

Physics Informed Neural Networks for a Wigner-Fokker-Planck model of Open Quantum Systems

Isaul Garcia^{1,2}[0009-0004-1313-1613] and Jose Morales Escalante²[0000-0001-6650-9510]

¹ Boston University, Boston MA 02215, USA igarcia1@bu.edu
<https://www.bu.edu/physics/profile/isaul-garcia/>

² University of Texas at San Antonio, San Antonio TX 78249, USA
jose.morales4@utsa.edu
<https://josemoralesescalante.wordpress.com/>

Abstract. We present the preliminary results of a Physics-Informed Neural Network (PINN) for a Wigner-Fokker-Planck (WFP) equation modeling open quantum systems such as electron transport in semiconductors. The WFP equation is a mathematical model that considers diffusion and friction introduced by the environment into an open quantum (sub-)system, describing the problem via a continuous quantum variable formulation. Recent developments in scientific machine learning have demonstrated that PINNs are useful in providing data-driven solutions to partial differential equations (PDE) and for data-driven discovery [20] in the estimation of model parameters, particularly when constrained to small or noisy data. PINNs minimize a residual additional to the typical Neural Network approach related to the satisfaction of a PDE that represents the “Physics” of the problem, along with the traditional loss function that estimates the fit to the data. The former residual implicitly trains the model to respect conservation principles following from the PDE that models the Physics phenomena. Since optimization is not solely dependent on minimizing the fit to the training data, such as in traditional machine learning models, it can perform exceedingly well for inverse problems to estimate model parameters, such as the diffusion and friction parameters in the particular case of our Wigner-Fokker-Planck model, when constrained to small data. This work used the PINN methodology to solve a data-driven discovery problem for the Wigner-Fokker-Planck equation. In particular, we solved an inverse problem with synthetic data obtained from a Monte Carlo forward solver of the Wigner-Fokker-Planck equation to estimate the elements of our diffusion matrix as parameters of the model representing noise introduced by the environment into the open quantum system.

Keywords: Data-driven discovery of PDEs · Physics Informed Neural Networks · Wigner function · Fokker-Planck · open quantum systems · electron transport in semiconductor materials.

1 Introduction

1.1 Physics Informed Neural Networks

Physics-informed neural networks are a new machine learning model that allows the computation of data-driven solutions and the data-driven discovery of partial differential equations (PDEs). The primary deviation from a traditional neural network is the addition of a PDE-related residual in the loss function. The minimization of this residual forces the network to produce a computational solution closer to the true solution of the PDE related to the physics of the problem, consequently enforcing the solution to respect implicitly (through penalization) the conservation principles originating from the physical laws expressed in the PDE as mentioned above [20]. Conventional deep learning models rely heavily on the quality and amount of data, since even the best models fail to perform well with insufficient data. This understanding has resulted in a data-centric approach focused on machine learning instead of a model-centric approach. This is problematic in scientific machine learning, where data acquisition is commonly prohibitive for physical systems and also measurement noise is potentially unavoidable [20]. Enforcing PINNs to respect the physics of the problem through a PDE-related residual increases the robustness of the network, giving adequate results when trained with noisy data or with a smaller amount of data [20], making them advantageous for "machine-learning" physical systems.

We will consider as in [20] a parametrized PDE of the form

$$w_t + \mathcal{L}[w; \lambda] = 0, \quad x \in \Omega, \quad t \in [0, T], \quad (1)$$

where $w(t, x)$ is the unknown/latent/hidden solution, $\mathcal{L}(\cdot, t)$ is an operator (possibly nonlinear) with a parameter λ , and Ω is a subset of \mathbb{R}^D constituting the domain of the problem. PINNs can tackle two types of problems when one has systems measurements with noise. The first is the so-called "data-driven solution of PDEs" [20], where given some fixed known parameters λ , we approximate the unknown solution $w(t, x)$. The second is the "data-driven discovery of PDEs" [20], where we find the model parameters λ that better fit the data for $w(t, x)$. These problems are traditionally referred to as forward and inverse modeling, respectively, in the mathematical modeling community [23].

In the Data-Driven Solution case, PINNs is focused on the forward modeling of a PDE for a time evolution problem in the general form

$$w_t + \mathcal{L}[w] = 0, \quad x \in \Omega, \quad t \in [0, T]. \quad (2)$$

Since all PDE parameters are assumed to be known in this case, λ is omitted. $w(t, x)$ is the unknown solution as above mentioned, $\mathcal{L}(\cdot)$ is a (possibly nonlinear) differential operator (for example, if it was linear, it could be a Sturm-Liouville operator for boundary value problems), and $\Omega \subset \mathbb{R}^D$ is the domain. Our residual $f(t, x)$ will be the left-hand side of the equation (2),

$$f := w_t + \mathcal{L}[w]. \quad (3)$$

Using a neural network, we can approximate $w(t, x)$, and then derive $f(t, x)$ by using the chain rule for the derivatives of composed functions via the well-known process in Machine Learning of automatic differentiation as in [2]. This approximates $f(t, x)$, constituting our Physics-Informed-Neural-Network or PINN. Quoting [20], "This network can be derived by applying the chain rule for differentiating compositions of functions using automatic differentiation [2], and has the same parameters as the network representing $[w(t, x)]$, albeit with different activation functions due to the action of the differential operator" \mathcal{L} .

