Determining Quantum Wavefunctions of Electrons in Atoms using Physics-Informed Neural Networks

Daniel Johnstone

Supervisor: Dr. Mohammad Golbabaee

June 2023

Abstract

In this project a neural network was designed which is able to approximate solutions to the Schrödinger equation for electrons around an atom by leveraging new developments in Physics-Informed Neural Networks (PINNs). These types of neural network saw increase in use after being formulated by Raissi et al. in 2017 [1] in which it was shown they could be used to encode and model the equations of physical systems. Using a neural network was shown to be a very effective way to approximate a solution due to the how it learns the governing equation as part of it's loss function as well as it's initial/boundary conditions which are enforced in the network. This method is useful for systems that are described by partial differential equations (PDEs) due to their lack of exact solutions. One such PDE is the Schrödinger equation, the equation of state for quantum particles, which is challenging to solve. A PINN was created based on one published by Mattheakis et al. [2] in 2022 which was used to approximate the wavefunction and eigen-energy of a H_2^+ ion, this was then extended to be used on atoms of larger mass with more electrons. This provides an alternative method to calculate the solution to the Schrödinger equation for atoms rather than using other computational methods. The performance of the model was WHAT. The model had an average difference of WHAT to the solution using Density Functional Theory method and a difference of WHAT to a Hartree-Fock method solution for a WHAT atom. The corresponding difference in eigen-energy was WHAT and WHAT. The primary findings from the results are: WHAT,WHAT, and WHAT.

1 Introduction

The are some problems in science where the data is scarce and new methods need to be created that use our existing knowledge about a physical system. PINNs take the underlying physics in the form of a partial differential equation (PDE) and enforce it in a neural network to create an approximation of the solution [3]. A PDE contains partial derivatives which are functions that depend on multiple variables that have had their derivative taken with respect to one of those variables, while holding the others constant. These types of equations occur many times in physics in domains such as: electrodynamics, thermodynamics, fluid dynamics, elasticity, general relativity, and quantum mechanics. But finding solutions to these is, most of the time, impossible and in general PDEs are much harder to solve than ordinary differential equations. The importance of PDEs is exemplified by the millennium prize problem of the smoothness in the Navier-Stokes equations. The Navier-Stokes equations are central to fluid mechanics and describe the motion of fluids. The problem involves proving that given a set of initial conditions, smooth solutions will always exist. There have been partial results to the problem but it still remains unsolved. PINNs provide a new method of finding solutions to PDEs like Navier-Stokes and allow us to gain a deeper understanding of the form of the solutions.

The PINN approximates the PDE solutions by training a neural network to minimise a loss function which reflects the PDE including initial and boundary conditions [4]. The PDE residual is enforced at selected points referred to as collocation points. It can be thought of as an unsupervised learning approach not requiring labelled data and so results from experiments aren't required. PINNs can also be used to solve for the inverse problem where the model parameters are learnt from observable data, for example characterising fluid flows from sensor data. PINNs have advantages over alternate methods. For example they enable on-demand solution computation after a training stage, and they allow solutions to be made differentiable using analytical gradients.

The work done in this project is based on a paper by Mattheakis, Schleder, Larson, and Kaxiras which proposes a neural network which discovers parametric eigenvalue and eigenfunction surfaces of quantum systems [2]. This method applies a PINN to solve the Schrödinger equation for the H_2^+ ion using the Coulomb potential yielding realistic wavefunctions. It produces neural wavefunction solutions that are continuous and differentiable by the interatomic distance, this form of solution is useful for further calculations such as forces and vibrational frequencies. The neural network incorporates the approximate Linear Combination of Atomic Orbitals (LCAO) solutions which is a quantum superposition of the orbitals of the two hydrogen atoms. The neural energy is also calculated as the output of a separate feed-forward neural network (FFNN) that acts independently of the electron's positions.

