,2,\dots$) satisfies the equation

$$P'_q = \begin{bmatrix} 0 & 1 \\ (\bar{V} - E) & 0 \end{bmatrix} P_q + \begin{bmatrix} 0 & 0 \\ \Delta V(\delta) & 0 \end{bmatrix} P_{q-1}, \quad P_q(0) = 0$$

Solution of the Schrödinger problem using CPM

Calculation of the corrections

$$P_q = \begin{bmatrix} u_q(h) & v_q(h) \\ u'_q(h) & v'_q(h) \end{bmatrix}$$

- The correction matrix P_q ($q=1,2,\dots$) satisfies the equation

$$P'_q = \begin{bmatrix} 0 & 1 \\ (\bar{V} - E) & 0 \end{bmatrix} P_q + \begin{bmatrix} 0 & 0 \\ \Delta V(\delta) & 0 \end{bmatrix} P_{q-1}, \quad P_q(0) = 0$$

- If $\Delta V(\delta)$ is a polynomial, the correction terms can be calculated analytically.
- Therefore $V(x_i + \delta)$ is approximated by a polynomial of degree N :
 $V(x_i + \delta) \approx \sum_{n=0}^N V_n h^n P_n^*(\delta/h)$ then $\bar{V} = V_0$ and
 $\Delta V(\delta) = \sum_{n=1}^N V_n h^n P_n^*(\delta/h)$

- Least squares: $V_n = \frac{2n+1}{h^{n+1}} \int_0^h V(x_i + \delta) P_n^*(\delta/h) d\delta$

Solution of the Schrödinger problem using CPM

```

> # construct u1(delta) :
C[1,0]:=delta->integrate(DV(x),x=0..delta)/2:
for m from 1 to Hmax/2+1 do
  C[1,m]:=unapply(simplify(-1/2/delta^m*int(delta1^(m-1)*
    diff(C[1,m-1](delta1),delta1$2),delta1=0..delta)),delta):
od:
u:=unapply(xi(delta)+sum(C[1,k](delta)*delta^(2*k+1)*eta[k],
  k=0..(Hmax/2+1)),delta):
up:=unapply(Z*eta[0]+C[1,0](delta)*xi+
  sum(simplify(diff(C[1,k](delta),delta$1)+delta*C[1,k+1](delta))
  delta^(2*k+1)*eta[k],k=0..(Hmax/2)),delta):

```

Calculation of the

The expressions of the corrections are computed
in a symbolic software package (Maple)

```

> # calculate
for i from
# --> REDU
for m fr_____
  R[i,m]:=unapply(REDUCE(expand(DV(delta)*C[i-1,m](delta)),
    Hmax-2*m+3),delta):
od:
C[i,0]:=delta->0:
for m from 1 to Hmax/2+1 do
  C[i,m]:=unapply(simplify(1/2/delta^m*int(delta1^(m-1)*
    (R[i,m-1](delta1)-diff(expand(C[i,m-1](delta1)),delta1$2)),
    delta1=0..delta)),delta):
od:
u:=unapply(u(delta)+sum(C[i,k](delta)*delta^(2*k+1)*eta[k],
  k=0..(Hmax/2+1)),delta):

```

Solution of the Schrödinger problem using CPM

Calculation of the eigenvalues

Algorithm 1: The shooting procedure

1. Choose a meshpoint x_m ($0 \leq m \leq n$) as the matching point.
2. Set up initial values for y_L, y'_L satisfying the BC at a and initial values for y_R, y'_R satisfying the BC at b . Choose a trial value for E .
3. **repeat**
4. **for** $i = 1$ to m
5.
$$\begin{bmatrix} y_L(x_i) \\ y'_L(x_i) \end{bmatrix} = \begin{bmatrix} u(\delta) & v(\delta) \\ u'(\delta) & v'(\delta) \end{bmatrix} \begin{bmatrix} y_L(x_{i-1}) \\ y'_L(x_{i-1}) \end{bmatrix}$$
6. **end for**
7. **for** $i = n$ down to $m + 1$
8.
$$\begin{bmatrix} y_R(x_{i-1}) \\ y'_R(x_{i-1}) \end{bmatrix} = \begin{bmatrix} v'(\delta) & -v(\delta) \\ -u'(\delta) & u(\delta) \end{bmatrix} \begin{bmatrix} y_R(x_i) \\ y'_R(x_i) \end{bmatrix}$$
9. **end for**
10. Form mismatch function $\phi(E)$ by comparing $y_L(x_m), y'_L(x_m)$ with $y_R(x_m), y'_R(x_m)$.
11. Adjust E to solve the eq. $\phi(E) = 0$. Newton iteration: $E_{t+1} = E_t - \phi(E_t)/\phi'(E_t)$
12. **until** E sufficiently accurate

Solution of the Schrödinger problem using CPM

MATSLISE

- Matlab package for the numerical solution of Sturm-Liouville and Schrödinger problems.
- Based on high order CPM.
- V. L., M. Van Daele and G. Vanden Berghe, ACM Trans. on Math. Soft. 31 (2005).