In PINNs, one has a single neural network for $w(t, x)$ and $f(t, x)$ (where the spatio-temporal derivatives are obtained through automatic differentiation) with multiple loss terms [20], and the goal is to minimize the so-called mean squared error loss function (encompassing the multiple loss terms), given by

$$MSE = MSE_w + MSE_f, \quad \text{with} \quad (4)$$

$$MSE_w = \frac{1}{N_w} \sum_{i=1}^{N_w} |w(t_w^i, x_w^i) - w^i|^2, \quad MSE_f = \frac{1}{N_f} \sum_{i=1}^{N_f} |f(t_f^i, x_f^i)|^2, \quad (5)$$

where $\{t_w^i, x_w^i, w^i\}_{i=1}^{N_w}$ represent the initial and boundary training data for $w(t, x)$, whereas $\{t_f^i, x_f^i\}_{i=1}^{N_f}$ are just collocation points of $f(t, x)$. The addition of MSE_f to the loss function enforces the neural network approximation of the solution to the PDE $w(t, x)$ to also respect the Physics embedded by equation (2), training the network to get closer to the PDE solution of the physical law, and consequently learn the Physics of the model [20].

The Machine Learning model reported by Raissi et al. [20] has claimed to observe empirical evidence that performing the minimization together with the additional PDE residual, as aforementioned, behaves as a regularization mechanism, allowing a relatively simple architecture, such as a feed-forward neural network, to perform well when trained with small amounts of potentially noisy data. They also argue that this technique is appropriate since back-propagation is currently the dominant approach for training deep learning models, where derivatives are taken with respect to (w.r.t.) the model's parameters (i.e., weights, bias), where PINNs also use automatic differentiation to take derivatives w.r.t. the input coordinates (both space and time) to construct the residual.

For the inverse modeling case of Data-Driven Discovery, the neural network is trained to learn the model parameters, denoted by λ , given some sparse measured solutions of the PDE as training data. Defining $f(t, x)$ to be the left-hand-side of equation (1) as below,

$$f := w_t + \mathcal{L}[w; \lambda], \quad (6)$$

we also approximate the solution $w(t, x)$ with a neural network and derive $f(t, x)$ using automatic differentiation, resulting in the PINN $f(t, x)$. In the inverse case, the parameters of \mathcal{L} become parameters of the PINN for $f(t, x)$, which the network is trained to learn. We can train the shared parameters and the PINNs' exclusive parameters with the same loss function defined in the forward case (5), but now $\{t_w^i, x_w^i, w^i\}_{i=1}^{N_w}$ denotes the training data of $w(t, x)$, which

includes any potential initial and boundary conditions, and $\{t_f^i, x_f^i\}_{i=1}^{N_f}$ specifies the collocation points, which are at the same location as the training data [20].

1.2 Literature Review

Regarding previous work on Physics-Informed Neural Networks, a brief literature review would need to start by mentioning the use of Machine Learning in Computational Physics. Previous work in this regard used machine learning, though just as a black-box tool [25], in which a study of physics-informed machine learning for predictive turbulence modeling was presented. Another example of machine learning to predict physical systems is [26], in which Bayesian deep convolutional encoder-decoder networks are used for surrogate modeling and uncertainty quantification, which is particularly related to the PINNs development since it considers similar approaches to the ones considered in deep learning for image-to-image regression tasks. We would also like to mention the work in [10] where differential equations with unknown constitutive relations are solved as recurrent neural networks (in which time series measurements of state variables are partially available, and recurrent neural networks are used to learn the ODE reaction rate as Physics of the problem from the data), as well as [24], where deep neural networks are used to learn surrogate models for numerical simulators and uncertainty quantification in high dimensions.

However, the work in [20] has been greatly seminal, since it revisited the activation and loss functions, but for the differential operator, removing the black-box approach by understanding the automatic differentiation in deep learning. This work uses automatic differentiation, as in deep learning, to "Physics inform" neural networks by differentiation w.r.t. the spatiotemporal coordinates. This creates some regularization and allows the use of a simple feed-forward neural network architecture and training with small data (the latter being quite important for Physics phenomena, in which the amount of data acquired is much smaller than for image recognition or data collection from mobile phones).

There has been work to understand why these kinds of machine-learning-based methodologies work, such as the work in "Why does deep & cheap learning work so well?" [15], which anticipated some preliminaries of the development of PINNs by exploring how physical properties such as symmetry, locality, etc., imply exceptionally simple neural networks. PINNs also partially build upon the work in [19] that combines a partial first principles model incorporating available prior knowledge about modeled processes with a neural network estimating unmeasured process parameters difficult to model from first principles, obtaining better properties than "black-box" neural network models, as well as the work in [14] on artificial neural networks for solving ordinary and partial differential equations (writing the trial solution as a term satisfying the initial and boundary conditions without adjustable parameters plus another term not affecting the initial/boundary conditions), and the work in [12] on a covariant hierarchical neural network architecture for learning atomic potentials (in which the atomic potential energy surfaces are learned to be used in molecular dynamics simulations), as well as the work in [13] on the generalization of equivariance and

convolution in neural networks to the action of compact groups (beyond only equivariance to translations in image recognition). There is also important work in [11] on the wavelet scattering regression of quantum chemical energies (where multiscale invariant dictionaries are introduced to estimate quantum chemical energies of organic molecules from training databases), and in [17] on the understanding of deep convolutional networks (which is a work of great relevance to PINNs and Machine Learning in general since it intends to set the foundations of deep convolutional networks with tools similar on how Physics deals with high dimensional problems in Statistical Mechanics).