The aim of this project is to recreate the neural network developed by Mattheakis et al. and extend it to the more complex case of solving the Schrödinger equation for larger atoms. These solutions are harder to obtain than those for the hydrogen atom due to the presence of multiple electrons interacting with one another. So this is a situation where using a PINN will be useful for approximating the electron wavefunctions. This can then be compared to the well established approximations of the Hartree-Fock method and density functional methods as well as experimental results to assess to quality of the PINN's output.

The resulting electron orbitals that are obtained from the wavefunction have an important role in chemistry. They determine the electron configuration of each atom which specifies the occupation of the different orbitals according to the Pauli exclusion and the Aufbau principles. They are used in chemical bonding to form of σ and π covalent bonds which are areas where the atomic orbitals between atoms overlap. The angles between atoms in molecules are affected by the orientation of atomic orbitals and as a result it's overall structural geometry. The atomic and molecular shape of the orbitals have an effect on the macroscopic chemical behaviour and properties. So having accurate knowledge about them is important to chemists in the development of new chemicals and materials. For more complex molecular structures PINNs can provide a way to approximate the orbital structure more efficiently than other methods.

2 Context and Literature Review

2.1 Developments in PINN

The usage of PINNs in solving PDEs and similar problems has greatly increased since they were introduced in a two-part paper by Raissi et al. in 2017 [1], [5] which was then merged and later published in a combined version in 2019 [6]. This put forward the method of solving two different classes of problems: finding the data-driven solution and data-driven discovery of PDEs. The training of a deep learning algorithm was eased by leveraging the physical laws that govern the system in question. This was demonstrated by approximating solutions to the Schrödinger, Burgers, and Allen-Cahn equations.

Leading up to this there were many developments in using neural networks for solving differential equations. One of the first implementations of what we would now consider a PINN goes as far back 1994 in a paper by Dissanayake and Phan-Thien [7]. In this a method was proposed that used a neural network to approximate the solution to a few differential equations including thermal conduction in a solid with non-linear heat generation.

In 1998 a simpler neural network was published by Lagaris et al., in it a multi-layer perceptron was used to solve several differential equations [8]. The model function used contained two terms: one which adhered to the initial and boundary conditions with no adjustable parameters and the second was a feed-forward neural network that satisfied the differential equation. Their results showed that using a neural networks was good for function approximation and that using a trial solution that conformed to boundary conditions created an unconstrained optimisation problem which is much easier to handle.

Using the PINN methodology several other methods have been developed such as the variational physics-informed neural network (VPINN) which is an approach based on the Petrov-Galerkin method [9]. The Petrov-Galerkin framework is used to approximate the solutions to PDEs in which the test function and solution function belong to different function spaces. This method incorporates a variational (weak) formulation of the problem and constructs a variational loss function, this is unlike the normal PINN methodology which uses a strong formulation. This provides some advantages such as: the smaller number of quadrature points used to compute the corresponding integrals than the penalising points in PINNs, for shallow networks the loss function can be expressed analytically which enables performing numerical analysis, and the domain can be decomposed into sub-domains each of which can use a separate number of test functions based on the local regularity of the solution, this yields a more flexible learning method. This development was shown to perform better than a PINN in terms of speed and accuracy.

An extension of the VPINN variation was developed know as hp-VPINN which uses hp-refinement [10]. This involves domain decomposition as h-refinement which divides the elements into smaller ones and projection onto space of high order polynomials as p-refinement which increases the polynomial degree. This method allows the flexibility to construct both local and global approximations each with a defined test function.

