Solution of the Two Dimensional Schrödinger Equation using 16/8 Daubechies Wavelet Scaling Functions as Basis Set

Moritz Braun [1], **Obiageli L. Ezenwachukwu** [1]

[1] Department of Physics, University of South Africa , PO Box 392, 003 UNISA, South Africa

Email: moritz.braun@gmail.com

Abstract. In this contribution we demonstrate the suitability of the 16/8 Daubechies wavelets scaling function for solving the two dimensional Schrödinger equation using the example of the isotropic two dimensional harmonic oscillator. We show that calculations with this basis set converge quickly both for the eigenvalues as well as the wave functions. A convergence order of approximately 9 is established.

1 Introduction

Symlet8 or sym8 wavelets are specialized wavelets within wavelet theory valued for their near-symmetry, smoothness, and high-order vanishing moments[1]. The sym8 wavelet is applied within the Discrete Wavelet Transform (DWT) to split signals into detailed and approximation components. The improved symmetry of sym8 is preferred over db8 in image processing, edge detection, and analysis where signal phase information is important[2].
In this contribution we make use of the associated scaling functions modified to satisfy periodic boundary bonditions as 1D basis set to solve the two dimensional Schrödinger equation making use of higher order Gauss Legendre integration within the Galerkin approach employing 2D basis functions constructed as tensor product of the one-dimensional ones.

1.1 Definition of the Basis Set

We use two dimensional basis functions based on shifted and scaled scaling sym8 wavelet functions

$$f_i(x) = \phi\left(x/h + x_{\max}/h - i\right)/\sqrt{h} \tag{1}$$

with $h = x_{\max}/N$, where $x_{\max}$ is the distance from the origin to he boundary.
We however modify the functions by giving them periodic boundary conditions via

$$g_i(x) = \sum_{k=-K_{\max},}^{K_{\max}-1} f_i(x + kL) \tag{2}$$

with $L = 2x_{\max}$. For $x_{\max}$ large enough very this will have a negligible effect. It can be shown, that these basic functions are orthonormal. The two dimensional basis functions are given by

$$F_{ij}(x, y) = g_i(x)g_j(y) \tag{3}$$

1.2 Hamiltonian

The Hamiltonian for the two-dimensional Schrödinger equation is given by

$$H = -\frac{\partial^2}{\partial x^2} - \frac{\partial^2}{\partial y^2} + V(x,y) \tag{4}$$

and the matrix element is obtained as

$$h_{ij,kl} = \iint dxdyF_{ij}\left[-\frac{\partial^2}{\partial x^2} - \frac{\partial^2}{\partial y^2} + V(x,y)\right]F_{kl} \tag{5}$$

Because of the product structure of 2D basis functions this becomes

$$h_{ij,kl} = q_{ik}\delta_{jl} + \delta_{ik}q_{jl} + v_{ij,kl} \tag{6}$$

with

$$v_{ij,kl} = \iint dxdyF_{ij}V(x,y)F_{kl} \tag{7}$$

and

$$q_{ij} = \int dxg'_i(x)g'_j(x) \tag{8}$$

If V is the sum of a function of x and a function of y the potential also factorizes into a sum of two tensor products. However this assumption is not made for the calculation.

2 Details of Calculation

The double indices like ij will be transformed into one-dimensional indices before solving the eigenvalue probem. The calculations were done with a Python code of 130 lines using the modules pywt[3], scipy[4] and numpy[5]. The integrals above are evaluated via repeated Gauss Legendre integration of order 11 in 1D and 2D. The tensor products were evaluated via the Kronecker numpy.kron function in Python. The eigenvalue problem was solved using the module eigh from scipy.linalg. The Python source code of 130 lines is available on request from moritz.braun@gmail.com.

3 Results

We consider the two-dimensional harmonic oscillator $V(x,y) = x^2 + y^2$ with the eigenvalues 2ν for $\nu = 1, 2, \cdots$ and degeneracy ν for level ν.
In the table below the resulting 10 lowest eigenvalues are given for $x_{\max} = 8$ and for $N = 12, 16$ and 20 from left to right:

ν			
1	2.00029488	2.00001603	2.00000158
2	4.00191520	4.00011385	4.00001107
3	4.00191520	4.00011385	4.00001107
4	6.00353437	6.00021149	6.00002052
5	6.01034282	6.00070568	6.00006927
6	6.01034396	6.00070585	6.00006930
7	8.01196025	8.00080302	8.00007865
8	8.01196025	8.00080302	8.00007865
9	8.03803395	8.00307480	8.00031353
10	8.03803395	8.00307480	8.00031353

