

Solving the Schrödinger Equation using Sinc Functions in one and two dimensions, employing Python and Numpy

Obiageli Lovenda Ezenwachukwu and Moritz Braun

Department of Physics, UNISA, PO Box 392, Pretoria 0003

Abstract. In this contribution orthonormal sinc basis function are used to numerically solve the Schrödinger equation in one and two dimensions for a number of potentials. The calculations are done using Python and the numpy module. The convergence is found to be quick and, for the Morse potential, also agrees with the theoretically expected behaviour. In the two dimensional case code optimizations lead to a large speed-up.

1. Introduction

Sinc functions as a basis set have been used extensively for obtaining the approximate solutions of ordinary differential equations, partial differential equations and integral equations [1, 2]. The sinc numerical method is easily implemented and the results obtained are of good accuracy [3, 4]. It has been shown that sinc numerical methods are distinguished by exponentially decaying errors [5, 6]; they have convergence rates of $O\left(\exp\left(-k\sqrt{N}\right)\right)$ with some $k > 0$, [7], where N is the number of nodes or basis functions used.

In this contribution the numerical solution of the Schrödinger equation for a variety of potentials in both one and two dimensions is demonstrated using the scripting language Python[8] together with the module numpy[9].

2. Solving the one dimensional Schrödinger equation using Sinc Basis Functions

The normalized sinc function is on the line $-\infty < x < \infty$ defined by

$$\text{sinc}(x) = \frac{\sin(\pi x)}{\pi x}, \quad \int_{-\infty}^{\infty} \text{sinc}(x) dx = 1. \quad (1)$$

The zeroes of this function are all integer values of x . An orthonormal set of functions can be obtained from the sinc function by shifting it by all positive and negative integers. In order to make this basis set also suitable for the expansion of quickly varying functions, we define the basis functions, also introducing a stretching parameter h , as

$$s_i(x) = \frac{1}{\sqrt{h}} \frac{\sin\left(\pi\left(\frac{x}{h} - i\right)\right)}{\pi\left(\frac{x}{h} - i\right)}. \quad (2)$$

Here h is the distance between subsequent zeroes of these basis functions, i is the index running in principle over all integer values from $-\infty$ to ∞ and the first factor on the right hand side provides for normalization.

We now restrict i to the interval $[-n, n]$ and define the basis functions for numerical calculations by

$$f_j(x) = s_{-n+j-1}(x), \quad (3)$$

with $j = 1, 2, \dots, 2n+1$. This is an orthonormal basis of dimension $N = 2n+1$. Choosing n and h is equivalent to a cut-off at $x_{max} = nh$. This fixes the boundary conditions and determines the accuracy.

Thus the ansatz for the solution of the Schrödinger equation is given by

$$\psi(x) = \sum_{i=1}^N c_i f_i(x). \quad (4)$$

The Schrödinger equation

$$\left\{ -\frac{d^2}{dx^2} + V(x) \right\} \psi_\nu(x) = e_\nu \psi_\nu(x) \quad (5)$$

is now solved via the variational method, leading to the eigenvalue problem

$$H \mathbf{u}_\nu = e_\nu \mathbf{u}_\nu \quad (6)$$

with

$$h_{ij} = k_{ij} + v_{ij}, \quad (7)$$

$$k_{ij} = \int f'_i(x) f'_j(x) dx = \begin{cases} \frac{\pi^2}{3h^2} & i = j; \\ (-1)^{|i-j|} \frac{1}{h^2} \frac{2}{|i-j|^2} & i \neq j. \end{cases} \quad (8)$$

and

$$v_{ij} = \int_{-x_{max}}^{x_{max}} f_i(x) V(x) f_j(x) dx. \quad (9)$$

The analytic expression for the kinetic energy matrix elements has been obtained by using the Fourier expansion of the sinc function and the Parseval theorem. The potential matrix was evaluated by repeated Gauss-Legendre integration, where the intervals are coinciding with the intervals between subsequent zeroes of the sinc basis functions. For this integration we used $N_{GL} = 10$ points and weights in each interval, resulting in a total number of integration points of $20n$.

3. Numerical Results for one Dimension

3.1. Numerical calculations for the one dimensional harmonic oscillator

For verification purposes we first considered the harmonic oscillator. The energy eigenvalues of the quantum harmonic oscillator with the Hamiltonian

$$H = -\frac{d^2}{dx^2} + x^2 \quad (10)$$

are

$$E_\nu = 2\nu + 1, \quad (11)$$

i.e. all odd numbers $1, 3, 5, 7, \dots$.