If a non-linear conservation law is being used then a PINN variation called conservative physics-informed neural networks (cPINN) can be applied, it uses multiple PINNs each with it's own neural network. A cPINN uses discrete domains and the solutions to each

sub-domain are patched together [11]. The conserved quantities are preserved by enforcing their continuity at the common interfaces of neighbouring sub-domains. The advantage of segmenting the domain like this is that it easily allows the network to be parallelised which is critical for obtaining computing efficiency. Also due to deployment of multiple neural networks the representation capacity of the network increases which is important for accurately predicting complex solutions. Another benefit is that there is less error propagation in comparison to a single network in a PINN, the solution to which can be polluted especially during the early training period. Whereas in a cPINN different networks in each sub-domain provide additional information about the fluxes at the interfaces and this results in the reduction of error propagation in the neighbouring sub-domains as well as faster convergence. In addition to this cPINNs enable a reduction in generalisation, approximation, and optimisation error through selection of penalising points, the selection of the hyperparameters, and changing the size of the network.

Another variation is the physics-constrained neural network (PCNN) which enforces initial or boundary conditions with a custom neural network architecture and embeds the PDE in the training loss. This method was originally developed in the 1990s to solve ordinary differential equations but was limited by computing power [12].

PINN research is still ongoing and new varieties of neural network are currently being developed. Other PINNs published since 2019 include: Sparse Physics-based and partially Interpretable Neural Networks (SPINN) [13], Neural homogenisation based PINN (NH-PINN) [14], physics-informed hypernetworks (hyperPINN) [15], Physics-informed Graph Neural Network (PhyGNNet) [16], and PINN on complex geometries (Δ-PINN) [17].

2.2 Atomic Orbital Approximation

One of the major advancements towards understanding the structure of atoms came from the Geiger–Marsden experiment. From this Ernest Rutherford concluded that a new description of the structure of the atom was required which separated the negatively charged electrons from the positively charged nucleus [18]. This model was further improved by the Bohr model which split the electrons into orbits around the central nucleus analogous to planets around the sun [19]. This model proved it's merit by explaining the line spectra observed around the hydrogen atom but fell short by conflicting with the later discovered Heisenburg's Uncertainty Principle which does not allow for an electron's orbit and location to be measured simultaneously.

In 1926 the Schrödinger equation was formulated and became one of the most significant developments in quantum theory. It functions similarly to Newton's equations of motion but applies to quantum particles [20]. It provided the ability to calculate how a wavefunction, which is a representation of the state of a quantum particle, changes over space and time. Using the Time-Independent Schrödinger Equation (TISE) the stationary state can be determined and, using it, the exact analytical solutions can be calculated for the non-relativistic hydrogen atom. Taking the square of the wavefunction gives the probability of finding the electron at a given location. This gave rise to a new model of the atom in which the electrons have wave-particle duality, they can operate as both waves and particles depending on the circumstance, and the physical shape of the space the electrons occupy is referred to as an

atomic orbital. The orbitals take the form of different shapes which depend on the electron's energy, angular momentum, and spin.

Unfortunately solving the Schrödinger equation for an atom with many electrons is analytically intractable and so new methods had to be developed to calculate these higher order atomic orbitals. One of these is the Hartree-Fock method which treats each electron as moving in an average field created by all other electrons [21]. The wavefunction of the system is approximated as a single Slater determinant, which is a determinant of one-electron orbitals. This produces a set of Hartree-Fock equations which must be solved numerically. Due to the nonlinearities introduced by the Hartree–Fock approximation, the equations are solved using a nonlinear method such as iteration.

The Configuration Interaction (CI) methods improve upon the Hartree-Fock method. To account for interactions between electrons, CI uses a variational wavefunction that is a linear combination of multiple Slater determinants [22]. The dimension of a CI procedure grows factorially with the number of electrons. This results in a long CPU time and large memory usage for calculations and so this method is only applied to smaller systems.

Electron correlation can also be accounted for using Many-Body Perturbation Theory (MBPT) which applies perturbation theory to the Hartree-Fock equation. The idea behind this is that the system, after a perturbation has been applied, is similar but in a slightly modified state [23]. If the unperturbed system is known then successive corrections can be applied to gain an approximation for the perturbed system. The order of corrections applied to the unperturbed system categorises each MBPT method, however there is no guarantee that the sum the correction will converge fast.