The screenshot shows the 'Coffey_Evans' software window with the following settings:

- Input**
 - 1. Potential: $V(x) = -2*B*\cos(2*x)+B^2*\sin(2*x)^2$
 - 2. Integration interval: $a = -\pi/2$, $b = \pi/2$
 - 3. Boundary Conditions:
 - 1 $y(a) +$ 0 $y'(a) = 0$
 - 1 $y(b) +$ 0 $y'(b) = 0$
 - 4. Parameter:
 - parameter name(s) = B
 - parameter value(s) = 30
 - 5. Accuracy: $tol = 1e-12$
- Buttons: Construct, Stop, Plot potential, Exit

Solution of the Schrödinger problem using CPM

Input

Indices between

and

Eigenvalues between

and

Calculate eigenvalues >>

Results

Index	Eigenvalue	Estimated Error
0	-0.000000000011807	-1.1e-11
1	117.946307662044530	-2.4e-11
2	231.664929237114250	-7.3e-12
3	231.664929312956250	1.7e-13
4	231.664929388782240	-8.3e-12
5	340.888299809583540	-3.0e-11
6	445.283089582387330	-4.7e-11
7	445.283172306628390	-4.4e-11
8	445.283255031283030	-2.7e-11
9	544.418385149309530	-3.3e-12
10	637.682249873982190	-6.2e-11

Select All Plot Eigenfunction Workspace

Succeeded -- calculated 11 eigenvalues in 0.75 (s) time Help Close

$$-y'' + \left(-2\beta \cos 2x + \beta^2 \sin^2 2x\right) y = Ey, \quad x \in [-\pi/2, \pi/2], \quad \beta = 30$$

Input tolerance = 10^{-12} (37 steps)

Solution of the Schrödinger problem using CPM

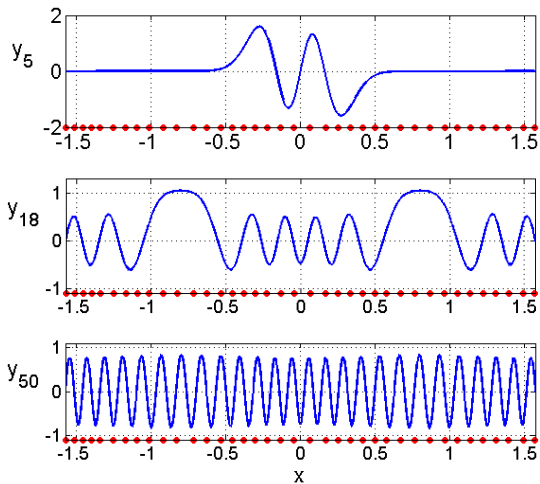
The screenshot shows a software window titled "Coffey_Evans: eigenvalues". The interface is divided into an "Input" section on the left and a "Results" section on the right. In the "Input" section, the "Indices between" radio button is selected, with input fields containing "100" and "110". Below it, the "Eigenvalues between" radio button is unselected, with empty input fields. At the bottom of the "Input" section are two buttons: "Calculate eigenvalues >>" and "Stop". The "Results" section contains a table with three columns: "Index", "Eigenvalue", and "Estimated Error". The table lists 11 rows of data. Below the table are four buttons: "Select All", "Plot", "Eigenfunction", and "Workspace". At the bottom of the window, a status bar displays the message "Succeeded -- calculated 11 eigenvalues in 0.56 (s) time". To the right of the status bar are "Help" and "Close" buttons.

Index	Eigenvalue	Estimated Error
100	10653.525435875912000	-1.3e-11
101	10856.476153183336000	-1.5e-11
102	11061.428299500636000	-9.1e-12
103	11268.381820087219000	-1.1e-11
104	11477.336662800288000	2.5e-12
105	11688.292777948102000	-7.3e-12
106	11901.250118153006000	-5.5e-12
107	12116.208638223135000	-5.5e-12
108	12333.168295032476000	-5.5e-12
109	12552.129047408476000	-3.6e-12
110	12773.090856026758000	-7.3e-12

$$-y'' + \left(-2\beta \cos 2x + \beta^2 \sin^2 2x\right) y = Ey, \quad x \in [-\pi/2, \pi/2], \quad \beta = 30$$

Input tolerance = 10^{-12} (37 steps)

Solution of the Schrödinger problem using CPM



Outline

- 1 Introduction
- 2 Numerical computation of the eigenvalues
- 3 Piecewise Perturbation methods
- 4 Modified integral series methods**
 - Neumann methods
 - Magnus methods
- 5 Conclusion

Integral series methods for the Schrödinger problem

The Schrödinger problem $y'' = (V(x) - E)y$ is written as

$$Y'(x) = \begin{pmatrix} 0 & 1 \\ V(x) - E & 0 \end{pmatrix} Y(x) = A(x, E) Y(x),$$

with $Y(x)^T = [y(x), y'(x)]$.