Employing Python and numpy, and using the sinc basis functions for $x_{max} = 8$, the results of the numerical calculations obtained are shown in table 1, and the lowest eigenvalues $1, 3, 5, 7$ are obtained very accurately. The calculations were done for $nh = x_{max} = 8$. Convergence is seen to be rather quick as function of n .

Table 1. Results of numerical calculations for harmonic oscillator

n	h	E_0	E_1	E_2	E_3
8	1.0	1.00013274618	3.00388708934	5.01965658349	7.1426916954
9	0.8889	1.00001043728	3.00040366538	5.00262094231	7.026576271
10	0.8	1.00000059941	3.00002946274	5.00023921435	7.00326643666
11	0.7273	1.00000002524	3.00000153079	5.00001518651	7.00026666304
12	0.6667	1.00000000078	3.00000005692	5.00000067774	7.00001475924
13	0.6154	1.00000000002	3.00000000152	5.00000002142	7.00000056346
14	0.5714	1.0	3.00000000003	5.00000000048	7.000000015
15	0.5333	1.0	3.0	5.00000000001	7.00000000028
16	0.5	1.0	3.0	5.0	7.0

3.2. Numerical calculations for Morse potential

The Morse potential has been used to model the vibrational excitations of a chemical bond for diatomic molecules. The potential in this model given by[10]

$$V_{\text{Morse}}(x) = D(e^{-2x/a} - 2e^{-x/a}). \quad (12)$$

i.e. the Hamiltonian becomes

$$H = -\frac{1}{2\mu} \frac{d^2}{dx^2} + D(e^{-2x/a} - 2e^{-x/a}) \quad (13)$$

and, for $\mu = 1/2$ and $a = 1$, the eigenvalues are given by[10]

$$E_\nu = -D \left\{ 1 - \frac{1}{\sqrt{D}} \left(\nu + \frac{1}{2} \right) \right\}^2 \quad (14)$$

with ν restricted by the condition, that the contents of the curly bracket must be positive. Employing the Python code for the sinc basis function for $D = 9$ and $x_{\text{max}} = 15$, the results as shown in table 2 are obtained.

From the theoretical convergence studies [7] we expect the following to hold for the ν -th

Table 2. Results of numerical calculations for Morse potential

n	h	E_0	E_1	E_2
10	1.5	-3.76039	-0.25006	0.07333
15	1.0	-5.61145	-1.35904	0.00106
16	0.9375	-5.78207	-1.54867	-0.02423
17	0.88235	-5.91164	-1.70946	-0.05513
18	0.83333	-6.00860	-1.84134	-0.08795
19	0.78947	-6.08010	-1.94651	-0.11947
20	0.75	-6.13196	-2.02827	-0.14810
22	0.68182	-6.19503	-2.13663	-0.19210
25	0.6	-6.23361	-2.21183	-0.22830
27	0.55555	-6.24283	-2.23225	-0.23947
30	0.5	-6.24798	-2.24473	-0.24679

eigenvalue resulting from the calculations as function of N :

$$E_\nu(N) = E_\nu + b_\nu \exp(-c_\nu \sqrt{N}). \quad (15)$$

Gnuplot was used both for creating the figures as well as fitting the results to the expected convergence behaviour. From figure 1 and figure 2 it is evident, that the eigenvalues agree reasonably well with the fits to equation (15).