Regarding the particular use of Machine Learning for open quantum systems, previous work on this line has been developed in [18] on deep reinforcement learning for quantum state preparation with weak nonlinear measurements. There is also work in [5] on the machine learning based noise characterization and correction on neutral atoms NISQ devices, as well as work in [1] on the quantum Fokker-Planck master equation for continuous feedback control. There is also work in [7] on solving inverse stochastic problems from discrete particle observations using the Fokker-Planck Equation and Physics-informed Neural Networks. There is work in [16] on solving multiscale steady radiative transfer equations using neural networks with uniform stability. However, to our knowledge, nobody has applied Physics-Informed Neural Networks to study of open quantum systems yet, particularly the Wigner-Fokker-Planck model of open quantum systems such as electron transport in semiconductors. This is our paper's contribution.

1.3 Wigner-Fokker-Planck Equation

We aim to solve an inverse modeling problem for the Wigner-Fokker-Planck (WFP) model for open quantum systems, under a harmonic potential, with a Physics Informed Neural Network. Equations of this type need efficient computing methods for its forward and inverse modeling, which we wish to study using this technique for the latter case. Open quantum systems are quantum subsystems that interact with their environment (through energy exchanges, for example) [21]. WFP is a kinetic quantum model in continuous variables (position-momentum based) for open quantum systems. It finds particular applications in electronic charge transport in semiconductor materials and devices [9], quantum optics, and quantum information science and engineering (QISE), for example. WFP describes the time evolution of a so-called quasi-probability Wigner function $w(x, k, t)$, which is obtained by applying a Wigner transform [6] to the density matrix $\rho(x, y, t)$ [9], where x, y are position variables and $p = \hbar k$ is the proportionality relation between the momentum p and the wave vector k .

The Wigner-Fokker-Planck equation is given by

$$w_t + k \cdot \nabla_x w + \Theta_{\hbar}[V](w) = Q_{\hbar,FP}(w), \quad (7)$$

where $Q_{\hbar,FP}(w)$ is the Quantum Fokker-Planck operator, which models the averaged environmental interactions with the system, and $\Theta_{\hbar}[V](w)$ is the integral pseudo-differential operator that takes into account the non-local action of

the potential V [21]. Though formally speaking the potential would include both the self-consistent part of the electrons interacting among themselves and an applied external bias, for this benchmark study we will assume the total potential is known, and that this total potential is harmonic, so that we can compare against the benchmark steady-state analytical solution for the harmonic case obtained in [22]. The Quantum Fokker-Planck operator is local and represents diffusion and friction (under Markovian approximations for the noise), given by

$$Q_{\hbar,FP}(w) = 2\gamma\nabla_k \cdot (kw) + D_{qq}\nabla_x^2 w + 2\frac{D_{pq}}{m}\nabla_x \cdot \nabla_k w + \frac{D_{pp}}{m^2}\nabla_k^2 w, \quad (8)$$

where the following parameters,

$$\gamma = \frac{\lambda}{2m}, \quad D_{qq} = \frac{\lambda\hbar^2}{12m^2K_bT}, \quad D_{pq} = \frac{\lambda\Omega\hbar^2}{12\pi mk_bT}, \quad D_{pp} = \lambda k_bT \quad (9)$$

are such that γ is a friction coefficient, and the diffusion matrix is

$$D = \begin{pmatrix} D_{qq} & D_{qp} \\ D_{pq} & D_{pp} \end{pmatrix}, \quad D_{pq} = D_{qp}. \quad (10)$$

\hbar is Planck's constant, m is the mass of the particle, k_b is Boltzmann's constant, T is temperature, λ is the coupling constant, and Ω is the cutoff frequency. The constants above satisfy the Lindblad condition $D_{qq}D_{pp} - D_{pq}^2 \geq \hbar^2\gamma^2/4$, and $\Omega \leq k_bT/\hbar$, which guarantee the quantum mechanically correct evolution of the system (as in the respective density matrix being completely positive trace-preserving, referred to as CPTP). Under the classical limit $\hbar/S^* \rightarrow 0$, with S^* a characteristic action unit of the system, the WFP will converge to the classical Fokker-Planck dynamics [9]. We choose units so that the following normalization holds: $\hbar = \lambda = m = K_b = T = 1$, and together with $\Omega = 0$, the parameters simplify to $D_{pq} = 0$, $D_{qq} = D_{pp} = 1$, $\lambda = \frac{1}{2}$.