We consider two expansions to solve problems of the form $Y(x)' = A(x) Y(x)$:

- 1 Neumann expansion
- 2 Magnus expansion

Integral series methods for the Schrödinger problem

$$y' = A(t)y, \quad y(0) = y_0$$

- Neumann expansion

$$y(t) = y_0 \left(1 + \int_0^t A(\tau) d\tau + \int_0^t A(\tau) \int_0^\tau A(\tau_1) d\tau_1 d\tau \right. \\ \left. + \int_0^t A(\tau) \int_0^\tau A(\tau_1) \int_0^{\tau_1} A(\tau_2) d\tau_2 d\tau_1 d\tau + \dots \right)$$

- Magnus expansion (\Rightarrow Lie-group method)

$$y(t) = \exp[\Omega(t)]y_0$$

where

$$\Omega(t) = \int_0^t A(\tau) d\tau - \frac{1}{2} \int_0^t \left[\int_0^\tau A(\tau_1) d\tau_1, A(\tau) \right] d\tau \\ + \frac{1}{4} \int_0^t \left[\int_0^\tau \left[\int_0^{\tau_1} A(\tau_2) d\tau_2, A(\tau_1) \right] d\tau_1, A(\tau) \right] d\tau \\ + \frac{1}{12} \int_0^t \left[\int_0^\tau A(\tau_1) d\tau_1, \left[\int_0^\tau A(\tau_2) d\tau_2, A(\tau) \right] \right] d\tau + \dots$$

Using integral series directly ?

Apply Neumann or Magnus integrator directly on $Y'(x) = A(x, E)Y(x)$
where

$$A(x, E) = \begin{pmatrix} 0 & 1 \\ V(x) - E & 0 \end{pmatrix} ?$$

Using integral series directly ?

Apply Neumann or Magnus integrator directly on $Y'(x) = A(x, E)Y(x)$ where

$$A(x, E) = \begin{pmatrix} 0 & 1 \\ V(x) - E & 0 \end{pmatrix} ?$$

Moan (1998) applied a Magnus method directly:

- Poor approximations can be expected for large eigenvalues: the error in a p th-order method grows as $O(h^{p+1}E^{p/2-1})$.
- Truncations of Magnus series have poor convergence properties when A has large norm and is non-oscillatory.

Moan/Niesen (2006): If $\int_0^t \|A(\tau)\| d\tau < \pi$ then the Magnus series converges.

- This finite radius of convergence implies that there is a relation between the max. allowable stepsize and the magnitude of the eigenvalues we are seeking.

Modified schemes

- A. Iserles (2002) / I. Degani and J. Schiff (2006) suggested modified schemes for highly oscillatory equations.
- The coefficient matrix is decomposed into its constant and varying parts $A(x, E) = \bar{A}(E) + \Delta A(x)$:

$$\begin{pmatrix} 0 & 1 \\ V(x) - E & 0 \end{pmatrix} \approx \begin{pmatrix} 0 & 1 \\ \bar{V} - E & 0 \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ \Delta V(x) & 0 \end{pmatrix}, \quad \Delta V(x) = V(x) - \bar{V}$$

The variables are changed locally:

$$Y(E, x, x_i) = e^{(x-x_i)\bar{A}} U(x - x_i), \quad x \in [x_i, x_{i+1}]$$

where U satisfies

$$U'(\delta) = B(\delta)U(\delta), \quad U(0) = Y_i, \quad \delta = x - x_i \in [0, h_i]$$

with

$$B(\delta) = e^{-\delta\bar{A}} \begin{pmatrix} 0 & 0 \\ \Delta V(x_i + \delta) & 0 \end{pmatrix} e^{\delta\bar{A}}.$$

Modified Schemes

$$B(\delta) = e^{-\delta\bar{A}} \begin{pmatrix} 0 & 0 \\ \Delta V(\mathbf{x}_i + \delta) & 0 \end{pmatrix} e^{\delta\bar{A}}.$$

Modified Schemes

$$B(\delta) = e^{-\delta\bar{A}} \begin{pmatrix} 0 & 0 \\ \Delta V(x_i + \delta) & 0 \end{pmatrix} e^{\delta\bar{A}}.$$

or

$$B(\delta) = -\Delta V(x_i + \delta) \begin{pmatrix} \delta\eta_0(Z_{2\delta}) & \frac{1 - \xi(Z_{2\delta})}{2(E - \bar{V})} \\ -\frac{1 + \xi(Z_{2\delta})}{2} & -\delta\eta_0(Z_{2\delta}) \end{pmatrix},$$

where $Z_\gamma = Z(\gamma) = (\bar{V} - E)\gamma^2$ and $\xi(Z)$ and $\eta_0(Z)$ defined as

$$\xi(Z) = \begin{cases} \cos(|Z|^{1/2}) & \text{if } Z \leq 0, \\ \cosh(Z^{1/2}) & \text{if } Z > 0, \end{cases} \quad \eta_0(Z) = \begin{cases} \sin(|Z|^{1/2})/|Z|^{1/2} & \text{if } Z < 0, \\ 1 & \text{if } Z = 0, \\ \sinh(Z^{1/2})/Z^{1/2} & \text{if } Z > 0, \end{cases}$$

Modified schemes

The new linear system $U'(\delta) = B(\delta)U(\delta)$ has some crucial advantages over the original system.

