



A Critical Evaluation of Using Physics-Informed Neural Networks for Simulating Voltammetry: Strengths, Weaknesses and Best Practices

Haotian Chen^a, Christopher Batchelor-McAuley^a, Enno Kätelhön^b, Joseph Elliott^a, Richard G. Compton^{a,*}

^a Department of Chemistry, Oxford University, South Parks Road, Oxford OX1 3QZ, Great Britain, United Kingdom

^b MHP Management- und IT-Beratung GmbH, Königsallee 49, 71638 Ludwigsburg, Germany

ARTICLE INFO

Keywords:

Physics-Informed Neural Network
Artificial Intelligence
Efficient numerical simulation of voltammetry
Diffusion in variable geometries

ABSTRACT

The recent explosion of applications of physics-informed neural networks (PINNs) as a discretization-free tool to solve partial differential equations (PDEs) shows great potential for applications in electroanalytical simulations. However, a simple, naive PINN approach may fail to make analytical level predictions in even only moderately complicated systems. Here, we explore eight test cases, spanning 1D to 3D simulations, including both cyclic voltammetry and chronoamperometry, and a wide selection of electrode geometries from macro-electrode, (hemi-)spherical electrode, microband electrode, cube electrode and microdisc electrode to serve the dual-purposes of expanding PINNs to more challenging electrode geometries and to recommend best practices in the context of electroanalytical simulation. These best practices, include the use of dimensionless parameters, non-zero conditioning times, mathematical transformation of PDEs, sequence-to-sequence training, adaptive weights algorithms, optimal batch sizing, domain decomposition, learning rate scheduling and transformation of coordinates. These suggested best practices are the intended key contribution of this paper, as to position future PINN users with a well-informed starting position for generic electroanalytical PINN simulations, to avoid known difficulties and to skip the trial-and-error phase with hyperparameter tuning. We believe that these recommendations can serve as primers for PINN simulations for sophisticated Multiphysics problems, and make PINN simulations more accessible.

1. Introduction

Physics-informed neural networks (PINNs), are a class of neural networks embedding physical laws, in the form of partial differential equations (PDEs), into their loss functions. Since the introduction of PINNs by Raissi [1], PINNs have become increasingly popular for modeling and/or understanding physical problems due to the simplicity brought by PINN in enabling mesh-free simulations compared with finite difference method (FDM) or finite element method (FEM). A review of physics-informed machine learning and tutorials about PINNs can be found in literature [2,3]. We note abundant recent publications applying PINNs in fluid mechanics [4], including inverse problems in supersonic flows [5], turbulence modelling [6], flow and mass transport in porous media [7,8], blood flow in cardiovascular tissues [9], etc. The underlying neural network structure, has evolved from fully connected neural networks [1], to convolutional neural networks [10,11], coupling residual block [12] and generative-adversarial networks [13] for their specific physical scenarios. Despite the rich presence of publications in fluid mechanics and heat transfer

problems, PINN simulations for chemistry are less extensive, but recent literature includes solving chemical kinetics [14,15], predicting drug-target interaction [16], modeling of fermentation for β -carotene production [17], and understanding of catalytic CO₂ methanation [18]. In the context of electrochemistry, applications of PINNs for cyclic voltammetry and hydrodynamic voltammetry were reported recently by the present authors [19,20]. Instead of reiterating the already reported success of PINN in voltammetry, this work focuses on the pros, cons and best practices using PINNs for electrochemistry.

In the following we apply PINNs to a wide range of voltammetric problems requiring simulations in varying dimensions with the aim of drawing generic conclusions about the strengths and weaknesses of the PINN approach to voltammetry. 1D simulations reported are of chronoamperometry and cyclic voltammetry at a macro electrode and chronoamperometry at a (hemi-)spherical electrode. 2D simulations are of chronoamperometry at a microband electrode. Last 3D simulations of both cyclic voltammetry and chronoamperometry at a cube electrode and chronoamperometry at a microdisc electrode (using Cartesian coordinates) are discussed. Validation of chronoampero-

* Corresponding author.

E-mail address: Richard.compton@chem.ox.ac.uk (R.G. Compton).

grams is made via comparison with analytical expressions wherever possible to allow evaluations of relative errors. The ability of extrapolation, an advantage of PINNs over conventional data-driven artificial neural networks, was also evaluated in some of these scenarios [21,22]. Simulation of cyclic voltammetry at a cube, which has not been reported before to the best of our knowledge, was validated by comparison with voltammograms generated by FDM. These simulations serve as the scenarios for the later discussion and evaluation of PINNs for application in voltammetry.

The wide range of scenarios addressed enables a critical evaluation of the pros and cons of the PINN approach to simulation. We note the success of solving 3D problems with complex boundary conditions using PINN, with accuracy comparable to that of FDM. We also note that PINN, as a mesh-free simulation tool, allows easy implementation of simulation in high dimensions.

It is evident from previous reports that PINN approach is not always problem free: Raissi has pointed out the bottleneck in higher dimensional problems, as the total number of collocation points needed to enforce physical laws increases exponentially, making computation very expensive [1]; difficulty of convergence to satisfactory levels of accuracy have also been noted [23]; multiple loss functions (for PDEs and boundary conditions) may compete with each other during training [2]; the spectral bias, the learning bias of neural network toward low frequency functions, may lead to trivial solutions [24]; the difficulty of optimizing jagged loss landscape for more complex problems [25] is a major failure mode of PINNs and the unbalanced back-propagated gradient during training is another [26]. In addition, the boundary condition representing the electrode surface may increase the stiffness of PDEs, making it harder for PINNs to untangle.

In the light of the mentioned difficulties above, this work aims to identify and elaborate caveats for training PINN in general and voltammetry in specific. The scenarios up to three-dimensional simulations help to identify best practices of applying PINNs to voltammetric simulation, including the dimensionless parameters, the conditioning time for chronoamperometry, sequence-to-sequence training, optimal batch size, the learning rate scheduler, subdomains of temporal spatial collocation points and adaptive weights algorithm. Best practices will be introduced in specific scenarios to show their remedies to the difficulties of training PINN. Lastly, we believe the best practices obtained from the scenarios mentioned above will serve as the starting point for more researchers to apply PINNs to more challenging scenarios, and to integrate multiple physical laws in chemistry and electrochemistry to solve cutting edge problems. Implementation of the scenarios and best practices can be found at <https://github.com/nmerovgian/PINN-Voltammetry-Best-Practices>.

2. Theory

Throughout this paper we exclusively consider the following electrochemical reaction in respect of chronoamperometry and cyclic voltammetry at different electrode geometries:



where A and B are stable, solution phase species. The mass transport to a microelectrode in three-dimensions is

$$\begin{cases} \frac{\partial c_A}{\partial t} = D_A \left(\frac{\partial^2 c_A}{\partial x^2} + \frac{\partial^2 c_A}{\partial y^2} + \frac{\partial^2 c_A}{\partial z^2} \right) \\ \frac{\partial c_B}{\partial t} = D_B \left(\frac{\partial^2 c_B}{\partial x^2} + \frac{\partial^2 c_B}{\partial y^2} + \frac{\partial^2 c_B}{\partial z^2} \right) \end{cases} \quad (2)$$

where D_A and D_B are the diffusion coefficients for species A and B . The diffusion equations can be further simplified for specific electrode geometries as mentioned later. Assuming both species have equal diffusion coefficients, solving only for species A implies the concentration of B as $c_A + c_B = c_A^*$ where c_A^* is the bulk concentration of A and assuming $c_B^* = 0$.

For chronoamperometry, if a high overpotential is applied to fully 'drive' reaction (eq (1)), the boundary condition at the surface of electrode is simply $c_A = 0$. For cyclic voltammetry, the boundary condition is assumed to obey the Nernst equation:

$$E = E_f^0(A/B) + \frac{RT}{F} \ln \frac{c_A}{c_B} \quad (3)$$

where E and E_f^0 are the applied potential and formal potential respectively. R , T and F are the Gas constant, temperature and the Faraday constant, respectively. The triangular waveform for cyclic voltammetry as a function of time, t is:

$$\begin{cases} 0 < t < t_{\text{switch}}, E = E_i - \nu t \\ t = t_{\text{switch}}, E = E_{\text{switch}} \\ t_{\text{switch}} < t < 2t_{\text{switch}}, E = E_{\text{switch}} + \nu(t - t_{\text{switch}}) \end{cases} \quad (4)$$

where E_i , E_{switch} are the starting and switching potential, respectively. t_{switch} is the time point at the switch and ν is the voltage scan rate.

PINN, like all other neural network, are highly sensitive to the system scale and/or parameterization coefficient of the governing PDEs [27]. For example, using a typical diffusion coefficient of $10^{-9} \text{m}^2 \text{s}^{-1}$ and concentration of 10mM in an experimental context, the neural networks would be highly susceptible to diminishing gradient problems, leading to the failure of simulations. Thus, using dimensionless parameters for simulation [28], as multiply stressed in the literature [4,28,29], is desirable. The dimensionless parameters for specific electrode geometries are articulated in Table 1. The dimensionless parameters also remove the dependence on electrode size and diffusion coefficients, increasing the generality of the model.