The pseudo-differential integral operator for a given potential V acting over w is defined as

$$\Theta_{\hbar}[V]\{w\} = \frac{\int_{\mathbb{R}^{2n}} \left[V\left(x + \frac{\hbar\eta}{2m}\right) - V\left(x - \frac{\hbar\eta}{2m}\right) \right] w(x, p, t) e^{i\eta \cdot (k-p)} d\eta dp}{-i\hbar(2\pi)^n}. \quad (11)$$

Because this is indeed an integral operator, this makes it the most computationally expensive term in a numerical simulation of WFP. However, if we assume the confining potential V to be harmonic, as in the quadratic form below,

$$V(x) = \frac{\omega_0^2}{2}|x|^2 + a \cdot x + b, \quad a \in \mathbb{R}^d, \quad b \in \mathbb{R}, \quad \omega_0 \geq 0, \quad (12)$$

then the pseudo-differential operator simplifies to [22]

$$\Theta[V]\{w\} = -\omega_0^2 x \cdot \nabla_k w. \quad (13)$$

This results in the Wigner-Fokker-Planck equation simplifying to the convection-diffusion equation below,

$$w_t + k \cdot \nabla_x w - \omega_0^2 x \cdot \nabla_k w = Q_{h,FP}(w), \quad (14)$$

$$x \in \mathbb{R}^d, k \in \mathbb{R}^d, t \in \mathbb{R}^+, w(t=0, x, k) = w_0(x, k).$$

This can be written more compactly as

$$w_t + \nabla_{(x,k)} \cdot (P(x, k)w) = \nabla_{(x,k)} \cdot (D\nabla_{(x,k)}w), \quad w(0, x, k) = w_0(x, k), \quad (15)$$

with $w_0(x, k)$ the initial condition of the Wigner function at time $t = 0$ (for example, a harmonic ground state), where

$$D := \begin{pmatrix} D_{qq}\mathbb{I}_d & D_{pq}\mathbb{I}_d \\ D_{pq}\mathbb{I}_d & D_{pp}\mathbb{I}_d \end{pmatrix}, \quad P(x, k) := \begin{pmatrix} k \\ -\omega_0^2 x - 2\gamma k \end{pmatrix}. \quad (16)$$

We will study this simplified case of the Wigner-Fokker-Planck equation under a known harmonic potential since there is a known analytical steady-state solution for it, found by Sparber et al. on [22]. The above makes the problem perfect as a benchmark for any computational solver of the WFP equation, since the computational simulations at long times can be compared to the analytical steady state solution known to this problem. Additionally, our research group has developed a Monte-Carlo forward solver for WFP under this benchmark case [8], which we used to produce synthetic training data for the Wigner Function $w(k, x, t)$ in our PINNs methodology for inverse problems, as we will explain in detail in the next section.

Under a particular choice of values for our parameters, the diffusion matrix and the transport vector $P(x, k)$ (which also includes the friction term) become

$$D := \begin{pmatrix} \mathbb{I}_d & 0 \\ 0 & \mathbb{I}_d \end{pmatrix}, \quad P(x, k) := \begin{pmatrix} k \\ -x - k \end{pmatrix}. \quad (17)$$

This is the particular problem we will study in our inverse modeling study through PINNs for WFP, which we describe in the following sections.

2 Methodology

For the Wigner-Fokker-Planck equation, we are interested in the inverse problem case of finding the diffusion matrix elements, knowing beforehand that their true value is $D_{pq} = 0 = D_{qp}$, $D_{qq} = 1 = D_{pp}$ and confirming the PINNs method will recover values close to the true ones as a benchmark problem. We can define $f(x, k, t)$ as the left-hand-side of equation (15), which will be our residual corresponding to the PDE representing the Physics of the model,

$$f := w_t + \nabla_{(x,k)} \cdot (P(x, k)w) - \nabla_{(x,k)} \cdot (D\nabla_{(x,k)}w). \quad (18)$$

We approximate the Wigner function $w(x, k, t)$ by a single output neural network and derive $f(x, k, t)$ using automatic differentiation, resulting in the

PINN $f(x, k, t)$. The parameters D_{pp} , D_{qq} , and D_{pq} turn into exclusive parameters of the Physics-Informed Neural Network $f(x, k, t)$, which the network will estimate. The diffusion parameters of the Wigner-Fokker-Planck equation, as well as the parameters of the neural networks $[w(x, k, t)]$, $[f(x, k, t)]$, are learned by minimizing the following mean squared loss,

$$MSE = MSE_w + MSE_f, \quad \text{where} \quad (19)$$

$$MSE_w = \frac{1}{N_w} \sum_{i=1}^{N_w} |w(x^i, k^i, t^i) - w^i|^2, \quad MSE_f = \frac{1}{N_f} \sum_{i=1}^{N_f} |f(x^i, k^i, t^i)|^2. \quad (20)$$

To solve for the steady-state solution of WFP under a harmonic potential, we have generated a synthetic training dataset from our Monte Carlo forward solver [8] with the same normalization in 1D in x and 1D in p (Figs. 1-2).