Another method of approximating the wavefunction of is using the Density Functional Theory (DFT). DFT doesn't directly solve the Schrödinger equation but instead finds the electron density that minimises the total energy functional. This uses the Hohenberg-Kohn theorem which states that the electron density uniquely determines the ground-state wavefunction and energy of a system [24]. From DFT the Kohn-Sahm equations are derived which are Schrödinger-like equations describing a system of non-interacting particles that generate the same density as a given system of interacting particle.

Information on electron orbitals can also be measured experimentally. These include: infrared spectroscopy which provides information on the vibrational modes of molecules which depend on the orbitals present, ultraviolet-visible spectroscopy involves the absorption of photons and by analysing the absorption spectra it is possible to determine the energy levels of different electron orbitals, nuclear magnetic resonance spectroscopy provides insights into the local chemical environment and electron density distribution, and x-ray crystallography reveals information about electron orbitals through the diffraction of X-rays by a crystal lattice. This gives scientists physical evidence to compare the approximations against.

There have been two main deep learning approaches to solving the Schrödinger equation. The first, which was used in this project, considers a PINN that predicts both the wavefunction and energy. This method minimises a loss function contain the Schrödinger PDE and the neural network approximately calculates the solutions [25]. The second approach uses the variational principle, the neural network only returns the wavefunction which can be used to calculate the energy using the expectation value of the hamiltonian which is minimised as part of the optimisation [26].

References

[1] M. Raissi, P. Perdikaris, and G. E. Karniadakis, "Physics informed deep learning (part i): Data-driven solutions of nonlinear partial differential equations," 2017. arXiv: 1711.10561 [cs.AI]. [Online]. Available: https://arxiv.org/abs/1711.10561.

[2] M. Mattheakis, G. R. Schleder, D. T. Larson, and E. Kaxiras, "First principles physics-informed neural network for quantum wavefunctions and eigenvalue surfaces," 2022. arXiv: 2211.04607 [cs.LG]. [Online]. Available: https://arxiv.org/abs/2211.04607.

[3] S. Kollmannsberger, D. D'Angella, M. Jokeit, and L. Herrmann, "Physics-informed neural networks," in *Deep Learning in Computational Mechanics: An Introductory Course*. Springer International Publishing, 2021, pp. 55–84, ISBN: 978-3-030-76587-3. DOI: 10.1007/978-3-030-76587-3_5. [Online]. Available: https://doi.org/10.1007/978-3-030-76587-3_5.

[4] S. Cuomo, V. S. D. Cola, F. Giampaolo, M. R. G. Rozza, and F. Piccialli, "Scientific machine learning through physics–informed neural networks: Where we are and what's next," *Journal of Scientific Computing*, 92 2022. DOI: https://doi.org/10.1007/s10915-022-01939-z.

[5] M. Raissi, P. Perdikaris, and G. E. Karniadakis, "Physics informed deep learning (part ii): Data-driven discovery of nonlinear partial differential equations," 2017. arXiv: 1711.10566 [cs.AI]. [Online]. Available: https://arxiv.org/abs/1711.10566.

[6] M. Raissi, P. Perdikaris, and G. E. Karniadakis, "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," *Journal of Computational Physics*, vol. 378, pp. 686–707, 2019, ISSN: 0021-9991. DOI: https://doi.org/10.1016/j.jcp.2018.10.045. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0021999118307125.

[7] M. W. M. G. Dissanayake and N. Phan-Thien, "Neural-network-based approximations for solving partial differential equations," *Communications in Numerical Methods in Engineering*, vol. 10, no. 3, pp. 195–201, 1994. DOI: https://doi.org/10.1002/cnm.1640100303. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/cnm.1640100303.