For chronoamperometry, the simulation time is simply the chronoamperometric time window. For cyclic voltammetry, the dimensionless potential window is noted as $[\theta_i, \theta_{\text{switch}}]$, where θ_i and θ_{switch} are the starting and switching potentials respectively. The time frame of the scan is thus $T_{\text{sim}} = \frac{2[\theta_i - \theta_{\text{switch}}]}{\sigma}$, where σ is the dimensionless scan rate. The maximum distance of simulation is guided by Einstein's work on Brownian motion [30] and the root-mean-square displacement of a particle in one dimension is $x_{\text{RMS}} = \sqrt{2Dt}$. The outer boundary of simulation, x_{sim} is thus located at $x_{\text{sim}} = 6\sqrt{Dt_{\text{sim}}}$ to ensure that the electrode surface is unperturbed by the interfacial electrochemical reaction. The dimensionless boundaries in all three dimensions are thus $X_{\text{sim}} = Y_{\text{sim}} = Z_{\text{sim}} = 6\sqrt{T_{\text{sim}}}$. The boundary condition at a macro-electrode for 1D chronoamperometry is:

$$C_A(X = 0, T) = 0 \quad (5)$$

That of 1D cyclic voltammetry assuming Nernst electrode kinetics, is:

$$C_A(X = 0, T) = \frac{1}{1 + \exp(-\theta)} \quad (6)$$

PINN simulation of 1D voltammetry permits first the prediction of the temporal evolution of the concentration profile, followed by the mathematical approximation of the flux at the electrode surface. The PDE and boundary conditions are described in the following form:

$$\frac{\partial C}{\partial T} - \frac{\partial^2 C}{\partial X^2} = 0 \text{ on } \mathcal{T} \times \Omega_X \quad (7.1)$$

$$C = f(T) \text{ on } \mathcal{T}, X = 0 \quad (7.2)$$

$$C = 1 \text{ on } \mathcal{T}, X = X_{\text{sim}} \quad (7.3)$$

$$C = 1, \text{ on } T = 0, \Omega_X \quad (7.4)$$

where $\mathcal{T} \in [0, T_{\text{sim}}]$ and $\Omega_X \in [0, X_{\text{sim}}]$ represent the temporal and one-dimensional spatial domains. Eq (7.1) represents the diffusion equation where the dimensionless diffusion coefficient is always 1 assuming equal diffusion coefficients of both species. Eq (7.2), 7.3 and 7.4 are

Table 1

Dimensionless parameters for the 1D macro and (hemi-) sphere electrode, 2D microband electrode, 3D cube and microdisc electrode.

Parameter	1-D macro	1-D (hemi-) sphere electrode	2-D microband electrode	3D cube electrode	3D microdisc electrode
Concentration	$C_j = \frac{c_j}{c_A}$				
Diffusion coefficient	$d_j = \frac{D_j}{D_A}$				
Spatial coordinates	$X = \frac{x}{r_e}$	$R = \frac{r}{r_e}$	$X = \frac{x}{a}, Y = \frac{y}{a}$	$X = \frac{x}{a}, Y = \frac{y}{a}, Z = \frac{z}{a}$	$X = \frac{x}{r_e}, Y = \frac{y}{r_e}, Z = \frac{z}{r_e}$
Time	$T = \frac{D_A t}{r_e^2}$	$T = \frac{D_A t}{r_e^2}$	$T = \frac{D_A t}{w^2}$	$T = \frac{D_A t}{a^2}$	$T = \frac{D_A t}{r_e^2}$
Potential	$\theta = \left(\frac{F}{RT}\right)(E - E_f^0)$				
Scan rate	$\sigma = \left(\frac{r_e^2}{D_A}\right)\left(\frac{F}{RT}\right)\nu$	$\sigma = \left(\frac{r_e^2}{D_A}\right)\left(\frac{F}{RT}\right)\nu$	$\sigma = \left(\frac{w^2}{D_A}\right)\left(\frac{F}{RT}\right)\nu$	$\sigma = \left(\frac{a^2}{D_A}\right)\left(\frac{F}{RT}\right)\nu$	$\sigma = \left(\frac{r_e^2}{D_A}\right)\left(\frac{F}{RT}\right)\nu$
Current	$J = \frac{I}{\pi r_e F C_A D_A}$	$J_{\text{hemi-spherical}} = \frac{I}{2\pi r_e F C_A D_A}, J_{\text{spherical}} = \frac{I}{4\pi r_e F C_A D_A}$	$J = \frac{I}{F C_A D_A b}$	$J = \frac{I}{F C_A D_A a} (\text{each face})$	$J = \frac{I}{\pi r_e F C_A D_A}$

c_j and D_j are the concentration and diffusion coefficient for species j respectively. r_e is the radius of disc or spherical electrode; w and b are the width and length of the microband electrode; a is the half length of the edge of a cube. Note that temperature is included in some definitions, notably σ and θ . No specific temperature is assumed in the results reported.

Dirichlet boundary conditions representing electrode surface, outer boundary of simulation and initial condition respectively. The electrode surface boundary condition is a function of time: in the case of chronoamperometry, $C = 0$ for the full duration of the scan and in the case of cyclic voltammetry $C = \frac{1}{1+\exp\theta}$. The training data for PINN are thus four sets of randomly generated collocation points within each temporal spatial domain as shown in Fig. 1. For example, \mathcal{N} collocation points, $\{T_i, X_i\}_{i=1}^{\mathcal{N}}$ on $\mathcal{T} \times \Omega_X$ are used to enforce eq (7.1). Three other sets of collocation points, $\{T_i, 0\}_{i=1}^{\mathcal{N}}$, $\{T_i, X_{\text{sim}}\}_{i=1}^{\mathcal{N}}$ and $\{0, X_i\}_{i=1}^{\mathcal{N}}$ are used to enforce eq7.2 to 7.4 respectively [31].

PINNs are trained by enforcing the physical laws in the form of PDEs. By integrating the governing PDEs, initial conditions and boundary conditions into their loss functions, PINNs make physics consistent predictions by minimizing these losses during training. Mean squared error (MSE) functions are used to enforce the physics laws. For example, for the diffusion law shown in eq (7.1), the loss function is:

$$MSE_f = \frac{1}{\mathcal{N}} \sum_{i=1}^{\mathcal{N}} \left(\frac{\partial C_i}{\partial T_i} - \frac{\partial^2 C_i}{\partial X_i^2} \right)^2 \quad (8)$$

where C_i are concentrations predicted by the neural network, $\frac{\partial C_i}{\partial T_i}$ and $\frac{\partial^2 C_i}{\partial X_i^2}$ are derivatives produced by auto-differentiation in the neural network frameworks using `tf.gradients()` in TensorFlow or `torch.autograd.grad()` in PyTorch. Similarly, the boundary condition at the electrode surface can be enforced by minimizing MSE_{surf} as

$$MSE_{\text{surf}} = \frac{1}{\mathcal{N}} \sum_{i=1}^{\mathcal{N}} (C_i - f(T))^2 \quad (9)$$

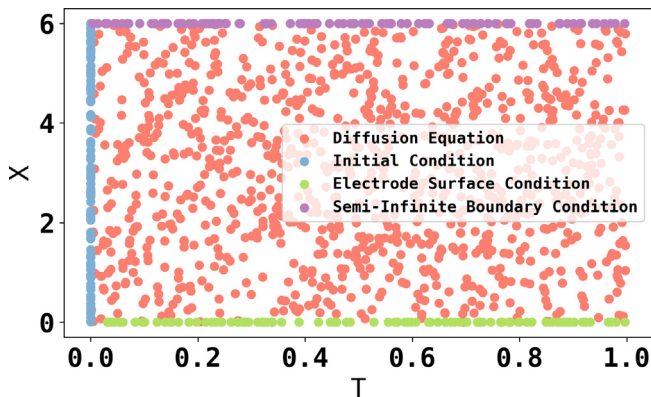


Fig. 1. Visualization of four sets of collocation points for PINN simulation of diffusion to a macroelectrode.

Combining the four loss functions for physical laws, boundary conditions and initial conditions, the total loss function used by PINN is composed as:

$$\mathcal{L} = \omega_f MSE_f + \omega_{\text{surf}} MSE_{\text{surf}} + \omega_{\text{outer}} MSE_{\text{outer}} + \omega_{\text{ini}} MSE_{\text{ini}} \quad (10)$$

where MSE_{outer} and MSE_{ini} are losses for the outer boundary and initial condition respectively. ω is a hyperparameter setting the weights of each loss functions. Setting all weights to 1 was found to be a good starting point, and if PINN predictions are suboptimal, an adaptive weighting algorithm is introduced in later section as a potential remedy. The optimizer used is Adam [32] with an initial learning rate of 10^{-3} .

The spatial coordinates for a spherical electrode are normalized against the radius of the electrode, and the dimensionless distance is given by $R = \frac{r}{r_e}$ whilst the simulation space is $1 \leq R \leq 1 + 6\sqrt{T_{\text{sim}}} = R_{\text{sim}}$. Note that $R = 1$ represents the surface of a spherical electrode. The PDEs and boundary conditions for PINN simulation of spherical electrode are:

$$\frac{\partial C}{\partial T} - \frac{\partial^2 C}{\partial R^2} - \frac{2}{R} \frac{\partial C}{\partial R} = 0 \text{ on } \mathcal{T} \times \Omega_R \quad (11.1)$$

$$C = f(T) \text{ on } \mathcal{T}, R = 1 \quad (11.2)$$

$$C = 1 \text{ on } \mathcal{T}, R = R_{\text{sim}} \quad (11.3)$$

$$C = 1, \text{ on } T = 0, \Omega_R \quad (11.4)$$

where $\Omega_R \in [1, R_{\text{sim}}]$.

It is also possible to convert the radial diffusion equation to apparent linear diffusion by a mathematical substitution [33], letting $U = CR$, the PDEs and other conditions are:

$$\frac{\partial U}{\partial T} - \frac{\partial^2 U}{\partial R^2} = 0 \text{ on } \mathcal{T} \times \Omega_R \quad (12.1)$$

$$U = f(T) \times 1 \text{ on } \mathcal{T}, R = 1 \quad (12.2)$$

$$U = 1 \times R_{\text{sim}} \text{ on } \mathcal{T}, R = R_{\text{sim}} \quad (12.3)$$

$$U = 1 \times R, \text{ on } T = 0, \Omega_R \quad (12.4)$$

By transforming radial diffusion into apparent linear diffusion, the PDEs may be easier for PINNs to solve. Their comparative performance for PINN prediction is discussed in a later section.

The diffusion equation for a 2D microband electrode is:

$$\frac{\partial C}{\partial T} - \frac{\partial^2 C}{\partial X^2} - \frac{\partial^2 C}{\partial Y^2} = 0 \text{ on } \mathcal{T} \times \Omega_X \times \Omega_Y \quad (13)$$

where $\Omega_Y \in [0, Y_{\text{sim}}]$.