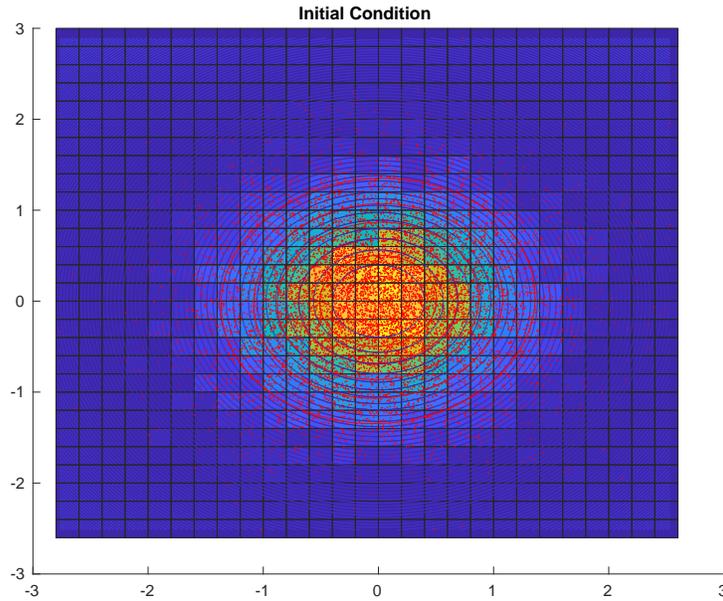


Fig. 1. Wigner-Fokker-Planck Initial Condition: Harmonic Groundstate, represented as a point distribution by sampling the respective Gaussian Wigner function [8].

The models were also trained with randomly sampled data, with half the models having 1% uncorrelated Gaussian noise applied to the training data to test its performance, given noisy measurements. Our data contains 2640 points in phase space and 500 temporal points to the steady-state solution, resulting in a total dataset of size $NT = 1,320,000$. To also test how well the model performs with small data, we trained the model with only 6600 randomly sampled points, $N_{\text{Train}} = 6600$, corresponding to only 0.5% of the total available data. We use

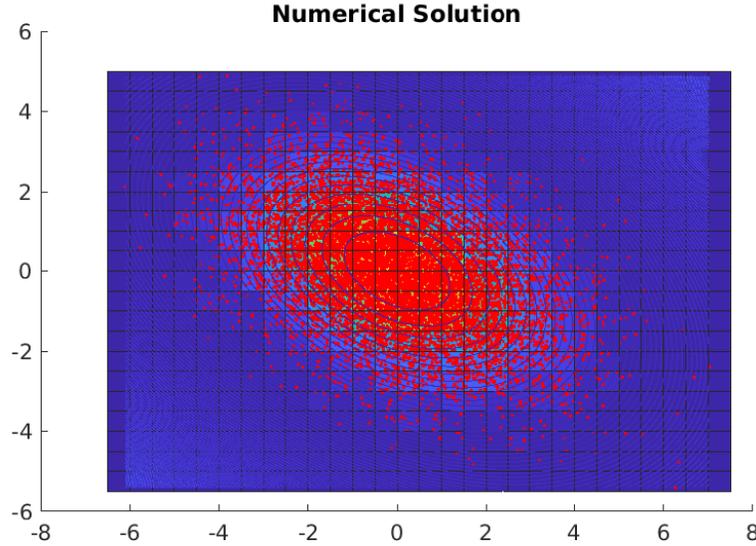


Fig. 2. Wigner-Fokker-Planck Numerical Steady State Solution obtained by a Monte Carlo based Euler-Maruyama method [8].

the same architecture as the Navier-Stokes PINN in [20], which has 8 hidden layers and 20 neurons per layer with hyperbolic tangent activation functions. However, the WFP PINN has a 3D input of $[x, k, t]$ and a 1-dimensional output $[w(x, k, t)]$.

3 Results

3.1 Wigner-Fokker-Planck Equation

We use as an initial condition for the WFP equation the harmonic ground state (as in Fig. 1), given by the following Wigner function (in units such that $\hbar = 1$ and then $2/\hbar = 1/\pi$),

$$w(x, k, t)|_{t=0} = w_0(x, k) = \frac{1}{\pi} \exp(-[(x/L)^2 + (kL)^2]) \quad (21)$$

where $L = 1$ is the characteristic unit of length for the system. Since our forward solver for the WFP equation relies on a Monte Carlo-based Euler-Maruyama method rather than a finite element/difference method, no boundary conditions are needed. The initial condition is in the form of a point distribution obtained by sampling from a Gaussian distribution in position-momentum phase space (x, k) related to the harmonic groundstate in the Wigner form abovementioned in (21), so the sampling will be concentrated around the maximum of the Gaussian

distribution for the harmonic ground-state (which is the origin), and will have a very low probability of sampling beyond 3 standard deviations after the origin in either the position or momentum variable. In the forward solver, we just let each one of the points sampled in the initial condition to advect under noise according to the quantum diffusive transport of WFP via the Euler-Maruyama method, so we do not predetermine the domain in phase space in which the sampled points will move over the forward solution of the problem numerically. We used $N_{\text{samples}} = 10^4$ sampled initial points for our forward solver, which were afterwards evolved according to our Monte Carlo solver for our WFP quantum diffusive transport model. We chose $N_f = N_{\text{train}} = 6600$, since the number of points sampled from the entire data set for training was $N_{\text{train}} = 6600$, and we estimated $N_w = 13$ for the initial condition (no boundary conditions are imposed, therefore, there are no boundary points for N_w). For the WFP PINN, we ran each model 110 times. For one set of models, we applied 1% uncorrelated Gaussian noise to the training data, representing measurement noise.

Table 1. Summary of Different Models’ Performance for WFP. Average of 140 Trials Reported. All were trained on NVIDIA Tesla V100 from the Arc UTSA HPC facilities.