[8] I. E. Lagaris, A. Likas, and D. I. Fotiadis, "Artificial neural networks for solving ordinary and partial differential equations," *IEEE Transactions on Neural Networks*, vol. 9, no. 5, pp. 987–1000, 1998. DOI: 10.1109/72.712178. [Online]. Available: https://ieeexplore.ieee.org/document/712178.

[9] E. Kharazmi, Z. Zhang, and G. E. Karniadakis, "Variational physics-informed neural networks for solving partial differential equations," *CoRR*, vol. abs/1912.00873, 2019. [Online]. Available: http://arxiv.org/abs/1912.00873.

[10] E. Kharazmi, Z. Zhang, and G. E. Karniadakis, "Hp-vpinns: Variational physics-informed neural networks with domain decomposition," *CoRR*, vol. abs/2003.05385, 2020. [Online]. Available: https://arxiv.org/abs/2003.05385.

[11] A. D. Jagtap, E. Kharazmi, and G. E. Karniadakis, "Conservative physics-informed neural networks on discrete domains for conservation laws: Applications to forward and inverse problems," *Computer Methods in Applied Mechanics and Engineering*, vol. 365, 2020, ISSN: 0045-7825. DOI: https://doi.org/10.1016/j.cma.2020.113028. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0045782520302127.

[12] H. Lee and I. S. Kang, "Neural algorithm for solving differential equations," *Journal of Computational Physics*, vol. 91, no. 1, pp. 110–131, 1990, ISSN: 0021-9991. DOI: https://doi.org/10.1016/0021-9991(90)90007-N. [Online]. Available: https://www.sciencedirect.com/science/article/pii/002199919090007N.

[13] A. A. Ramabathiran and P. Ramachandran, "Spinn: Sparse, physics-based, and partially interpretable neural networks for pdes," *Journal of Computational Physics*, vol. 445, 2021, ISSN: 0021-9991. DOI: https://doi.org/10.1016/j.jcp.2021.110600. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0021999121004952.

[14] W. T. Leung, G. Lin, and Z. Zhang, "Nh-pinn: Neural homogenization-based physics-informed neural network for multiscale problems," *Journal of Computational Physics*, vol. 470, 2022, ISSN: 0021-9991. DOI: https://doi.org/10.1016/j.jcp.2022.111539. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0021999122006015.

[15] F. A. Belbute-Peres, Y. Chen, and F. Sha, "HyperPINN: Learning parameterized differential equations with physics-informed hypernetworks," in *The Symbiosis of Deep Learning and Differential Equations*, 2021. [Online]. Available: https://openreview.net/forum?id=LxUuRDUhRjM.

[16] L. Jiang, L. Wang, X. Chu, Y. Xiao, and H. Zhang, *Phygnnet: Solving spatiotemporal pdes with physics-informed graph neural network*, 2023. [Online]. Available: https://arxiv.org/abs/2208.04319.

[17] F. S. Costabal, S. Pezzuto, and P. Perdikaris, Δ*-pinns: Physics-informed neural networks on complex geometries*, 2022. [Online]. Available: https://arxiv.org/abs/2209.03984.

[18] E. Rutherford, "The scattering of α and β particles by matter and the structure of the atom," *Philosophical Magazine*, vol. 21, 6 1911. [Online]. Available: https://www.lawebdefisica.com/arts/structureatom.pdf.

[19] N. Bohr, "I. on the constitution of atoms and molecules," *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, vol. 26, no. 151, pp. 1–25, 1913. DOI: 10.1080/14786441308634955. [Online]. Available: https://doi.org/10.1080/14786441308634955.

[20] E. Schrödinger, "An undulatory theory of the mechanics of atoms and molecules," *Phys. Rev.*, vol. 28, pp. 1049–1070, 6 Dec. 1926. DOI: 10.1103/PhysRev.28.1049. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRev.28.1049.