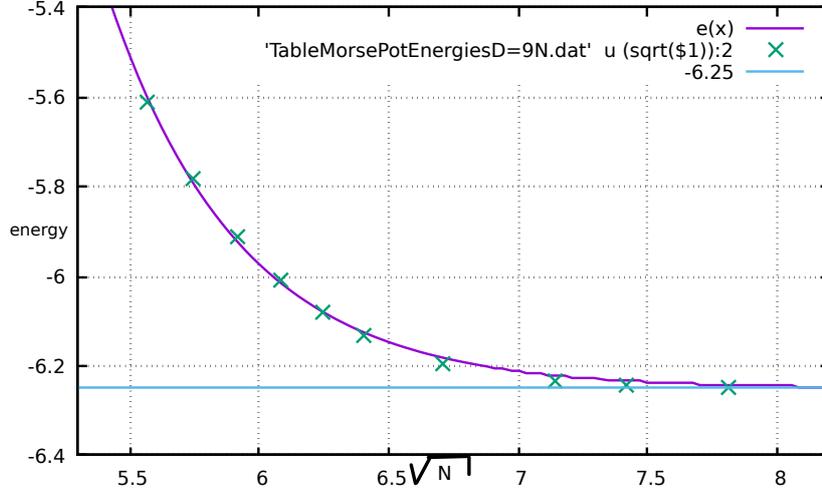


Figure 1. Convergence for ground state of Morse potential

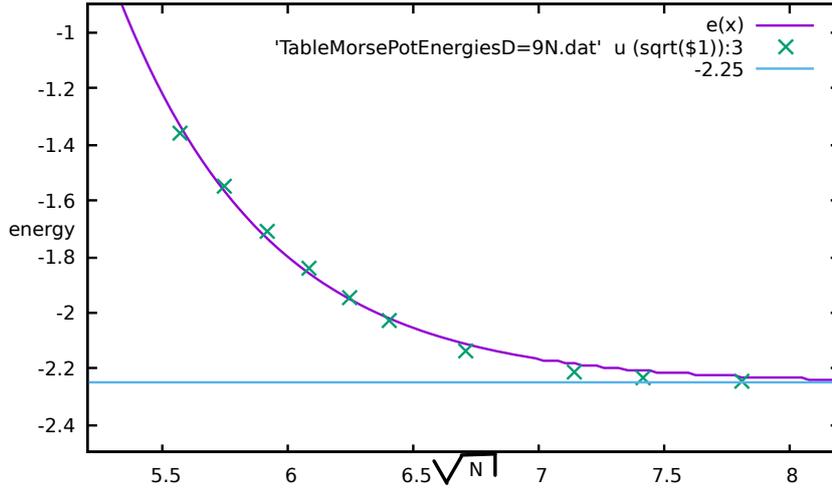


Figure 2. Convergence for first excited state of Morse potential

4. Solving the two dimensional Schrödinger equation

The two dimensional Schrödinger equation was solved numerically by expanding the wave function $\psi(x, y)$ in terms of products of one dimensional basis functions, i.e.

$$\psi(x, y) = \sum c_{\alpha} f_{i_{\alpha}}(x) f_{j_{\alpha}}(y), \quad (16)$$

where i_{α} and j_{α} are suitably defined functions of α . Thus the basis functions used in x and y are identical, the domain of expansion is $[-x_{\max}, x_{\max}] \times [-x_{\max}, x_{\max}]$, and the total number of basis functions becomes N^2 .

In two dimensions the Hamiltonian is given by

$$H = -\frac{\partial^2}{\partial x^2} - \frac{\partial^2}{\partial y^2} + V(x, y). \quad (17)$$

Evaluation of the matrix elements $h_{\alpha\beta}$ then proceeds analytically for the kinetic energy term and numerically for the potential energy term in a similar fashion as in one dimension. It should be noted, that the numerical integration requires much more CPU time, since there are now $4n^2 N_{\text{GL}}^2$ integration points.

5. Numerical Results for two Dimensions

5.1. Results for two dimensional harmonic oscillator

Numerical Calculations were done for $nh = x_{\text{max}} = y_{\text{max}} = 8$ and $N_{\text{GL}} = 10$ and the results are given in table 3

Table 3. Results of Numerical Calculations for 2-D Harmonic Oscillator

n	h	E_0	E_1	E_2	E_3	E_4	E_5
7	1.1429	2.002428	4.027192	4.027192	6.051474	6.099747	6.100884
8	1.0	2.000262	4.003994	4.003994	6.007651	6.019370	6.019519
9	0.8889	2.000021	4.000412	4.000412	6.000794	6.002584	6.002597
10	0.8	2.000001	4.000030	4.000030	6.000058	6.000236	6.000237
11	0.7273	2.000000	4.000002	4.000002	6.000003	6.000015	6.000015
12	0.6667	2.	4.000000	4.000000	6.000000	6.000000	6.000000
13	0.6154	2.	4.	4.	6.	6.000000	6.000000
14	0.5714	2.	4.	4.	6.	6.	6.
15	0.5333	2.	4.	4.	6.	6.	6.

Table 4. Parameters of Numerical Calculations for Harmonic Oscillator for $N_{\text{GL}} = 10$

n	h	N_{EVP}	N_{int}	$T_{\text{Eval.of B.F.}}$	$T_{\text{Eval.of vmat}}$	T_{EVP}
7	1.1429	225	19600	18.5929	1.4714	0.1197
8	1.0	289	25600	31.1685	3.1726	0.1915
9	0.8889	361	32400	48.4563	6.2601	0.3513
10	0.8	441	40000	73.5441	11.5954	0.6294
11	0.7273	529	48400	109.2896	20.1906	1.0501
12	0.6667	625	57600	154.8675	33.5829	1.7290
13	0.6154	729	67600	211.3824	53.5678	2.6635
14	0.5714	841	78400	279.7736	82.5839	3.9894
15	0.5333	961	90000	389.9374	123.8101	5.9517