Modified schemes

The new linear system $U'(\delta) = B(\delta)U(\delta)$ has some crucial advantages over the original system.

$$B(\delta) = -\Delta V(\delta) \begin{pmatrix} \delta\eta_0(Z_{2\delta}) & \frac{1 - \xi(Z_{2\delta})}{2(E - \bar{V})} \\ -\frac{1 + \xi(Z_{2\delta})}{2} & -\delta\eta_0(Z_{2\delta}) \end{pmatrix}$$

For $E > \bar{V}$ ($Z < 0$):

- The entries of B are themselves rapidly oscillating functions.
- Solving the transformed equation $U' = B(\delta)U$ by an integral series method involves repeated integration of B .
- Integration is a “smoothing” operator: the amplitude is decreased once the integrand is integrated. Moreover the higher the oscillation, the faster the convergence and the faster the decay in local error.

Modified schemes

Over each interval $[x_i, x_{i+1}]$

- 1 Apply integral series (Neumann or Magnus) on transformed equation $U' = B(\delta)U$.
- 2 Solution Y in $x = x_{i+1}$ is then obtained from

$$Y(E, x_{i+1}) = e^{h_i \bar{A}} U(h_i)$$

Note that $e^{h_i \bar{A}}$ is the known solution of the system with constant potential \bar{V} :

$$\exp \left[h_i \begin{pmatrix} 0 & 1 \\ \bar{V} - E & 0 \end{pmatrix} \right] = \begin{pmatrix} \xi(Z(\delta)) & \delta \eta_0(Z(\delta)) \\ Z(\delta) \eta_0(Z(\delta))/h & \xi(Z(\delta)) \end{pmatrix}$$

Outline

- 1 Introduction
- 2 Numerical computation of the eigenvalues
- 3 Piecewise Perturbation methods
- 4 Modified integral series methods**
 - **Neumann methods**
 - Magnus methods
- 5 Conclusion

Neumann method

Over interval $[x_i, x_{i+1}]$ with step size h :

- 1 Apply Neumann integral series on transformed equation $U' = B(\delta)U$:

$$U(h) = Y_i \left[1 + \int_0^h B(x) dx + \int_0^h \int_0^{x_1} B(x_1) B(x_2) dx_2 dx_1 \right. \\ \left. + \int_0^h \int_0^{x_1} \int_0^{x_2} B(x_1) B(x_2) B(x_3) dx_3 dx_2 dx_1 + \dots \right]$$

Neumann method

Over interval $[x_i, x_{i+1}]$ with step size h :

- 1 Apply Neumann integral series on transformed equation $U' = B(\delta)U$:

$$U(h) = Y_i \left[1 + \int_0^h B(x)dx + \int_0^h \int_0^{x_1} B(x_1)B(x_2)dx_2dx_1 + \int_0^h \int_0^{x_1} \int_0^{x_2} B(x_1)B(x_2)B(x_3)dx_3dx_2dx_1 + \dots \right]$$

- 2 Solution Y in $x = x_{i+1}$ is then obtained from

$$Y_{i+1} = \begin{pmatrix} \xi(Z(h)) & h\eta_0(Z(h)) \\ Z(h)\eta_0(Z(h))/h & \xi(Z(h)) \end{pmatrix} U(h)$$

- Pruess method: $U(h) = Y_i$.
- Constant Perturbation Methods (CPM):
 - ▶ Each extra Neumann term = CPM correction term

CPM as modified Neumann method

$$\begin{aligned} \textcircled{1} \quad U(h) &= Y_i \left[1 + \int_0^h B(x) dx + \int_0^h \int_0^{x_1} B(x_1) B(x_2) dx_2 dx_1 + \dots \right] \\ \textcircled{2} \quad Y_{i+1} &= \begin{pmatrix} \xi(Z(h)) & h\eta_0(Z(h)) \\ Z(h)\eta_0(Z(h))/h & \xi(Z(h)) \end{pmatrix} U(h) = P_0 U(h) \end{aligned}$$

Each extra Neumann term = CPM correction term

- CPM: $Y_{i+1} = (P_0(h) + P_1(h) + P_2(h) + \dots) Y_i$
- $P'_q = \begin{pmatrix} 0 & 1 \\ \bar{V} - E & 0 \end{pmatrix} P_q + \begin{pmatrix} 0 & 0 \\ \Delta V(\delta) & 0 \end{pmatrix} P_{q-1}, \quad P_q(\delta) = 0.$

CPM as modified Neumann method

- 1 $U(h) = Y_i \left[1 + \int_0^h B(x) dx + \int_0^h \int_0^{x_1} B(x_1) B(x_2) dx_2 dx_1 + \dots \right]$
- 2 $Y_{i+1} = \begin{pmatrix} \xi(Z(h)) & h\eta_0(Z(h)) \\ Z(h)\eta_0(Z(h))/h & \xi(Z(h)) \end{pmatrix} U(h) = P_0 U(h)$