The diffusion equation in 3D for microdisc and cube electrodes is:

$$\frac{\partial C}{\partial T} - \frac{\partial^2 C}{\partial X^2} - \frac{\partial^2 C}{\partial Y^2} - \frac{\partial^2 C}{\partial Z^2} = 0 \text{ on } \mathcal{T} \times \Omega_X \times \Omega_Y \times \Omega_Z \quad (14)$$

$$\Omega_Z \in [0, Z_{sim}]$$

3. Simulation methods

Simulations in 1D, 2D and 3D using PINN or FDM were performed with different computing environment. 1D PINN simulations were performed with two E5-2640 CPU and 16 GB of RAM; 2D PINN simulations and 3D FDM simulations of a cube electrode were run on a workstation with an Intel I7 CPU and Nvidia P100 acceleration card with 32 GB of RAM. 3D PINN simulations were performed with 18 CPU cores, two Nvidia V100 acceleration card and 90 GB of RAM on Advanced Research Computing (ARC) at University of Oxford. PINNs were written in Python with TensorFlow [34] implementation of neural networks. 3D FDM simulations were written in C, compiled using NVCC compiler and the diffusion equations were solved using Hypr (Lawrence Livermore National Laboratory, CA, version 2.18.2) [35,36].

4. Results and discussion

In this section the PINN simulation of a broad range of electrode geometries, from 1D simulation of a macroelectrode to 3D simulation of a cube electrode, are used to analyze the pros, cons and best practices of PINN simulation in voltammetry. The first investigation reported is the 1D simulation of chronoamperometry at a macroelectrode, to showcase that a nonzero conditioning time, is required for simulation of chronoamperometry. The second example, 1D simulation of chronoamperometry at a (hemi-) spherical electrode, shows that transition from a harder physics problem (radial diffusion) into easier one (apparent linear diffusion) by a mathematical transformation of the diffusion equation decreases the learning barrier of PINN and increases the accuracy of extrapolation substantially. Such mathematical transformation, wherever possible, is beneficial to increase accuracy of extrapolation. The third scenario, 1D simulation of cyclic voltammetry at a (hemi-)spherical electrode with low scan rates, introduces sequence-to-sequence training to tackle the failure of original PINN approach to the simulation of longer time duration. 1D simulation of cyclic voltammetry at a macro electrode is the fourth scenario to evaluate the use of adaptive learning weights algorithm to facilitate convergence and/or increase accuracy. The effect of batch size on the accuracy of 2D simulation of chronoamperometry at a microband electrode are examined in the fifth scenario. The sixth and seventh scenarios cover the 3D simulation of chronoamperometry and cyclic voltammetry at a cube electrode, with former having a known solution against which output can be checked whilst the cyclic voltammetry problem has previously not been attempted to the author's knowledge. Domain decomposition and learning rate scheduler are introduced to increase the accuracy of simulation. The eighth problem addresses the difficulty of PINN simulation of the microdisc electrode when solved using cylindrical coordinates in 2D due to the gradient explosion problem, and introduced 3D simulation of the microdisc electrode in Cartesian space as a better alternative. Lastly, an analysis of comparative simulation times and hardware requirements was conducted to evaluate the pros and cons of PINN relative to conventional FDM.

4.1. Chronoamperometry at a macroelectrode

A PINN generates a concentration profile $C(T, X)$ for macroelectrodes or $C(T, R)$ for hemi-spherical electrodes by enforcing physical laws (PDEs), boundary conditions and initial conditions. As will be demonstrated below, when simulating a single-step chronoamperogram it can be beneficial to additionally consider the system prior to

the onset of the electrode reaction. This time prior to the onset of the electrode reaction will be referred to here as the conditioning time. Consequently, in the following for single-step chronoamperometry, the simulation starts at $T = 0$, and stays at a low overpotential where no electrochemical reaction occurs. Then at $T = T_{step}$, a high overpotential is applied such that the concentration at electrode surface is 0. The simulation time domain then made up of three time durations: conditioning time, $\mathcal{T}_{conditioning} \in [0, T_{step}]$, interpolation time, $\mathcal{T}_{interp} \in [T_{step}, T_{sim}]$ and an optional extrapolation time $\mathcal{T}_{extrap} \in [T_{sim}, T_{extrap}]$. Note that training is between $\mathcal{T} \in [0, T_{sim}]$ with optional extrapolation to T_{extrap} after training. $f_{CA}(T)$, the electrode surface condition for chronoamperometry is modified and integrated in eq (7.2) and (11.2) as:

$$f_{CA}(T) = \begin{cases} T < T_{step}, C = 1 \\ T \geq T_{step}, C = 0 \end{cases} \quad (15)$$

From these PINN simulations, a full 2D concentration profile was returned. For example, a PINN simulation with a $T_{step} = 0.1$ and a $T_{sim} = 1$ for a macroelectrode was trained for 300 epochs. Each of the four sets of collocation points mentioned in eq 7 had $N = 2 \times 10^5$ collocation points randomly distributed without any preferred location. The PINN had three fully-connected layers with 32 neurons in each layer and the activation function was a hyperbolic tangent (tanh). Fig. 2A shows the resulting concentration profile as a function of the dimensionless space (X) and time (T). The observed decrease in concentration near the electrode is expected as the high overpotential reduces the surface concentration of the reactant. Further, the PINN extrapolated concentration profile continues this pattern. Importantly, evaluation of this concentration profile allows the flux and hence current at the electrode surface to be numerically assessed. Where the flux is estimated using $J = \frac{C_{T,X=dX} - C_{T,X=0}}{dX}$, where

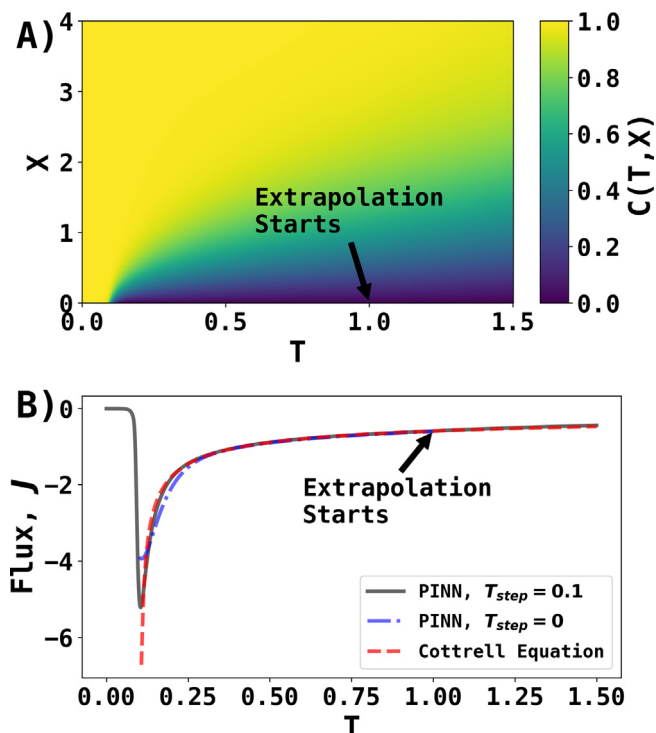


Fig. 2. PINN prediction of chronoamperogram at a macroelectrode. PINN was trained from $T = 0$ to $T = 1$. (A) The predicted concentration profile with $T_{step} = 0.1$. (B) The predicted chronoamperograms for $T_{step} = 0.1$ (black) and $T_{step} = 0$ (blue) were compared with the Cottrell equation (red). Only $T_{step} = 0.1$ transient is extrapolated to $T = 1.5$.

dX is a small spatial step and equal to 10^{-3} in this scenario. The chronoamperometric current time profile for a macroelectrode is a known exact result (the Cottrell equation); consequently, in the following we consider the accuracy of the PINN simulation by comparison to this known result.

Two PINN simulations for chronoamperometry with a potential step at $T_{step} = 0$ (no conditioning time) and $T_{step} = 0.1$ were simulated and the resulting dimensionless flux at the electrode surface is plotted as a function of time, as shown in Fig. 2B as the blue ($T_{step} = 0$) and black ($T_{step} = 0.1$) lines respectively. To enable direct comparison of these two results the blue line has been shifted by to $T = 0.1$. Also overlaid is the exact result in the form of the Cottrell equation [37] (dashed red line). The Cottrell equation, before being adjusted for step time, gives the analytical shape of chronoamperogram:

$$J = -\frac{1}{\sqrt{\pi T}} \quad (16)$$

First, as can be seen from equation (16), at $T=0$ the interfacial flux is analytically predicted to be infinitely high. Second, as can be seen from Fig. 2B at the onset of the interfacial reaction ($T = 0.1$ in this example) neither of the simulations provide an accurate assessment of the expected interfacial flux. Third, the simulation in which a conditioning time has been used improves the accuracy of the simulation in this short time regime. Using the known interfacial flux (eq (16)) the accuracy of the simulation can be assessed where the error of interpolated chronoamperogram is calculated by the arithmetic average of relative deviation from Cottrell equation for n time points uniformly distributed between $[T_{step} + 0.05, T_{sim}]$:

$$\varepsilon = \frac{1}{n} \sum_i \frac{J_{PINN,i} - J_{Cottrell,i}}{J_{Cottrell,i}} \times 100\% \quad (17)$$

where $J_{PINN,i}$ and $J_{Cottrell,i}$ are PINN predicted flux and Cottrell equation calculated flux at time point i . The error analysis starts from $T_{step} + 0.05$ to avoid the infinite flux predicted by the Cottrell equation when $T = 0$.

In the following we assess the error in the PINN simulation. The average error with other conditioning times, $[0, T_{step}]$, is shown in Table 2, showing that with finite conditioning times, the error of prediction can be decreased from 6.33% to less than 0.5%.

With the use of non-zero conditioning times, as in the case of black transient shown in Fig. 2, the average error of the interpolated portion of chronoamperogram as $T_{step} = 0.1$ is -0.26%, and the average error of extrapolated portion of chronoamperogram is -3.8%. The relatively small errors for both interpolated and extrapolated chronoamperogram showcase the ability of PINN to make physics-consistent interpolation/extrapolation, in the context of chronoamperometry. In the case of limited computational resources and/or timeframe, physics-informed extrapolation poses an advantage of conventional FDM or FEM: it can be trained with a fraction of the spatiotemporal domain, and the rest extrapolated using negligible computing resources.