[21] D. R. Hartree and W. Hartree, "Self-consistent field, with exchange, for beryllium," *Proceedings of the Royal Society of London. Series A - Mathematical and Physical Sciences*, vol. 150, no. 869, pp. 9–33, 1935. DOI: 10.1098/rspa.1935.0085. [Online]. Available: https://royalsocietypublishing.org/doi/abs/10.1098/rspa.1935.0085.

[22] C. D. Sherrill and H. F. Schaefer, "The configuration interaction method: Advances in highly correlated approaches," in ser. Advances in Quantum Chemistry, P. Löwdin, J. R. Sabin, M. C. Zerner, and E. Brändas, Eds., vol. 34, Academic Press, 1999, pp. 143–269. DOI: https://doi.org/10.1016/S0065-3276(08)60532-8. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0065327608605328.

[23] "Chapter 10 - correlation of the electronic motions," in *Ideas of Quantum Chemistry*, L. Piela, Ed., Elsevier, 2007, pp. 551–557, ISBN: 978-0-444-52227-6. DOI: https://doi.org/10.1016/B978-044452227-6/50011-9. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780444522276500119.

[24] P. Hohenberg and W. Kohn, "Inhomogeneous electron gas," *Phys. Rev.*, vol. 136, B864–B871, 3B Nov. 1964. DOI: 10.1103/PhysRev.136.B864. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRev.136.B864.

[25] H. Jin, M. Mattheakis, and P. Protopapas, "Physics-informed neural networks for quantum eigenvalue problems," 2022. arXiv: 2203.00451 [cs.LG]. [Online]. Available: https://arxiv.org/abs/2203.00451.

[26] D. Pfau, J. S. Spencer, A. G. D. G. Matthews, and W. M. C. Foulkes, "Ab-initio solution of the many-electron schrödinger equation with deep neural networks," *Physical Review Research*, vol. 2, no. 3, Sep. 2020. [Online]. Available: https://doi.org/10.1103%5C%2Fphysrevresearch.2.033429.

3 Appendix

3.1 Project Plan

Week Beginning:	Tasks to Undertake:
3/7	Adapt the code that functions for H_2^+ ion so that it works for the hamiltonian of a larger atom with electrons.
10/7	Experiment with simple atoms and/or molecules. There may be issues in making sure code is working so this week will also be used for fixing any issues that arise.
17/7	Analysis of the PINN. How well does it perform in comparison to the hydrogen model?
24/7	Experiment to find if it is possible to improve upon the PINN architecture.
31/7	Depending on if previous tasks have been reached by this point will try to compute orbitals using Hartree-Fock and DFT based approximations. This will allow for comparison with the performance of the more common approximation methods.
7/8	Try to finish all outstanding work by this week. This week is set aside for gathering all relevant graphs for the thesis. Begin writing thesis this week with a rough draft/outline.
14/8	Thesis Writing: Theory and Methodology sections.
21/8	Thesis Writing: Results, Discussion, and conclusion sections.
28/8	Thesis Writing: Finishing off all sections ready for submission.

3.2 Risk Assessment

This project doesn't involve any major risks related to access to data due to the fact the PINNs don't require labelled data. Therefore there are no ethical decisions to make regarding the usage of private people's data. As far as I'm aware there aren't any social, environmental, or ethical impact that results from the use of PINNs.

The neural network used in [2], which this project is based on, is coded in python using pytorch. There is a low risk of either of these pieces of software vausing problems, and I have run both on my computer to make sure they are working properly.

Another source of risk is my personal computer which recently broke down. If this were to happen during the project and have to be factory reset again, this could majorly push back the timeline of my work. It is unknowable if this will happen but due to it's potential impact I have decided to takes steps to mitigate it. Firstly, I set up a GitHub repository for my work so that nothing can get lost. Secondly, I have applied through the university for an Amazon Web Service (AWS) learner lab account which I will set with an Elastic Compute Cloud instance on which I can run my neural network if my computer is in a state of disrepair.