Each extra Neumann term = CPM correction term

- CPM: $Y_{i+1} = (P_0(h) + P_1(h) + P_2(h) + \dots) Y_i$
- $P'_q = \begin{pmatrix} 0 & 1 \\ \bar{V} - E & 0 \end{pmatrix} P_q + \begin{pmatrix} 0 & 0 \\ \Delta V(\delta) & 0 \end{pmatrix} P_{q-1}, \quad P_q(\delta) = 0.$
- $\left(\exp \left[-\delta \begin{pmatrix} 0 & 1 \\ \bar{V} - E & 0 \end{pmatrix} \right] P_q \right)' = \exp \left[-\delta \begin{pmatrix} 0 & 1 \\ \bar{V} - E & 0 \end{pmatrix} \right] \begin{pmatrix} 0 & 0 \\ \Delta V(\delta) & 0 \end{pmatrix} P_{q-1}$
- $P_q(\delta) = P_0(\delta) \int_0^\delta P_0^{-1}(s) \begin{pmatrix} 0 & 0 \\ \Delta V(s) & 0 \end{pmatrix} P_{q-1}(s) ds.$

CPM as modified Neumann method

- 1 $U(h) = Y_i \left[1 + \int_0^h B(x) dx + \int_0^h \int_0^{x_1} B(x_1) B(x_2) dx_2 dx_1 + \dots \right]$
- 2 $Y_{i+1} = \begin{pmatrix} \xi(Z(h)) & h\eta_0(Z(h)) \\ Z(h)\eta_0(Z(h))/h & \xi(Z(h)) \end{pmatrix} U(h) = P_0 U(h)$

Each extra Neumann term = CPM correction term

- CPM: $Y_{i+1} = (P_0(h) + P_1(h) + P_2(h) + \dots) Y_i$
- $P_q(\delta) = P_0(\delta) \int_0^\delta P_0^{-1}(s) \begin{pmatrix} 0 & 0 \\ \Delta V(s) & 0 \end{pmatrix} P_{q-1}(s) ds.$
- $P_1(\delta) = P_0(\delta) \int_0^\delta B(s_1) ds_1$
- $P_q(\delta) = P_0(\delta) \int_0^\delta B(s_1) \int_0^{s_1} B(s_2) \dots \int_0^{s_{q-1}} B(s_q) ds_q \dots ds_2 ds_1,$

CPM as modified Neumann method

$$1 \quad U(h) = Y_i \left[1 + \int_0^h B(x) dx + \int_0^h \int_0^{x_1} B(x_1) B(x_2) dx_2 dx_1 + \dots \right]$$

$$2 \quad Y_{i+1} = \begin{pmatrix} \xi(Z(h)) & h\eta_0(Z(h)) \\ Z(h)\eta_0(Z(h))/h & \xi(Z(h)) \end{pmatrix} U(h)$$

with

$$B(\delta) = -\Delta V(\delta) \begin{pmatrix} \delta\eta_0(Z_{2\delta}) & \frac{1 - \xi(Z_{2\delta})}{2(E - \bar{V})} \\ -\frac{1 + \xi(Z_{2\delta})}{2} & -\delta\eta_0(Z_{2\delta}) \end{pmatrix}$$

Quadrature of the (multivariate) integrals?

- multivariate quadrature usually a hard problem
- surprisingly effective and cheap quadrature can be used here:
 - ▶ CPM: replacing ΔV by polynomial
 - ▶ \sim Filon quadrature (see further)
 - ▶ consistent with oscillatory character (for $E > \bar{V}$)

Outline

- 1 Introduction
- 2 Numerical computation of the eigenvalues
- 3 Piecewise Perturbation methods
- 4 Modified integral series methods**
 - Neumann methods
 - Magnus methods**
- 5 Conclusion

The modified Magnus scheme

Over interval $[x_i, x_{i+1}]$ with step size h :

$$1 \quad U(h) = \exp[\Omega(h)] Y_i$$

$$2 \quad \Omega(h) = \int_0^h B(x) dx - \frac{1}{2} \int_0^h \int_0^{x_1} [B(x_2), B(x_1)] dx_2 dx_1 \\ + \frac{1}{4} \int_0^h \int_0^{x_1} \int_0^{x_2} [[B(x_3), B(x_2)], B(x_1)] dx_3 dx_2 dx_1 \\ + \frac{1}{12} \int_0^h \int_0^{x_1} \int_0^{x_2} [B(x_3), [B(x_2), B(x_1)]] dx_3 dx_2 dx_1 + \dots$$

$$3 \quad Y_{i+1} = \begin{pmatrix} \xi(Z(h)) & h\eta_0(Z(h)) \\ Z(h)\eta_0(Z(h))/h & \xi(Z(h)) \end{pmatrix} U(h)$$

- Truncation of Magnus series
- Replacement of integrals by quadrature

The modified Magnus scheme

Over interval $[x_i, x_{i+1}]$ with step size h :

$$\textcircled{1} \quad U(h) = \exp[\Omega(h)] Y_i$$

The exponential can be evaluated explicitly:

- $A, B, \Omega \in \mathfrak{sl}(2)$
- $\Rightarrow \exp[\Omega(h)]$ can be written down explicitly, since

$$\exp \begin{bmatrix} a & b \\ c & -a \end{bmatrix} = \begin{bmatrix} \xi(\omega) + a\eta_0(\omega) & b\eta_0(\omega) \\ c\eta_0(\omega) & \xi(\omega) - a\eta_0(\omega) \end{bmatrix},$$

where

$$\omega = a^2 + bc$$

and a, b, c functions of x and E .