Table 2
The error of interpolated chronoamperogram as a function of T_{step} in the duration $[T_{step} + 0.05, T_{sim}]$.

T_{step}	ε
0	6.33%
0.001	0.52%
0.01	0.12%
0.1	-0.26%
0.2	-1.04%
0.3	1.76%

4.2. Chronoamperometry at a (hemi-)spherical electrode

PINN simulations of chronoamperogram at a (hemi-)spherical electrode were performed to evaluate the influence of mathematical transformation of physical laws on PINN performance. Two neural networks were built independently: Net A was governed with the radial diffusion equation in eq 11, and Net B was governed with apparent linear transformed diffusion equation in eq 12. Except for the difference in the governing PDEs and their corresponding boundary conditions, the two networks had identical structures, optimizers, loss functions and training epochs. They were both trained at $\mathcal{T} \in [0, 1]$ with $\mathcal{N} = 2 \times 10^5$ and predictions extrapolates to $T = 3$. The single potential step is imposed at $T_{step} = 0.1$ to allow for a conditioning time.

The simulation results are shown in Fig. 3. Fig. 3A shows the concentration profile predicted by Net A, where a decrease in concentration near the electrode surface ($R = 1$) when $T > 0.1$ is observed.

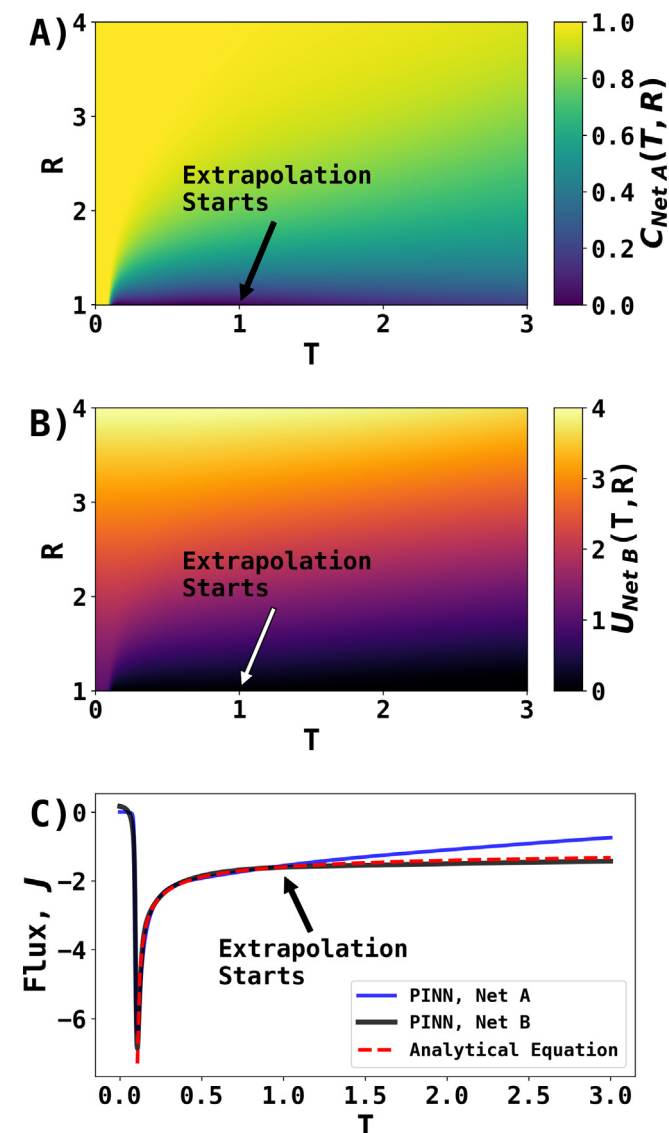


Fig. 3. PINN prediction of chronoamperogram at a (hemi-)spherical electrode. PINN was trained from $T = 0$ to $T = 1$, and the prediction of chronoamperogram extends to $T = 3$. Net A learnt the original radial diffusion equation and Net B learnt the transformed radial diffusion equation (see text). (A) The concentration profile predicted by Net A. (B) The U profile predicted by Net B, where $U = CR$. (C) Comparing the chronoamperograms predicted by Net A, Net B and analytical expression.

Fig. 3B shows the transformed property, $U = CR$, predicted by Net B. Diminishing U near the surface of electrode is also indicative of electrochemical reaction and mass transport.

To validate the two neural networks with different forms of PDEs for the same underlying physics, Fig. 3C shows the comparison between the predicted chronoamperograms extrapolated to $T = 3$ and analytical expression. To calculate the chronoamperogram by Net B, the parameter U was first converted back to C . The analytical expression for chronoamperogram at a (hemi-)spherical electrode [38], before being adjusted for the finite conditioning time, is:

$$J = -\frac{1 + \sqrt{\pi T}}{\sqrt{\pi T}} \quad (18)$$

Although both Nets performed very well when interpolating the voltammogram, Net A significantly underperformed Net B when extrapolation starts by increasingly underestimating the magnitude of flux. When interpolating, the average errors of magnitude are 0.1% and -0.6% for Net A and Net B, respectively. When extrapolating, the average errors are -23.9% and 6.1% for Net A and Net B, respectively. Based on the errors of extrapolation, it is inferred that Net B learnt the physics better. The drastic difference in extrapolation was not hard to explain: Net B learnt the simpler physics, the transformed radial diffusion, which is in effect linear diffusion. This scenario illustrates the third recommendation that transforming existing physics into simpler physics, wherever possible, may decrease the learning barriers for PINNs and boost their performance.

4.3. Cyclic voltammetry at a (hemi-) spherical electrode

By comparing PINN simulation of cyclic voltammetry at a (hemi-)spherical electrode at a low scan rate, with or without sequence-to-sequence training, we show that sequence-to-sequence training is essential to solve voltammetry problem with a long timescale. Sequence-to-sequence training, in the context of PINN, is to train small time intervals sequentially with the prediction of the last time interval becoming the initial condition of the next time interval. The weights of the neural network are reinitialized at each interval.

Simulation of voltammetry at a low scan rate, for example, $\sigma = 0.5$, requires a long simulation timeframe, $T_{sim} = 80$ if the potential window is $[-10, 10]$. In addition, larger T_{sim} will also increase X_{sim} or R_{sim} , requiring a larger number of collocation points to cover the expanded spatiotemporal domain. Krishnapriyan et al. [25], pointed out that PINN prediction of a large spatiotemporal domain at once (the original PINN approach [11]) may fail miserably, possibly due to the ill-conditioned loss landscape. They also suggested a remedy to this problem, known as sequence-to-sequence training mentioned above. The training scheme, in the context of voltammetry at a (hemi-)spherical electrode at $\sigma = 0.5$, is shown in Fig. 4. The temporal domain was divided into 10 timeframes ($\Delta T = 8$), and PINN was only trained for one ΔT at a time. Then, the prediction at ΔT was the initial condition for training the next timeframe.

To compare and validate sequence-to-sequence training on simulation of voltammetry at low scan rates, cyclic voltammetry at a (hemi-)spherical electrode was used as the challenge. Eq 11 was used as the diffusion equation. The scan rate is $\sigma = 0.5$ as mentioned above and $T_{sim} = 80$. A PINN with five layers was constructed to simulate voltammetry with/without sequence-to-sequence training for 150 epochs. To allow for fair a comparison, the total number of collocation points, \mathcal{N} is fixed at 10^6 , so PINN was trained with 10^5 collocation points for each timeframe using sequence-to-sequence training. Hence the training time is approximately equal. Fig. 5A illustrates the concentration predicted by PINN with sequence-to-sequence training. The reduced concentration near the electrode surface at $20 < T < 60$ is indicative of electrochemical reaction. Fig. 5B compares the voltammogram predicted by PINN with and without sequence-to-sequence training and

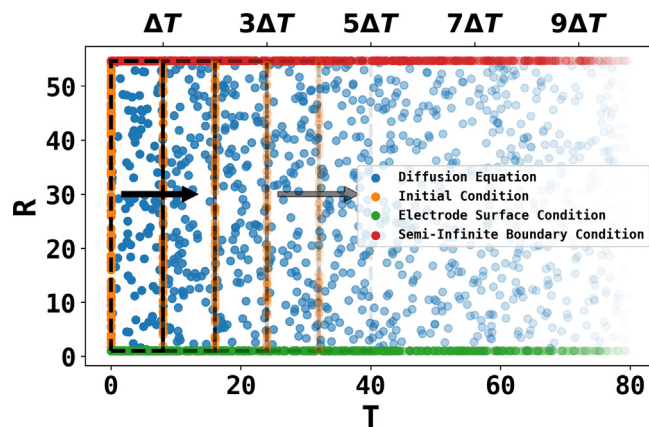


Fig. 4. Schematic scheme of sequence-to-sequence training of PINN. The simulation domain was divided into 10 timeframes, and PINN was trained sequentially as indicated by the black arrows.