Table 5. Results of Numerical Calculations for HO for $N_{\text{GL}} = 10$ with code optimization

n	h	N_{EVP}	N_{int}	$T_{\text{Eval.of B.F.}}$	$T_{\text{Eval.of vmat}}$	T_{EVP}
7	1.1429	225	19600	0.1613	1.4806	0.0969
8	1.0	289	25600	0.2341	3.1731	0.1933
9	0.8889	361	32400	0.3343	6.2837	0.3545
10	0.8	441	40000	0.4597	11.5784	0.6382
11	0.7273	529	48400	0.6081	20.1427	1.0403
12	0.6667	625	57600	0.8030	33.3005	1.6976
13	0.6154	729	67600	1.0367	53.4464	2.6106
14	0.5714	841	78400	1.3088	82.2574	3.9483
15	0.5333	961	90000	1.6377	122.7418	5.8455

Table 6. Results of Numerical Calculations for HO for $N_{\text{GL}} = 4$ with code optimization

n	h	N_{EVP}	N_{int}	$T_{\text{Eval.of B.F.}}$	$T_{\text{Eval.of vmat}}$	T_{EVP}	E_0	E_1	E_2	E_3
7	1.1429	225	3136	0.0626	0.2395	0.0954	2.002419	4.027176	4.027176	6.051453
8	1.0	289	4096	0.0934	0.5131	0.1923	2.000261	4.003997	4.003997	6.007657
9	0.8889	361	5184	0.1260	1.0041	0.3553	2.000021	4.000412	4.000412	6.000795
10	0.8	441	6400	0.1699	1.8476	0.6252	2.000001	4.000030	4.000030	6.000058
11	0.7273	529	7744	0.2202	3.2080	1.0613	2.000000	4.000002	4.000002	6.000003
12	0.6667	625	9216	0.2857	5.3602	1.7419	2.	4.000000	4.000000	6.000000
13	0.6154	729	10816	0.3621	8.4907	2.6021	2.	4.	4.	6.
14	0.5714	841	12544	0.4483	13.1560	4.0709	2.	4.	4.	6.
*15	0.5333	961	14400	0.5563	19.5872	5.8239	2.	4.	4.	6.

From table 3 it is evident that there is fast convergence of the energy levels, also showing the expected degeneracy. However from table 4 it is clear, that a lot of time was spent evaluating the basis functions at all integration points. Splitting one time consuming python loop into

two small loops, i.e. unrolling, for code optimization, resulted in the time spent with these evaluations to be vastly reduced, as shown in table 5. In addition, via changing N_{GL} to 4, the time for evaluating the basis functions was reduced even further, while the eigen values remained the same as shown in table 6.

6. Conclusions

Convergence for the energies of ground state, first, second and third excited states was very fast for the harmonic oscillator, irrespective of the dimension. For the Morse potential, the error of the eigenvalues showed good agreement with the theoretically expected behaviour. The two dimensional Python and numpy code was accelerated via unrolling a loop.

Further development of the method and the Python code is currently underway to calculate the ground state of the hydrogen molecular ion H_2^+ and also to extend the code to three dimensions.

Acknowledgments

O.L. Ezenwachukwu acknowledges financial support from UNISA.

References

- [1] Esmail Hesameddini and Elham Asadolahifard. Solving systems of linear volterra integro-differential equations by using Sinc-collocation method. *International Journal of Mathematical Engineering and Science*, 2, July 2013.
- [2] John Lund and Kenneth L. Bowers. *Sinc Methods for Quadrature and Differential Equations*. ISBN, 1992.
- [3] Abbas Saadatmandi Mehdi Dehghan. The numerical solution of a nonlinear system of second-order boundary value problems using the sinc-collocation method. *Mathematical and Computer Modelling, Issue 11-12*, 46:1434–1441, December 2007.
- [4] Jennifer L. Mueller and Thomas S. Shores. A new Sinc-Galerkin method for convection-diffusion equations with mixed boundary conditions. *Mathematical Subject Classification*, 1991.
- [5] Frank Stenger. summary of sinc numerical methods. *Journal of computational and applied mathematics*, 121:379–420, 2000.
- [6] T. Matsuo M. Sugihara. Recent developments of the Sinc numerical methods. *Journal of Computational and Applied Mathematics*, pages 164–165, 673 – 689, April 2003.
- [7] Frank Stenger. Numerical Methods based on Whittaker Cardinal, or Sinc Functions. *Society for Industrial and Applied Mathematics*, 23(2):165–224, April 1981.
- [8] <http://www.python.org>, 2013.
- [9] <http://numpy.scipy.org>, 2013.
- [10] Philip M. Morse. Diatomic molecules according to the wave mechanics. ii. vibrational levels. *Phys. Rev.*, 34:57–64, Jul 1929.