Truncation of the Magnus series

A modified Magnus scheme $Y_{i+1} = \exp[h\bar{A}] \exp[\Omega(h)] Y_i$ with

$$\bar{V} = \frac{1}{h} \int_0^h V(x_i + \delta) d\delta$$

- order 4:

$$\Omega(h) = \int_0^h B(x) dx$$

- order 8

$$\Omega(h) = \int_0^h B(x) dx - \frac{1}{2} \int_0^h \int_0^{x_1} [B(x_2), B(x_1)] dx_2 dx_1$$

- order 10

$$\begin{aligned} \Omega(h) = & \int_0^h B(x) dx - \frac{1}{2} \int_0^h \int_0^{x_1} [B(x_2), B(x_1)] dx_2 dx_1 \\ & + \frac{1}{4} \int_0^h \int_0^{x_1} \int_0^{x_2} [[B(x_3), B(x_2)], B(x_1)] dx_3 dx_2 dx_1 \\ & + \frac{1}{12} \int_0^h \int_0^{x_1} \int_0^{x_2} [B(x_3), [B(x_2), B(x_1)]] dx_3 dx_2 dx_1 \end{aligned}$$

Replacement of the integrals by quadrature

Quadrature of the integrals: univariate case

- Consider the computation of the integral $\int_0^h B(x) dx$
- classical Gauss-Christoffel?
 - ▷ Basic idea: $B(x)$ is replaced by (interpolating) polynomial
 - ▷ Leads to scheme of the form

$$\int_0^h B(x) dx \approx h \sum_{l=1}^{\nu} b_l B(c_l h)$$

with c_1, c_2, \dots, c_{ν} quadrature nodes (e.g. zeros of the ν th Legendre polynomial)

Replacement of the integrals by quadrature

Quadrature of the integrals: univariate case

- Consider the computation of the integral $\int_0^h B(x) dx$
- classical Gauss-Christoffel?
 - ▷ Basic idea: $B(x)$ is replaced by (interpolating) polynomial
 - ▷ Leads to scheme of the form

$$\int_0^h B(x) dx \approx h \sum_{l=1}^{\nu} b_l B(c_l h)$$

with c_1, c_2, \dots, c_{ν} quadrature nodes (e.g. zeros of the ν th Legendre polynomial)

- For $E \gg \bar{V}$ the matrix function B is highly oscillatory

Replacement of the integrals by quadrature

Quadrature of the integrals: univariate case

$$\int_0^h B(x) dx = - \int_0^h \Delta V(\delta) \begin{pmatrix} \delta \eta_0(Z_{2\delta}) & \frac{1 - \xi(Z_{2\delta})}{2(E - \bar{V})} \\ -\frac{1 + \xi(Z_{2\delta})}{2} & -\delta \eta_0(Z_{2\delta}) \end{pmatrix} d\delta, \quad Z_{2\delta} = 4(\bar{V} - E)\delta^2$$

- Special quadrature methods for highly-oscillating integrands should be used.
- e.g. Filon quadrature delivers accuracy which improves with higher oscillation. (A. Iserles, S. Nørsett,...)
 - ▷ Basic idea: $\Delta V(\delta)$ is replaced by (interpolating) polynomial
 - ▷ Leads to scheme of the form

$$\int_0^h B(x) dx \approx h \sum_{l=1}^{\nu} b_l(\omega) \Delta V(c_l h)$$

- ▷ b_l depends now on $\omega = \bar{V} - E$.

Replacement of the integrals by quadrature

Quadrature of the integrals: multivariate case

- same idea: $\Delta V(\delta)$ is replaced by (interpolating) polynomial
- leads to schemes of the form

$$\int_0^h \int_0^{\delta_1} [B(\delta_2), B(\delta_1)] d\delta_2 d\delta_1 = h^2 \sum_{k=1}^{\nu} \sum_{l=1}^{\nu} b_{k,l}(\omega) \Delta V(c_k h) \Delta V(c_l h)$$

- Note that the values of ΔV that have been already evaluated for the quadrature of the univariate integral are reused and nothing else.

Note that in the nonoscillatory region ($E < \bar{V}$) a Filon-Legendre method is just as good as a Gauss-Legendre method with the same nodes. And in the oscillatory region ($E > \bar{V}$) the error of the Filon-Legendre method is $O(1/\sqrt{E - \bar{V}})$.