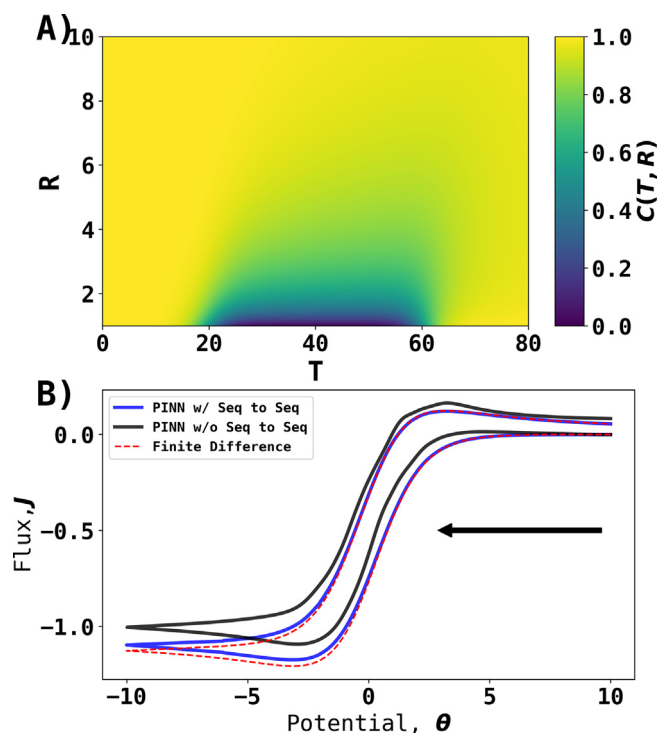


Fig. 5. Simulation of voltammetry at a (hemi-)spherical electrode when $\sigma = 0.5$. (A) Concentration profile predicted using sequence-to-sequence training. (B) Comparison of voltammogram predicted by PINN with and without sequence-to-sequence training and simulated by FDM. The black arrow indicates the start and initial direction of voltammetric scan.

the voltammogram simulated by FDM as a ground truth. It is obvious that sequence-to-sequence training has significantly increased the accuracy of the prediction, with minor deviation from ground truth when $\theta > 0$. Judging by peak flux of forward/reverse scan, sequence-to-sequence training has reduced the error from 9.5%/32.2% to 3.2%/-2.1%.

Based on the scenario of simulation of voltammetry at a low scan rate (long time span), we showed that sequence-to-sequence training is essential to generate acceptably accurate results. In addition, training with a fraction of the collocation points sequentially reduces the computer memory requirement without increasing training time, which may be hugely beneficial to long simulations for higher

dimensional problems. Thus, in general, using sequence-to-sequence training for low scan rates and longer time domains is another best practice.

4.4. Cyclic voltammetry at a macroelectrode

We next compare the performance of two identical PINNs with and without an adaptive weights algorithm, and show that, in the case of PINN simulation of cyclic voltammetry at a macroelectrode, adaptive weights can accelerate convergence and reduce training time. The adaptive weights algorithm, in the context of voltammetry, is introduced here.

Training a multi-objective loss function, like eq (10), can be a challenge for optimization algorithms [1]. For simulation of voltammetry to work, the boundary condition on the electrode surface must be strongly enforced. Balancing the weights of the loss function, using trial and error, is unsurprisingly inefficient and unsystematic. Wang et al. [26] showed that unbalanced gradients during training can adversely affect the accuracy of optimization. Steeper gradients were generally observed for PDE loss terms so an adaptive loss weights algorithm was introduced to normalize the boundary condition loss gradients to reduce the stiffness of gradient flow dynamics. To evaluate the effect of adaptive loss weights in the context of cyclic voltammetry, the performance of PINN was compared with and without the algorithm. By fixing w_f to 1, the other loss weights in eq (10) are updated after each epoch, e , using a moving average algorithm:

$$\hat{w}_i^{e+1} = \beta \hat{w}_i^e + (1 - \beta) \hat{w}_i^{e+1} \quad (19)$$

where $\beta = 0.9$ is a hyperparameter for moving average, \hat{w}_i^e and \hat{w}_i^{e+1} are weights in the current and next epoch. \hat{w}_i^{e+1} is calculated via gradient statistics:

$$\hat{w}_i^{e+1} = \frac{1}{\hat{w}_i^e} \max(|\nabla_{\vartheta} \mathcal{L}_f(\vartheta^e)|) / \text{mean}(|\nabla_{\vartheta} \mathcal{L}_i(\vartheta^e)|), i \in [\text{surf}, \text{outer}, \text{ini}] \quad (20)$$

where ∇_{ϑ} is an operator calculating the gradient of the loss in respect to the trainable parameters, ϑ , in the neural network. \mathcal{L} stands for the mean square error loss for each loss terms in eq (10).

To evaluate the effect of adaptive weighting, two identical networks were constructed except for the loss weights: Net A was constructed by setting all weights to 1, and Net B was Net A with adaptive weights to update weights in every epoch. When the dimensionless scan rate is $\sigma = 40$, both Nets were trained for 10 epochs. The difference of the predicted concentration profile is plotted Fig. 6A. Note that Net A, relative to Net B, tends to underestimate concentration near the electrode surface before $T = 0.25$ and after $T = 0.75$, and to overestimate concentration between $T = [0.25, 0.75]$. Fig. 6B compares the voltammogram predicted by both Nets, and voltammogram calculated by FDM. The peak current, J_p was validated against the dimensionless Randles–Ševčík equation, $J_p = 0.446\sqrt{\sigma}$. While voltammogram predicted by Net B agreed with both the FDM and the Randles–Ševčík equation, Net A failed as it tended to underestimate the magnitude of fluxes near $\theta = 0$, corresponding to failure to enforce boundary conditions at $T = 0.25$ for the forward scan and $T = 0.75$ for the reverse scan, when surface condition changes rapidly per the Nernst equation. On the contrary, the adaptive weights algorithm increased the weight on electrode surface condition, from $w_{\text{surf}} = 1$ at the first epoch to $w_{\text{surf}} \approx 20$ at the 10th epoch, to enforce the Nernst equation. Note that with additional training for 100 epochs, Net A can also make accurate predictions [19].

The above example shows that when trained for only 10 epochs, neural networks with adaptive weights can automatically balance the weights of the four loss functions in eq (10), and accurately resolve the electrode kinetics near the electrode surface. We thus recommend the adoption of adaptive weights, as a small add-on to PINNs, to facilitate convergence and reduce training time.

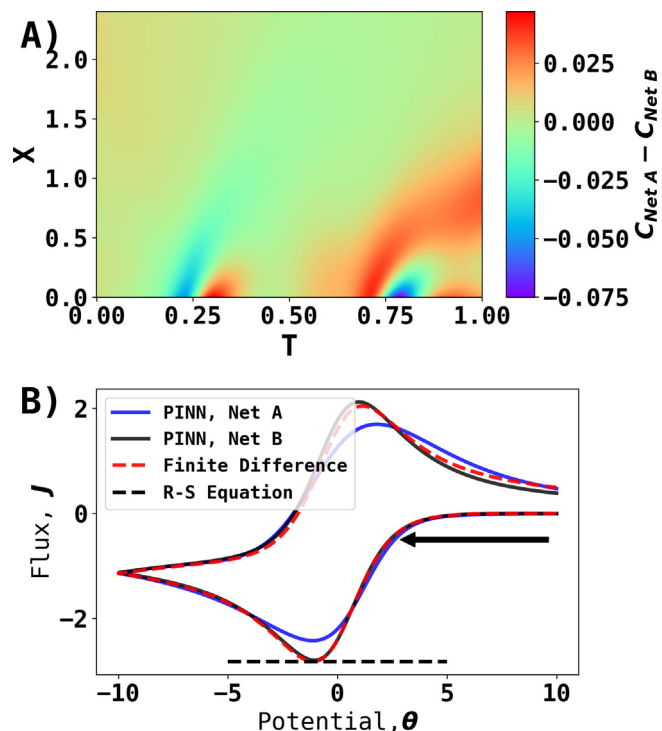


Fig. 6. Simulation of cyclic voltammetry by Net A and Net B (see text) at $\sigma = 40$. (A) The difference of concentration profile predicted by Net A and Net B. (B) Comparison between voltammograms predicted by Net A, Net B and simulated using FDM. The peak current was checked with Randles–Ševčík equation. Black arrow indicates the start and initial direction of voltammetry.

4.5. Chronoamperometry at a microband electrode

2D simulation of chronoamperometry at a microband electrode was used as a test case to evaluate the influence of batch size on the performance of PINN training by evaluating the accuracy of prediction. A PINN, as a variant of an artificial neural network, is usually trained in minibatches. In the case of interest, instead of feeding the entire dataset to the neural network at once, the PINN was trained with subsets of the dataset called minibatches with gradient calculations performed for each minibatch. A larger batch size usually takes less time per epoch as fewer gradient calculations are involved and changes to the network weights tend to be mitigated by averaging over more data points, leading to slower learning per epoch but smoother gradient descent. Compared with training with the whole dataset, using minibatches reduces memory and computational requirements, and is considered a standard practice when training large datasets [39]. The optimal size of minibatches (batch size, m), however, is still controversial: preference for a small batch size ($m < 100$) was reported for a general artificial neural network [40] and PINNs [41,42] specifically, while overperformance using a large batch size was also prevalent [43–45]. In this section experiments with different batch sizes are performed, to evaluate the effect of batch size on the PINN simulation of microband chronoamperometry, so as to provide recommendations for the starting batch size to use when approaching simulation using PINNs.

The PINN approach to voltammetry of a microband electrode includes a single diffusion domain and a mix of Dirichlet and Neumann boundary conditions:

$$\frac{\partial C}{\partial T} - \frac{\partial^2 C}{\partial X^2} - \frac{\partial^2 C}{\partial Y^2} = 0 \text{ on } \mathcal{T} \times \Omega_{X_1} \times \Omega_{Y_1} \quad (21.1)$$

$$C = f(T) \text{ on } \mathcal{T}, X = 0, Y = [0, 0.5] \quad (21.2)$$

$$\frac{\partial C}{\partial X} = 0 \text{ on } \mathcal{T}, X = 0, Y = [0.5, Y_{sim}] \quad (21.3)$$

$$\frac{\partial C}{\partial Y} = 0 \text{ on } \mathcal{T} \times \Omega_{X_1}, Y = 0 \quad (21.4)$$

where $\Omega_{X_1} \in [0, X_{sim}]$, $\Omega_{Y_1} \in [0, Y_{sim}]$. Eq (21.1) enforces the diffusion equation of the entire spatiotemporal domain. Eq (21.2) enforces the electrode kinetics for the boundary condition on the electrode surface. Note that because of symmetry along the central axis of a microband electrode, only half of the electrode is simulated, and the electrode is located at $X = 0, Y = [0, 0.5]$. Eq (21.3) and (21.4) enforce the no-flux boundary condition for the insulating surface and axis of symmetry respectively. The number of collocation points \mathcal{N} for each constraint is 2×10^6 . The PINN had 5 layers and was trained for 100 epochs with different batch sizes from 24 to 3.2×10^5 . The predicted chronoamperograms were compared with the Aoki equation [46,47] (with T adjusted for T_{step}) for a microband electrode:

$$J = -\left(\frac{1}{\sqrt{\pi T}} + 1 - \left(\frac{2\sqrt{3}}{\pi}\right) T^{\frac{3}{4}} \exp\left[-\frac{1}{8T}\right] \mathcal{U}\left(2, \frac{1}{\sqrt{2T}}\right)\right) \quad (22)$$

where \mathcal{U} is the parabolic cylinder function [31].