Quantity of interest	Error for noiseless training	Error for noisy training
Relative error in L_2 -norm for Wigner function	0.0021	0.0023
D_{pq} absolute error	0.0038	0.0037
D_{pp} relative error	1.509 %	1.512 %
D_{qq} relative error	0.0629 %	0.0623 %

Both functionals $MSE_w = J_d$ and $MSE_f = J_f$ are equally prioritized in the loss function (20). Therefore, quantitatively, once their values are of the same magnitude, we expect them to be equally minimized during the training, which is the observed behavior.

We present in Fig. 3 a plot of the J_f vs. J_d functionals over the iteration, as well as a plot of the. Loss function $J_f + J_d$ versus the Iteration number (for both figures, the broken red line indicates where the L-BFGS-B algorithm begins). We also present in Fig. 4 a plot of the neural network prediction for the steady state compared against the exact steady state, for which their similarity illustrates the successful learning of the WFP model by PINNs. In Fig. 5 we present a plot of the neural network prediction L_1 error norm, comparing the difference between the predicted steady state versus the exact steady state. The low magnitude of the values compared with the ones in Fig. 4 shows the small error norm obtained. We present in Fig. 6 as well a plot of the neural network predictions for the diffusion matrix elements over different iterations, which shows the convergence of the diffusion matrix elements towards the expected values of 0 or 1.

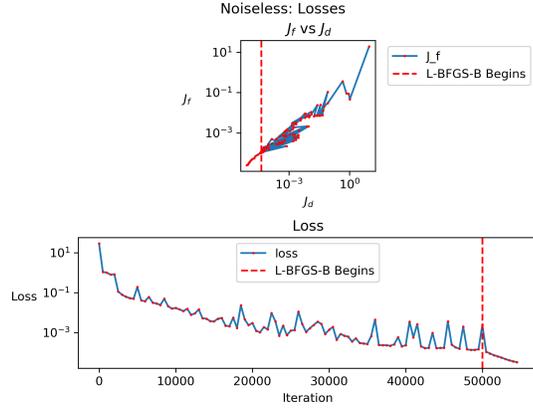


Fig. 3. Top: J_f vs. J_d functionals. Bottom: Loss function $J_f + J_d$ versus Iteration number. The broken line in red indicates where the L-BFGS-B algorithm begins.

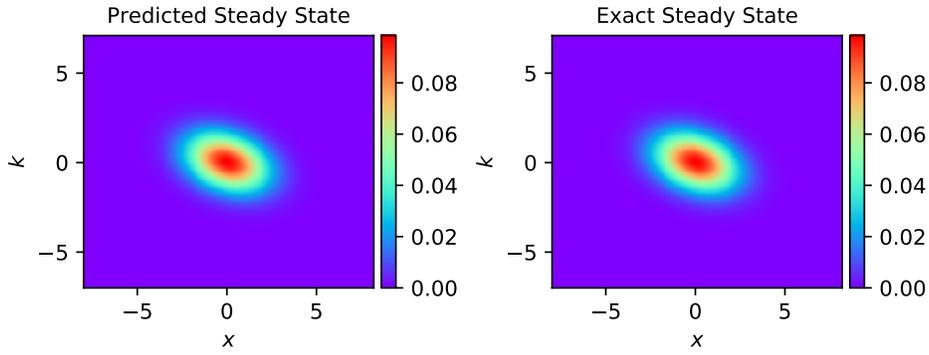


Fig. 4. Plot of the neural network prediction for the steady state compared against the exact steady state. Their similarity illustrates the successful learning of the WFP model by PINNs.

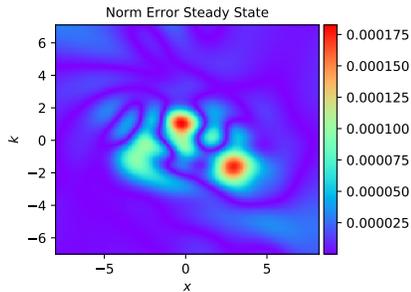


Fig. 5. Plot of the neural network prediction L_1 error norm, comparing the difference between the predicted steady state versus the exact steady state. The low magnitude of the values compared with the ones in Fig. 4 shows the small error norm obtained.

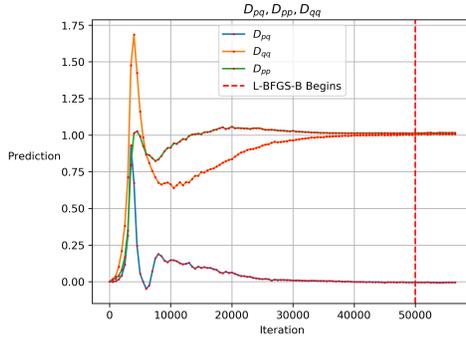


Fig. 6. Plot of the neural network predictions for the diffusion matrix elements over different iterations. The plot shows the convergence of the diffusion matrix elements to the expected values of 0, 1, depending on the respective element.