Example: method of order 10

$$\begin{aligned}\Omega(h) &= \int_0^h B(x)dx - \frac{1}{2} \int_0^h \int_0^{x_1} [B(x_2), B(x_1)]dx_2dx_1 \\ &\quad + \frac{1}{4} \int_0^h \int_0^{x_1} \int_0^{x_2} [[B(x_3), B(x_2)], B(x_1)]dx_3dx_2dx_1 \\ &\quad + \frac{1}{12} \int_0^h \int_0^{x_1} \int_0^{x_2} [B(x_3), [B(x_2), B(x_1)]]dx_3dx_2dx_1\end{aligned}$$

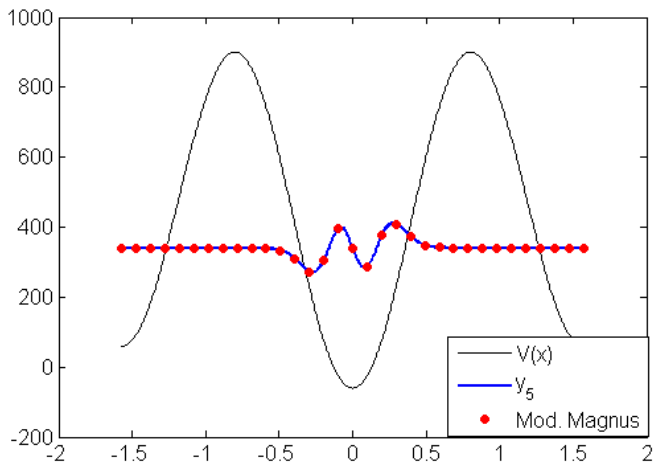
- replace $\Delta V(\delta)$ by interpolating polynomial through ν Legendre points

- or CPM-like: $V(x) \approx \sum_{s=0}^{\nu-1} V_s h^s P_s^*(\delta/h)$ with

$$V_s = \frac{(2s+1)}{h^{s+1}} \int_0^h V(x_i + \delta) P_s^*(\delta/h) d\delta.$$

- then $\bar{V} = V_0$ and $\Delta V(\delta) \approx \sum_{s=1}^{\nu-1} V_s h^s P_s^*(\delta/h)$.
- tenth-order Gauss-Legendre is used, requiring $\nu = 5$ function evaluations of V

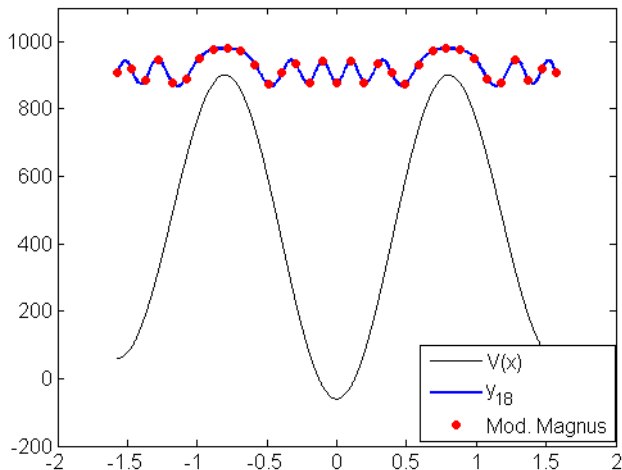
Results: Coffey-Evans problem



$$h = \pi/32$$

$$-y'' + \left(-2\beta \cos 2x + \beta^2 \sin^2 2x\right) y = Ey, \quad x \in [-\pi/2, \pi/2], \quad \beta = 30$$

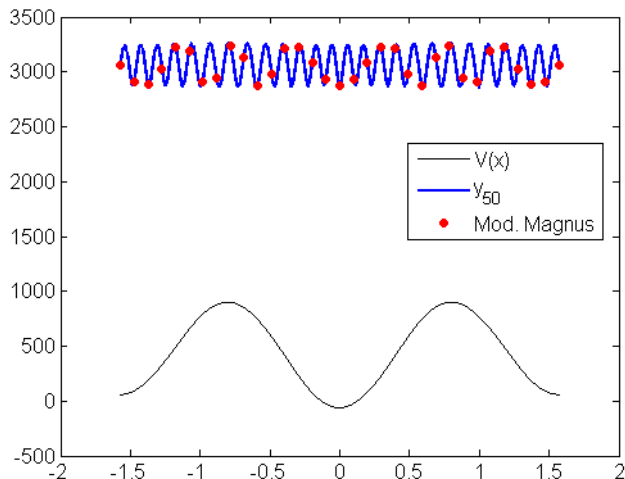
Results: Coffey-Evans problem



$$h = \pi/32$$

$$-y'' + \left(-2\beta \cos 2x + \beta^2 \sin^2 2x\right) y = Ey, \quad x \in [-\pi/2, \pi/2], \quad \beta = 30$$

Results: Coffey-Evans problem



$$h = \pi/32$$

$$-y'' + \left(-2\beta \cos 2x + \beta^2 \sin^2 2x\right) y = Ey, \quad x \in [-\pi/2, \pi/2], \quad \beta = 30$$