Table 3 tabulates the errors of the PINN prediction relative to the analytical expression as a function of batch size, m , and reveals a large magnitude of error at both extremes of batch size. The magnitude of error is smallest at -1.7% when $m = 4000$. A batch size within the range of 10^3 to 10^5 gives more accurate predictions with the magnitude of errors below 5%. Fig. 7A shows the concentration profile predicted by PINN when $m = 4000$ and $T = 0.5$. Fig. 7B compares chronoamperograms predicted at different batch sizes ($m = 24$ and 4000) with the analytical expression. At $m = 24$, the chronoamperogram is very noisy compared to $m = 4000$ and the magnitude of current is significantly underestimated. At $m = 4000$, the error of interpolated/extrapolated chronoamperogram is -1.7%/-2.9% respectively. Batch sizes that were too small performed poorly possibly due to the difficulty of converging towards the global minimum due to noisy gradients [48] whilst batch sizes that were too large tend to converge to sharp minimizers and lead to poor generalization [49] evidenced by training history found in the *batch size and simulation of a microband electrode section* in the [Supplementary Information](#). Training with small batch sizes not only generated poor results, but also took more time: ~21 hours when $m = 24$ compared with ~1 hour when $m = 4000$. Thus, we caution against ungrounded preference for large or small batch size regardless of dataset size, but recommend to start with a batch size of 4000, in the context of PINN simulation of chronoamperometry, when the number of collocation points, \mathcal{N} is around a million, or 0.2% of the dataset size as \mathcal{N} varies over orders of magnitude. Users can then systematically increase or decrease the

Table 3

The error of the interpolated chronoamperogram as a function of batch size after 100 epochs.

Batch Size, m	Error of Predicted Chronoamperogram
24	-15.9%
32	-30.5%
64	-5.3%
100	-10.1%
200	-20.9%
400	-8.2%
800	-5.9%
1000	-1.9%
4000	-1.7%
8000	-4.0%
16000	-2.4%
64000	-4.5%
128000	-6.3%
256000	-17.9%

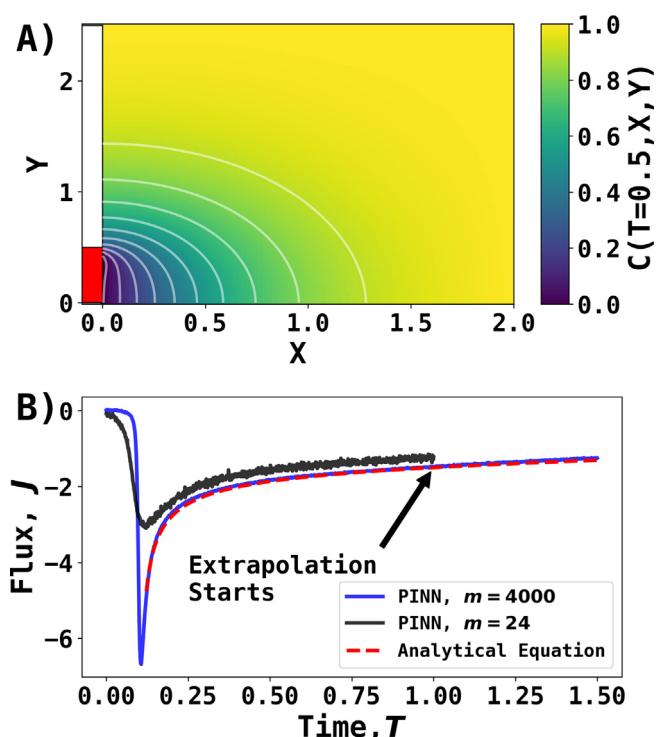


Fig. 7. PINN prediction of chronoamperometry at a microband electrode and the influence of batch size, m . (A) The predicted concentration profile at $T = 0.5$ and $m = 4000$. The red block and white blocks represent the surfaces of the microband electrode and insulating material respectively. (B) The chronoamperograms predicted by PINN with $m = 4000$ and $m = 24$ are compared with the analytical expression (see text).

batch size while checking the accuracy and stability of PINN simulations by tracking losses as training progresses.

4.6. Chronoamperometry at a cube electrode

The three-dimensional simulation of chronoamperometry at a cube electrode was performed using PINNs. Identical PINNs were trained with and without overlapping domain decomposition and by comparing chronoamperograms simulated using FDM, it is shown that overlapping domain decomposition is essential for PINN simulation of chronoamperometry at a cube electrode.

First, PINN simulation without domain decomposition was performed. Because of symmetry, only one eighth of the cube was simulated as shown in Fig. 8 and the simulated part of the cube has three electrochemically active faces parallel to the XY, XZ and YZ planes respectively. The dimensionless edge length of the simulated cube is 1. To enforce the diffusion mass transport, a total number of 6×10^7 of collocation points (shown in Fig. 9A) were generated randomly distributed in the spatiotemporal domain:

$$\frac{\partial C}{\partial T} - \frac{\partial^2 C}{\partial X^2} - \frac{\partial^2 C}{\partial Y^2} - \frac{\partial^2 C}{\partial Z^2} = 0 \text{ on } \mathcal{T} \times \Omega_X \times \Omega_Y \times \Omega_Z \text{ and } (X > 1 | Y > 1 | Z > 1) \quad (23)$$

where $(X > 1 | Y > 1 | Z > 1)$ ensures that the collocations points were not inside or on the cube (dimensionless edge length = 1). A neural network with 6 hidden layers was constructed and trained for 300 epochs and the predicted chronoamperograms for the three faces were compared with results from FDM [50] as shown in Fig. 9B. Although the transients for the three faces overlap each other, suggesting that the PINN “understood” the symmetry in this system, they clearly under-

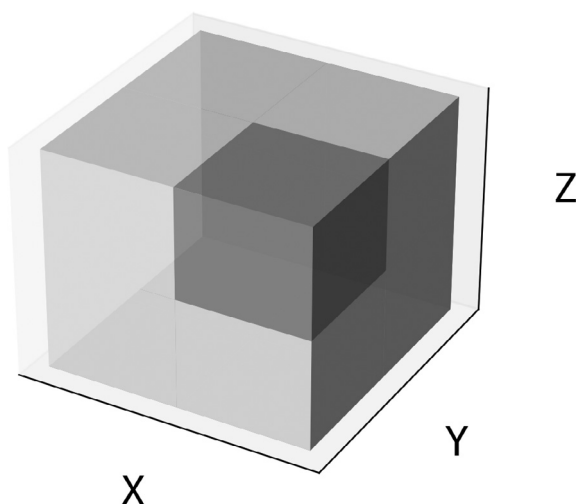


Fig. 8. Because of symmetry, only one eighth of a cube is simulated. It has three electrochemically active faces shown in grey.

estimate the magnitude of the expected fluxes suggesting that the spatiotemporal domain may be too complicated to be solved altogether.

Inspired by domain decomposition utilized to solve problems of linear algebra in parallel [51], the global diffusion domain mentioned above was divided into six overlapping subdomains. Using overlapping subdomains was advantageous as it does not need to account for interfaces specifically as the subdomains overlap by more than their interface. Fig. 9C shows two subdomains along the Z-axis:

$$\frac{\partial C}{\partial T} - \frac{\partial^2 C}{\partial X^2} - \frac{\partial^2 C}{\partial Y^2} - \frac{\partial^2 C}{\partial Z^2} = 0 \text{ on } \mathcal{T} \times \Omega_{X_1} \times \Omega_{Y_1} \times \Omega_Z \quad (24.1)$$

$$\frac{\partial C}{\partial T} - \frac{\partial^2 C}{\partial X^2} - \frac{\partial^2 C}{\partial Y^2} - \frac{\partial^2 C}{\partial Z^2} = 0 \text{ on } \mathcal{T} \times \Omega_{X_2} \times \Omega_{Y_2} \times \Omega_Z \quad (24.2)$$

where $\Omega_{X_1} \in [1, X_{sim}]$, $\Omega_{Y_1} \in [0, Y_{sim}]$, $\Omega_{X_2} \in [0, X_{sim}]$ and $\Omega_{Y_2} \in [1, Y_{sim}]$. Eq (24.1) and (24.2) correspond to the overlapping green and blue domains shown in Fig. 9C. The other four subdomains along the X and Y axis are available in the simulation of a cube electrode section Supplementary Information. Since the six subdomains are overlapping, there is no need to address interface conditions between domains. To allow fair comparison with a large domain with $\mathcal{N} = 6 \times 10^7$, the number of collocation points used was 10^7 in each of the six subdomains. The same PINN was trained with domain decomposition for 300 epochs. Fig. 9D illustrates the fluxes predicted by PINN with domain decomposition: the symmetry was enforced evidenced by overlapping fluxes on each surface, and the agreement with FDM showed that overlapping domain decomposition can help PINN to converge to the correct result, possibly due to a better ability to accurately capture the steep descents of concentration in solution [52]. The error of PINN simulation was -3.3% compared with FDM [50]. The flux density at $T = 0.5$ at the electrode surface is shown in Fig. 10 The flux density is highest near the edge of the cube, which shows characteristics of convergent diffusion, and lowest at the surface of the electrode distant from the cube edges, which shows characteristics of linear diffusion. Note that FDM simulation revealed that the maximum flux occurred at the edges and corner [50], suggesting that PINN might not resolve local flux at edges accurately even though global flux well agreed with FDM.