Table 1. Lowest ten eigenvalues obtained for N=12,16 and 20 from left to right

The convergence is quite fast.
We define $dE_1(N) = E_1(N) - 2$. A least square fit to

$$\log dE_1(N) = a + b \log N \tag{9}$$

for $N = 8, 10, 12, 14, 16, 18$ and 20 results in the plot below:

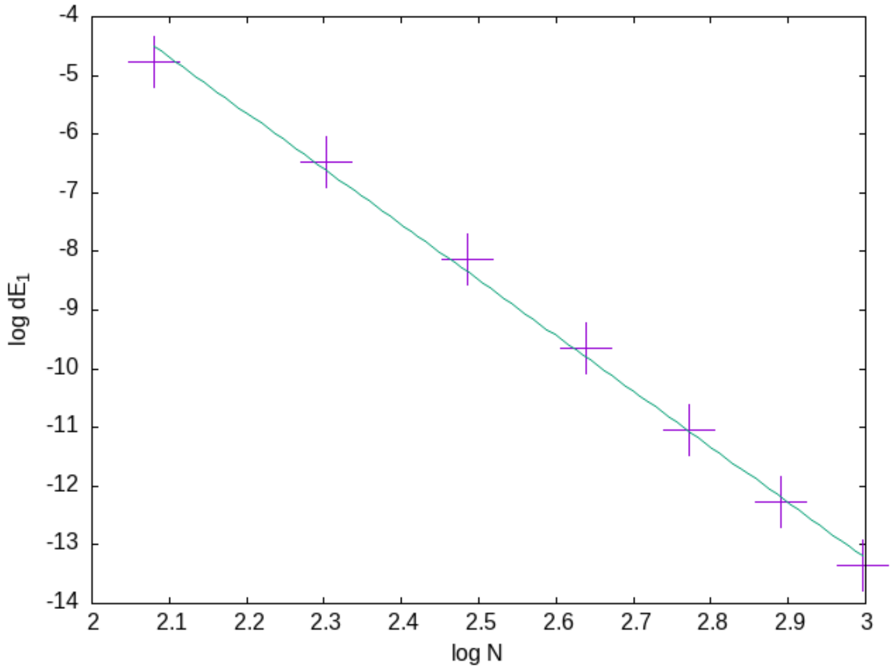

Figure 1. Linear Least Square Fit of $a + b \log N$ to $\log dE_1(N)$

3.1 Plots of wave functions

Below in Fig. 2 a) - 2 h) heat maps for the wave functions of the lowest 8 states of the 2D harmonic oscillator are shown.