Shooting for eigenvalues

A Sturm-Liouville solver based on a modified Magnus method

1. Use stepsize selection algorithm to construct mesh $a = x_0 < x_1 < \dots < x_n = b$
2. **for** $i = 1$ to n
3. Compute \bar{V} and V_s , $s = 1, \dots, 4$ for the i th interval.
4. **end for**
5. Choose a meshpoint x_m ($0 \leq m \leq n$) as the matching point.
6. Set up initial values for Y_L satisfying the BC at a and initial values for Y_R satisfying the BC at b . Choose a trial value for E .
7. **repeat**
8. **for** $i = 0$ to $m - 1$
9. $Y_L(x_{i+1}) = e^{h_i \bar{A}} e^{\Omega(h_i)} Y_L(x_i)$
10. **end for**
11. **for** $i = n$ down to $m + 1$
12. $Y_R(x_{i-1}) = e^{-h_i \bar{A}} e^{-\Omega(h_i)} Y_R(x_i)$
13. **end for**
14. Adjust E by comparing $Y_L(x_m)$ with $Y_R(x_m)$.
15. **until** E sufficiently accurate

Results: Coffey-Evans problem

Coffey-Evans problem			
k	E_k	$n = 128$	$n = 256$
0	0.0000000000000000	3.4E-10	2.2E-13
1	117.9463076620687587	1.5E-9	1.4E-12
2	231.6649292371271088	2.1E-9	1.1E-12
3	231.6649293129610125	1.1E-9	1.1E-12
4	231.6649293887949167	2.1E-9	7.9E-13
5	340.8882998096130157	4.5E-9	4.4E-12
6	445.2830895824354620	4.4E-9	3.6E-12
8	445.2832550313310036	4.4E-9	2.7E-12
10	637.6822498740469991	4.8E-9	4.2E-12
15	802.4787986926240517	2.8E-9	1.7E-12
20	951.8788067965913828	3.3E-9	3.7E-12

Table: Absolute value of (absolute) errors ΔE_k for the Coffey-Evans. n is the number of (equidistant) steps. $aE-b$ means $a \cdot 10^{-b}$.

Modified Neumann \leftrightarrow Modified Magnus

- second order method

- ▶ Pruess method, CPM(0): $Y_{i+1} = e^{h\bar{A}} Y_i$

- fourth order method

- ▶ Mod. Neumann, CPM(1):

$$Y_{i+1} = e^{h\bar{A}} \left(I + \int_0^h B(x) dx \right) Y_i$$

- ▶ Mod. Magnus:

$$Y_{i+1} = e^{h\bar{A}} e^{\int_0^h B(x) dx} Y_i$$

- ▶ same accuracy is reached, evaluation of matrix evaluation requires only very little extra time:

$$\exp \begin{bmatrix} a & b \\ c & -a \end{bmatrix} = \begin{bmatrix} \xi(\omega) + a\eta_0(\omega) & b\eta_0(\omega) \\ c\eta_0(\omega) & \xi(\omega)a\eta_0(\omega) \end{bmatrix}, \text{ where } \omega = a^2 + bc$$

Modified Neumann \leftrightarrow Modified Magnus

- eighth order method

- ▶ Mod. Neumann:

$$Y_{i+1} = e^{h\bar{A}} \left(I + \int_0^h B(x) dx + \int_0^h \int_0^{x_1} B(x_1) B(x_2) dx_2 dx_1 \right) Y_i$$

- ▶ Mod. Magnus: $Y_{i+1} = e^{h\bar{A}} e^{\Omega(h)} Y_i$ with

$$\Omega(h) = \int_0^h B(x) dx - \frac{1}{2} \int_0^h \int_0^{x_1} [B(x_2), B(x_1)] dx_2 dx_1$$

- ▶ Again amount of work and accuracy very similar

Modified Neumann \leftrightarrow Modified Magnus

- eighth order method

- ▶ Mod. Neumann:

$$Y_{i+1} = e^{h\bar{A}} \left(I + \int_0^h B(x) dx + \int_0^h \int_0^{x_1} B(x_1) B(x_2) dx_2 dx_1 \right) Y_i$$

- ▶ Mod. Magnus: $Y_{i+1} = e^{h\bar{A}} e^{\Omega(h)} Y_i$ with

$$\Omega(h) = \int_0^h B(x) dx - \frac{1}{2} \int_0^h \int_0^{x_1} [B(x_2), B(x_1)] dx_2 dx_1$$

- ▶ Again amount of work and accuracy very similar
- a truncated Neumann expansion does not respect Lie group structure
- For problems with higher dimension, the computation of the matrix exponential $e^{\Omega(h)}$ may be fairly expensive.

Outline

- 1 Introduction
- 2 Numerical computation of the eigenvalues
- 3 Piecewise Perturbation methods
- 4 Modified integral series methods
- 5 Conclusion**

Summary

- For high eigenvalues the solution is severely oscillatory and standard numerical integrators have to advance in small steps.
- However we want:
 - ▶ uniform accuracy over the whole energy-range
 - ▶ large step-sizes, even for high eigenvalues
- This can be realized using coefficient approximation methods in a shooting procedure.
- Integral series integrators (Neumann, Magnus) allow the construction of higher order methods.
 - ▶ These integral series integrators are very effective when they are applied on the “modified” equation ($E > V$ as well as $E < V$).
 - ▶ “Oscillation-proof” quadrature for $E \gg \bar{V}$