The challenge of solving diffusion to a cube in 3D highlights the strengths of overlapping domain decomposition, namely significantly increasing the accuracy of PINN prediction while keeping the total number of collocation points unchanged and training time unaffected. The advantage of overlapping domain decomposition, compared with recent literature using non-overlapping domain decomposition [52,53], is the simplicity of implementation without the need to consider domain interface conditions. Although a single network was used

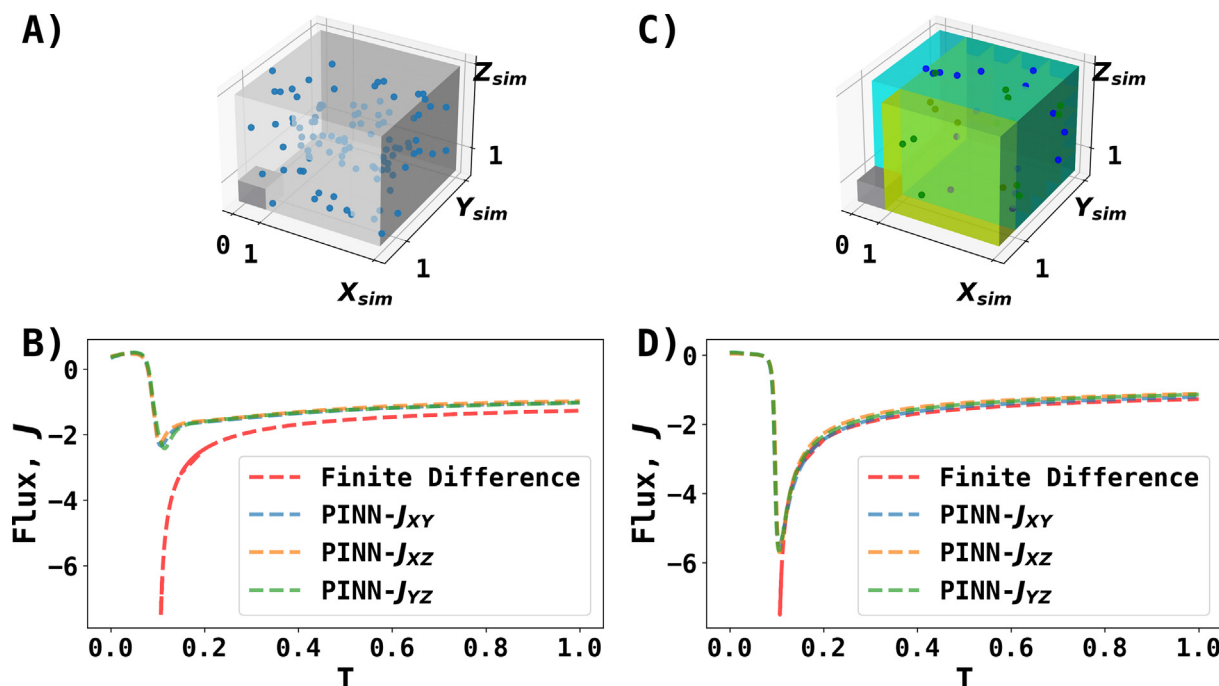


Fig. 9. Visualization of collocations points with and without overlapping domain decomposition and PINN simulation results. (A) Visualization of collocation points (blue scatter points) without domain decomposition and the electrode (small metallic cube) and (B) its predicted chronoamperograms for each surface of the electrode. (C) Visualization of collocation points for two subdomains (green and blue) along the Z-axis and two sets of collocation points in the subdomains. (D) Chronoamperograms predicted by PINN with domain decomposition for each surface of electrode. The red-dashed curves in (B, D) are the chronoamperograms simulated using finite difference method.

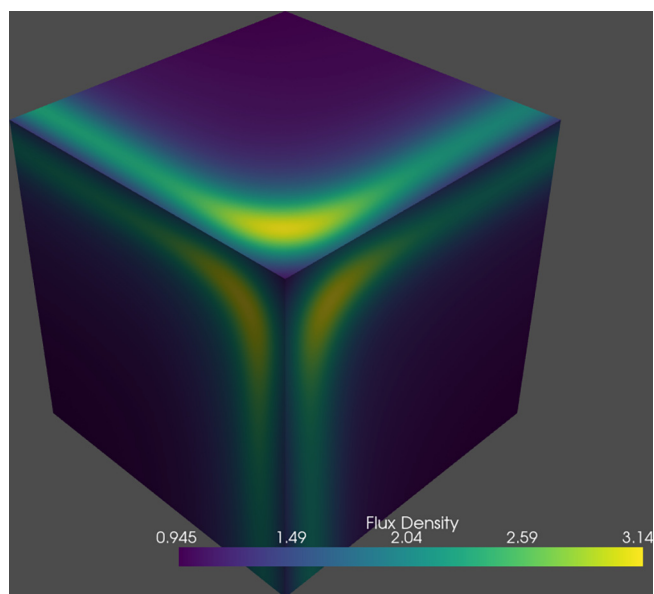


Fig. 10. Flux density at $T = 0.5$ predicted by PINN with overlapping domain decomposition for chronoamperometry at a cubic electrode.

for all the six subdomains, training a network for each subdomain is possible for parallel computing to save time [54]. In summary, an overlapping domain decomposition, is recommended when solving higher dimensional problems with complicated boundary conditions for better accuracy.

4.7. Cyclic voltammetry at a cube electrode

Simulation of cyclic voltammetry at a cube electrode using PINN, is a tougher challenge than simulation of chronoamperometry: PINN must adapt continuous changes in the electrode surface concentration which are particularly rapid near the formal potential as required by the Nernst equation. In the following, we note the importance of implementing learning rate decay at the later stage of training to help PINN converge to an analytical level of accuracy.

Voltammetry at a cube electrode at scan rates from $\sigma = 10$ to 80 was simulated. First, a PINN with 6 layers was constructed to predict voltammetry at $\sigma = 40$ without a learning rate (lr) scheduler for 350 epochs. During training, the learning rate stays at $\text{lr} = 10^{-3}$ and the number of collocation points $\mathcal{N} = 4 \times 10^7$. The second simulation was run with the same conditions, but for comparative purposes with learning rate schedule

$$\begin{cases} e \leq 200, \text{lr}_e = 10^{-3} \\ e > 200, \text{lr}_e = \text{lr}_{e-1} * \alpha \end{cases} \quad (25)$$

where α is the learning rate decay rate set at 0.98 and e is the number of epoch. Domain decomposition as discussed above for the cube electrode was used in both simulations to increase accuracy. Fig. 11A evaluates the impact of learning rate scheduling by the training history. Spikes in training losses started to disappear after 200 epochs when learning rate scheduling was employed, leading to more stable convergence to local minima than without. The resulting voltammograms are compared using FD simulation as “ground truth” in Fig. 11B. Although the learning rate scheduler did not impact the overall shape of voltammogram, learning rate scheduling increased the accuracy near the forward scan peak. Judging by forward scan peak flux, learning rate scheduling reduced the absolute error from 4.2% to 0.3%, showing in particular great improvement near $\theta = 0$.

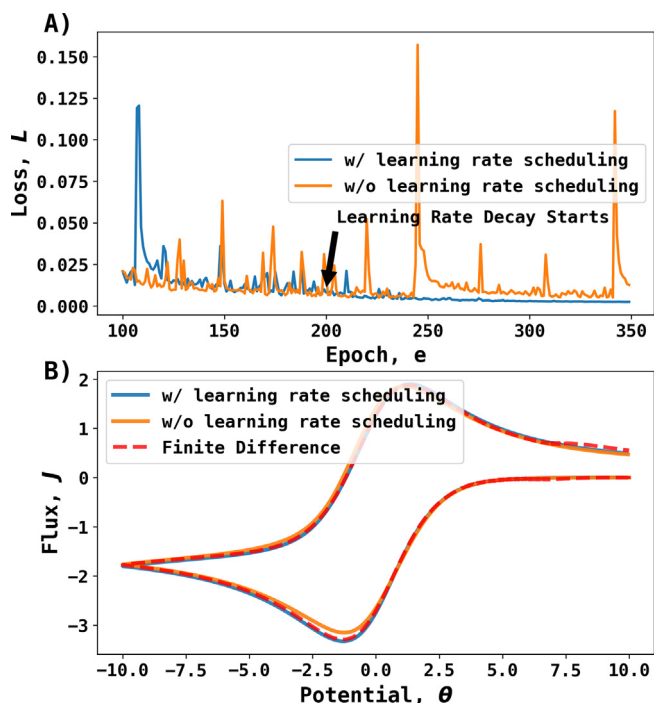


Fig. 11. PINN simulation of cyclic voltammetry at a cube electrode at $\sigma = 40$ with (blue traces) and without (orange traces) a learning rate scheduler. (A) The training loss from $e = 100$. (B) The voltammograms simulated by PINN and FDM (red-dashed).

With the learning rate schedule shown in eq (25) and domain decomposition, various PINN simulations (solid lines) of cyclic voltammetry at a cube for scan rates of $\sigma = 10$ to 80 were performed and shown in Fig. 12A and compared with FD simulations (dashed lines).

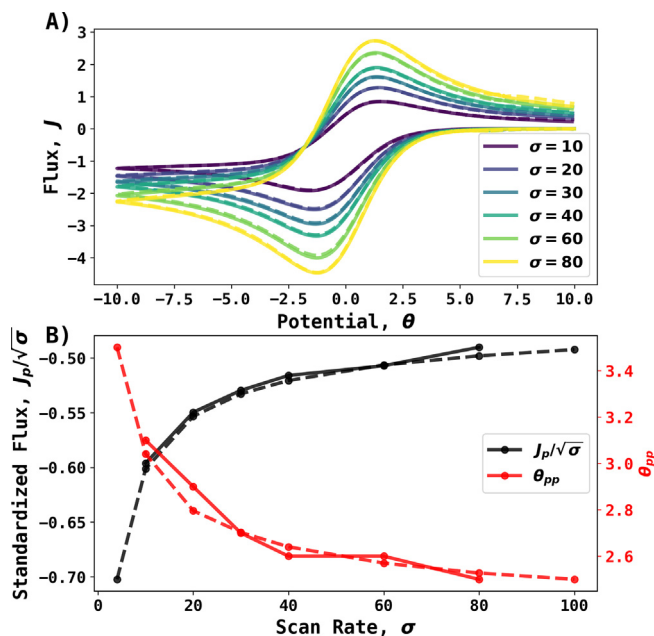


Fig. 12. PINN simulation of cyclic voltammetry at a cube at different scan rates, σ . (A). Cyclic voltammograms predicted by PINN (solid lines) and voltammograms predicted by FDM (dashed lines) from $\sigma = 10$ to 80. (B) Working curves for peak flux standardized by square root of scan rates ($J_p/\sqrt{\sigma}$, black curves) and peak to peak potential (θ_{pp} , red curves). Solid curves and dashed curves are from PINN and FDM, respectively.