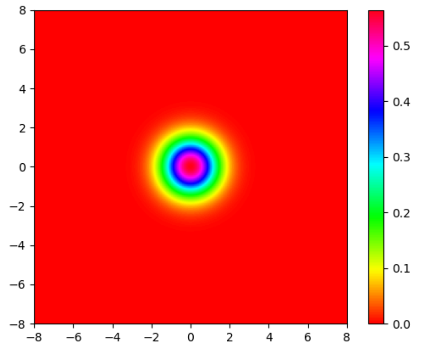

Fig 2(a): Heat map of ground state wave function

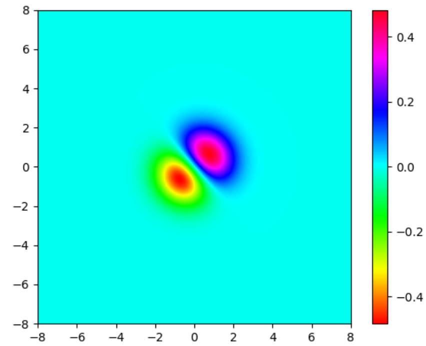

Fig 2(b): Heat map of first excited state

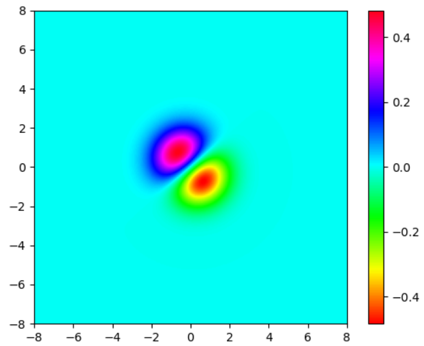

Fig 2(c): Heat map of second excited state

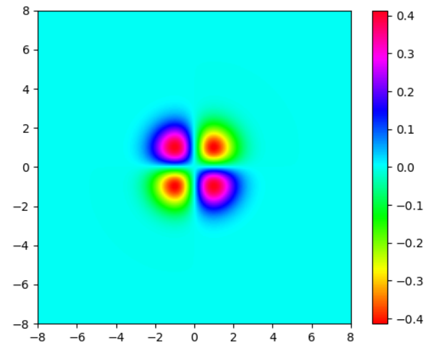

Fig 2(d): Heat map of third excited state

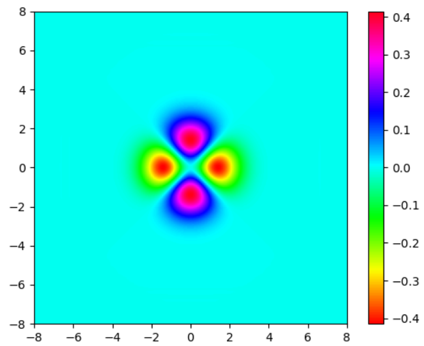

Fig 2(e): Heat map of fourth excited state

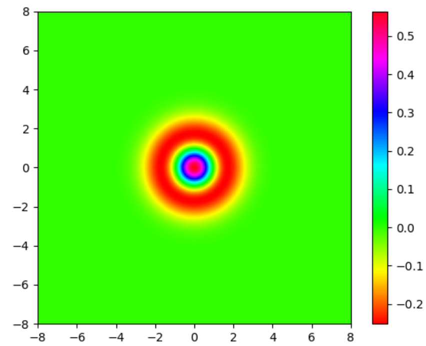

Fig 2(f): Heat map of fifth excited state

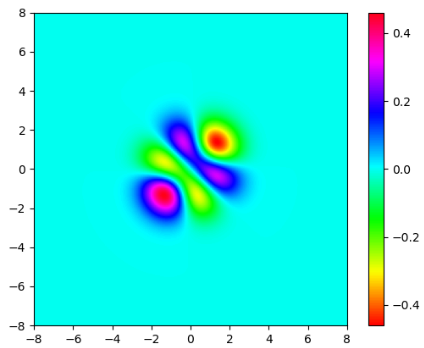

Fig 2(g): Heat map of sixth excited state

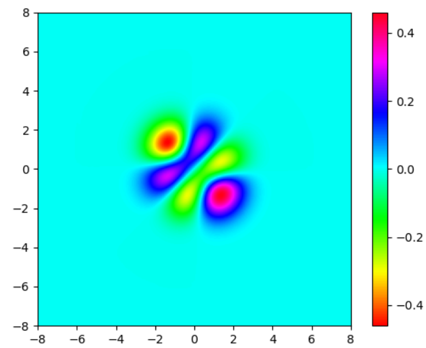

Fig 2(h): Heat mat of seventh excited state

Below a heatmap of the difference between the numerical and anylytical ground state wave function for $N = 20$ is shown:

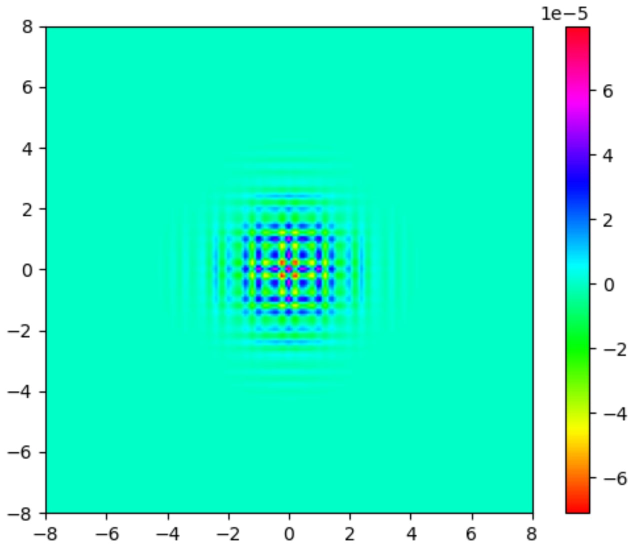

Figure 3. Heat map of difference between numerical and analytical ground state wave function

4 Conclusions

The two dimensional isotropic harmonic oscillator as a test case has been solved with wavelet scaling code. The convergence is quite rapid. In future we intend to use this two dimensional code for more interesting potentials.

Acknowledgements
This work was supported by the University of South Africa.

References

[1] A. Graps, "An introduction to wavelets," *IEEE Computational Science and Engineering, Summer 1995*, vol. 2, no. 2, 1995.

[2] A. Amid, "Comparative analysis of wavelet transformations for medical image processing with a focus on noise reduction," *ResearchGate*, Sep. 2023.

[3] G. R. Lee, R. Gommers, F. Waselewski, K. Wohlfahrt, and A. O'Leary, "Pywavelets: A python package for wavelet analysis," *Journal of Open Source Software*, 2019.

[4] E. Jones, T. Oliphant, P. Peterson *et al.*, "Scipy: Open source scientific tools for python," *http://www.scipy.org*, 2001.

[5] C. R. Harris, K. J. Millman, S. J. van der Walt, T. E. Oliphant *et al.*, "Array programming with NumPy," *Nature*, vol. 585, no. 7825, pp. 357–362, sep 2020.