4 Discussion

For the WFP PINN, we produce small errors, of the order 1% when estimating the diffusion parameters, while only training on 0.5% of available data. These results are of similar order to the ones found by Raissi et al. [20] for the inverse modeling of a Navier-Stokes problem where the viscosity coefficient was found. The above indicates that the PINN methodology is promising for future applications and development, for example, in the case of Quantum Information Science and Engineering, for state preparation via Lindbladians. In state preparation via Lindbladians, one can engineer the noise to conduct the state to a given desired form for sufficiently long times in certain cases. In the case of materials science problems, such as electron transport in semiconductor devices, the use of PINNs for the WFP equation could characterize the quantum diffusion matrix and friction coefficients representing the scattering processes that electron materials undergo when traveling across semiconductors. For example, in silicon, the most common scattering mechanisms are electron-phonon collisions, so the solution of this inverse problem for WFP in PINNs would determine the parameters representing the quantum scattering of electrons undergoing transport across semiconductor materials such as silicon, germanium, etc. PINNs could also be used for more complex non-harmonic potentials where there is no analytic solution, as for the real case of the electric potential made of self-consistent electron interactions and applied external biases (in which case, this potential has to be obtained by solving a Poisson equation coupled to WFP). Real-world lab data could be used and it would be necessary to study the non-harmonic case. We believe that the same performance of our Physics-Informed Neural Network can be accomplished for anharmonic potentials because the conceptual idea will be the same: provided we have data from a forward solver of the WFP equation for a given potential (harmonic or not, even if on the latter case we don't know the analytical formula for the steady state solution), since the prob-

lem in the forward solver is inherently of a convective-diffusive nature as in the Wigner-Fokker-Planck equation, the inverse modeling procedure carried through PINNs will be able to identify parameters such as the diffusion matrix and the friction coefficient because the Physics-Informed-Neural-Network has the same Physics that the forward model from which the data comes from, regardless of the potential being harmonic, anharmonic, or even more complex as having a self-consistent part obtained from a Poisson equation, as for more complex WFP-Poisson systems coupling these two equations for the problem of electron transport in semiconductors at a quantum scale.

5 Conclusion and Future Work

In the WFP case, the errors for the diffusion parameters D_{pq} and D_{qq} reported in Table 1 are smaller for the models trained with data that had noise applied to it. Though this could seem in principle unexpected given the prominent margins between the errors for models trained with noise and without it reported in [20], we would like to mention that noise can also have a stabilizing effect on neural network training and can also help prevent overfitting as discussed in [3, 4]. For example, we add Gaussian noise to the input variables and also for training. On the other hand, the used 1% noise is minimal and could be considered small for traditional feed-forward neural networks. PINNs are considered more noise resilient because of the PDE residual embedded in their optimization process. Future work could include studying how the accuracy degrades as noise increases, which could validate the noise resilience claimed for the studied PINN. Future work on the computational math side could also include hyperparameter tuning for the activation functions, parameters for the architecture, and potentially how much to scale the residuals corresponding to the data and PDE in a multi-objective optimization manner. Currently, they are equally weighted, but this can be adjusted through a multi-objective optimization methodology as aforementioned, where the functional optimized is made of a convex combination of both data and Physics-related functionals, with the convex combination parameter $\rho \in (0, 1)$ defining the weights given to each of the two functionals respectively. Pareto curves could be produced in that case for the set of multiple convex combinations of the data and Physics-related functionals to find the optimal values of ρ that minimize the total loss without overfitting. Future work on the materials science side could be related on trying to use real data from current-voltage characteristics ($I-V$ curves) to fit the respective quantum diffusion matrix and friction coefficient characterizing different semiconductor devices by performing steady-state simulations of the WFP-Poisson system and finding the parameter values whose observables and expectation averages fit the behavior of the $I-V$ curves available in the literature for the respective materials and devices under consideration.

Acknowledgments. We would like to thank UTSA’s University Tech Solutions’ (UTS) Research Computing Support Group (RCSG). This work received computational support from UTSA’s HPC cluster Arc, as we were given allocation time for its

use. Startup funds from the University of Texas at San Antonio are gratefully acknowledged as well.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Annby-Andersson, B., Bakhshinezhad, F., Bhattacharyya, D., De Sousa, G., Jarzynski, C., Samuelsson, P., Potts, P.P.: Quantum fokker-planck master equation for continuous feedback control. *Physical Review Letters* **129**(5), 050401 (2022)
2. Baydin, A.G., Pearlmutter, B.A., Radul, A.A., Siskind, J.M.: Automatic differentiation in machine learning: a survey. *Journal of machine learning research* **18**(153), 1–43 (2018)
3. Bishop, C.M.: Training with noise is equivalent to tikhonov regularization. *Neural Computation* **7**(1), 108–116 (01 1995). <https://doi.org/10.1162/neco.1995.7.1.108>, <https://doi.org/10.1162/neco.1995.7.1.108>
4. Bishop, C.M.: *Neural Networks for Pattern Recognition*. Oxford University Press, Inc., USA (1995)
5. Canonici, E., Martina, S., Mengoni, R., Ottaviani, D., Caruso, F.: Machine learning based noise characterization and correction on neutral atoms nisc devices. *Advanced Quantum Technologies* **7**(1), 2300192 (2024)
6. Case, W.B.: Wigner functions and Weyl transforms for pedestrians. *American Journal of Physics* **76**(10), 937–946 (10 2008). <https://doi.org/10.1119/1.2957889>, <https://doi.org/10.1119/1.2957889>
7. Chen, X., Yang, L., Duan, J., Karniadakis, G.E.: Solving inverse stochastic problems from discrete particle observations using the fokker-planck equation and physics-informed neural networks (2020), <https://arxiv.org/abs/2008.10653>
8. Edrisi, A., Patwa, H., Morales Escalante, J.A.: A numerical study of quantum entropy and information in the wigner-fokker-planck equation for open quantum systems. *Entropy* **26**(3) (2024). <https://doi.org/10.3390/e26030263>, <https://www.mdpi.com/1099-4300/26/3/263>
9. Gamba, I., Gualdani, M., Sharp, R.: An adaptable discontinuous galerkin scheme for the wigner-fokker-planck equation. *Communications in Mathematical Sciences* **7** (09 2009). <https://doi.org/10.4310/CMS.2009.v7.n3.a7>
10. Hagge, T., Stinis, P., Yeung, E., Tartakovsky, A.M.: Solving differential equations with unknown constitutive relations as recurrent neural networks (2017)
11. Hirn, M., Mallat, S., Poilvert, N.: Wavelet scattering regression of quantum chemical energies. *Multiscale Modeling and Simulation* **15**(2), 827–863 (Jan 2017). <https://doi.org/10.1137/16m1075454>, <http://dx.doi.org/10.1137/16M1075454>
12. Kondor, R.: N-body networks: a covariant hierarchical neural network architecture for learning atomic potentials (2018)
13. Kondor, R., Trivedi, S.: On the generalization of equivariance and convolution in neural networks to the action of compact groups (2018)
14. Lagaris, I., Likas, A., Fotiadis, D.: Artificial neural networks for solving ordinary and partial differential equations. *IEEE Transactions on Neural Networks* **9**(5), 987–1000 (1998). <https://doi.org/10.1109/72.712178>

15. Lin, H.W., Tegmark, M., Rolnick, D.: Why does deep and cheap learning work so well? *Journal of Statistical Physics* **168**(6), 1223–1247 (Jul 2017). <https://doi.org/10.1007/s10955-017-1836-5>, <http://dx.doi.org/10.1007/s10955-017-1836-5>
16. Lu, Y., Wang, L., Xu, W.: Solving multiscale steady radiative transfer equation using neural networks with uniform stability (2022), <https://arxiv.org/abs/2110.07037>
17. Mallat, S.: Understanding deep convolutional networks. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* **374**(2065), 20150203 (2016)
18. Porotti, R., Essig, A., Huard, B., Marquardt, F.: Deep Reinforcement Learning for Quantum State Preparation with Weak Nonlinear Measurements. *Quantum* **6**, 747 (Jun 2022). <https://doi.org/10.22331/q-2022-06-28-747>, <https://doi.org/10.22331/q-2022-06-28-747>
19. Psychogios, D.C., Ungar, L.H.: A hybrid neural network-first principles approach to process modeling. *AIChE Journal* **38**(10), 1499–1511 (1992). <https://doi.org/https://doi.org/10.1002/aic.690381003>, <https://aiche.onlinelibrary.wiley.com/doi/abs/10.1002/aic.690381003>
20. Raissi, M., Perdikaris, P., Karniadakis, G.: Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics* **378**, 686–707 (2019). <https://doi.org/https://doi.org/10.1016/j.jcp.2018.10.045>, <https://www.sciencedirect.com/science/article/pii/S0021999118307125>
21. Sharp, R., Gualdani, M., Gamba, I.: A discontinuous galerkin method for the wigner-fokker-planck equation with a non-polynomial approximation space. *Pamm* **7**, 2020105–2020106 (12 2007). <https://doi.org/10.1002/pamm.200700738>
22. Sparber, C., Carrillo, J., Dolbeault, J., Markowich, P.: On the long time behavior of the quantum fokker-planck equation. *Monatshefte fur Mathematik* **141** (05 2003). <https://doi.org/10.1007/s00605-003-0043-4>
23. Tarantola, A.: Inverse Problem Theory and Methods for Model Parameter Estimation. *Society for Industrial and Applied Mathematics* (2005). <https://doi.org/10.1137/1.9780898717921>, <https://epubs.siam.org/doi/abs/10.1137/1.9780898717921>
24. Tripathy, R.K., Bilonis, I.: Deep uq: Learning deep neural network surrogate models for high dimensional uncertainty quantification. *Journal of Computational Physics* **375**, 565–588 (Dec 2018). <https://doi.org/10.1016/j.jcp.2018.08.036>, <http://dx.doi.org/10.1016/j.jcp.2018.08.036>
25. Wu, J.L., Xiao, H., Paterson, E.: Physics-informed machine learning approach for augmenting turbulence models: A comprehensive framework. *Physical Review Fluids* **3**(7) (Jul 2018). <https://doi.org/10.1103/physrevfluids.3.074602>, <http://dx.doi.org/10.1103/PhysRevFluids.3.074602>
26. Zhu, Y., Zabaras, N.: Bayesian deep convolutional encoder–decoder networks for surrogate modeling and uncertainty quantification. *Journal of Computational Physics* **366**, 415–447 (Aug 2018). <https://doi.org/10.1016/j.jcp.2018.04.018>, <http://dx.doi.org/10.1016/j.jcp.2018.04.018>