Good agreement is apparent. Fig. 12B compares the working curves of forward scan peak fluxes standardized by square root of scan rates, $J_p/\sqrt{\sigma}$ and peak to peak separation, θ_{pp} . PINN predictions of $J_p/\sqrt{\sigma}$ (black-solid curve) have good agreement with FD (black-dashed curve) as do predictions of θ_{pp} (PINN red-solid; FD red-dashed curve).

This test case not only reports success of PINN simulation cyclic voltammetry at a cube electrode, not reported before to the authors' knowledge, but also, generically, stresses the importance of learning rate scheduling to achieve electroanalytical level of accuracy. We thus recommend a learning rate scheduler, like eq (25), to increase accuracy of PINN simulation for more complicated problems.

4.8. Chronoamperometry at a microdisc electrode

This test case addresses and resolves the difficulty of 2D PINN simulation of a microdisc electrode in cylindrical coordinates by introducing 3D simulation in Cartesian coordinates.

FD simulation of the microdisc electrode is usually performed in 2D using cylindrical coordinates reflecting the symmetry of the system [55]. The dimensionless diffusion equation is:

$$\frac{\partial C}{\partial T} = \frac{\partial^2 C}{\partial R^2} + \frac{1}{R} \frac{\partial C}{\partial R} + \frac{\partial^2 C}{\partial Z^2} \text{ on } \mathcal{T} \times \Omega_R \times \Omega_Z \quad (26)$$

where $\Omega_R \in [0, R_{sim}]$, $\Omega_X \in [0, Z_{sim}]$. PINN simulation of this equation, however, is more challenging: because $\lim_{R \rightarrow 0} \frac{1}{R} = \infty$, the PINN faces an exploding gradient problem when R approaches zero: the model weights became very large to Not A Number (NaN) values during training. Instead of tuning the network or transforming the diffusion equation to avoid the exploding gradient problem, 3D simulation in Cartesian coordinate (eq (14)) is suggested to take advantage of the simplicity of PINN simulation in higher dimensions. Because of the symmetry of the microdisc electrode, only a quarter of the circular electrode is simulated as shown in Fig. 13. The boundary condition for the electrode surface and insulating material near the surface is:

$$C = f(T) \text{ on } \mathcal{T}, X = \Omega_{R_e} \sin \Omega_{\theta_e}, Y = \Omega_{R_e} \cos \Omega_{\theta_e} \quad (27.1)$$

$$\frac{\partial C}{\partial Z} = 0 \text{ on } \mathcal{T} \times \Omega_X \times \Omega_Y \text{ and } (X^2 + Y^2 > 1) \quad (27.2)$$

Eq (27.1) enforces the electrode surface boundary condition and $\Omega_{R_e} \in [0, 1]$ and $\Omega_{\theta_e} \in [0, \frac{\pi}{2}]$ as the dimensionless radius of electrode is 1 and only a quarter of electrode is simulated. Eq (27.2) enforces the no-flux boundary condition and $(X^2 + Y^2 > 1)$ ensures that the collo-

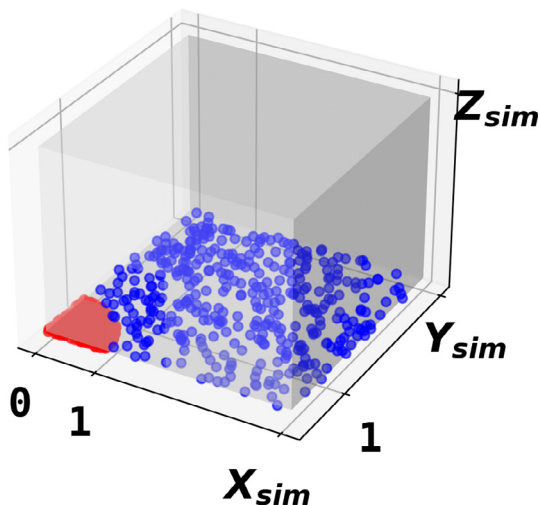


Fig. 13. Collocation points for the surface of microdisc electrode (red) and the insulating surface (blue).

cation points for boundary condition is not on the electrode surface. The number of collocation points used was $\mathcal{N} = 10^8$ therefore data generator was utilized in the neural network to generate training data in minibatches on demand and was trained for 800 epochs. A learning rate scheduler was used to decrease learning rate by 1% ($\alpha = 0.99$) every epoch after 400 epochs. The concentration profile predicted by PINN at $T = 0.5$ at the electrode surface ($Z = 0$) is shown in Fig. 14A and the concentric wedges in the concentration contour confirm the axial symmetry of the system. The chronoamperogram predicted by PINN, as shown in Fig. 14B was compared with Shoup-Szabo equation [56,57], an analytical expression for chronoamperometry at a microdisc electrode. The dimensionless form of the Shoup-Szabo equation is:

$$J = - \left(0.7854 + \frac{0.4431}{\sqrt{T}} + 0.2146 \exp \left(\frac{0.39115}{\sqrt{T}} \right) \right) \quad (28)$$

The proximity with analytical expression shown in Fig. 14B, which suggests that transforming a 2D problem in cylindrical coordinates into a 3D problem in Cartesian coordinates can avoid exploding gradient problems. Such transformation from 2D to 3D is made possible due to the ease of implementing PINN for higher dimensional problems. Thus, "brute force" simulation in higher dimensions with a different coordinate system is recommended to be explored when simulation in lower dimension encounters obstacles.

4.9. Computational time

The major advantage of PINN simulation, compared with FDM and FEM, is its simplicity of implementation by avoiding mesh generation and discretization of PDEs. But such simplicity may be a trade-off with computational time. Table 4 compares simulation time using PINN with that of FDM and the hardware. PINN was comparably efficient in 1D and 2D simulations. Taking simplicity of implementation into

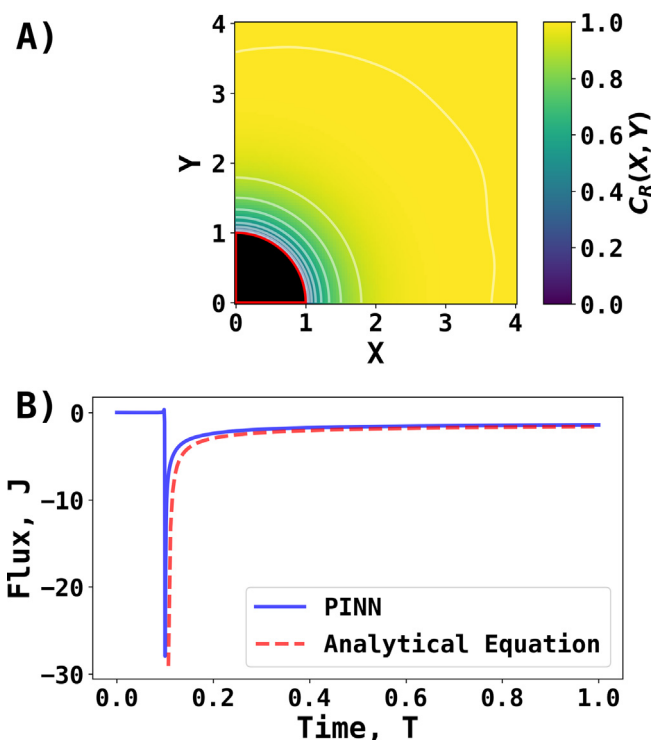


Fig. 14. PINN simulation of chronoamperometry at a microdisc electrode in 3D cartesian coordinate. (A) Concentration profile at $Z=0$ and $T=0.5$. The black wedge represents a quarter of the electrode. (B) Comparing chronoamperogram predicted by PINN with analytical equation (see text).

Table 4

Comparison of PINN simulation time with FDM. Both aim to achieve accuracy within 5%.

Scenario	PINN	Finite Difference
1D simulation of cyclic voltammetry/chronoamperometry	~ 5/15 mins, w/o adaptive weights ^a	< 1 min ^a
2D simulation of microband electrode	~ 1 hour ^b	~ 10 mins ^b
3D simulation of cube electrode	~ 36 hours ^c	~ 2 hours ^b
2D simulation of microdisc electrode	N/A	~ 1 hour ^b
3D simulation of microdisc electrode	~ 100 hours ^b	N/A

^a Intel Xeon E5-2640 v4 * 2, 16 GB RAM

^b Intel Core 6700K, Nvidia P100 16GB, 32 GB RAM

^c 18 CPU cores, Nvidia V100 32 GB * 2, 90 GB RAM (on a cluster)

consideration, discretization-free simulation offered by PINN appears more advantageous than the conventional FDM. When challenged by 3D simulations, the training time and hardware requirements increased markedly for the PINN approach, though implementation remains relatively easy. We thus recommend additional considerations including access to hardware and time sensitivity when deciding between finite difference/finite element and PINN for 3D simulations.

5. Conclusions

In 1D and 2D simulations, PINN generally achieved accuracies of ~0.5% and ~2% respectively, with comparable simulation time to the FDM, suggesting an overall comparative advantage of PINN considering the relative ease of implementation. Another major advantage of PINN is the possibility of using physics-informed extrapolation and the accuracy of extrapolated chronoamperograms was within ~1%/3% for 1D/2D simulations. Thus, when computing resources are limited and/or tasks are time sensitive, it is possible to simulate/interpolate part of the process and extrapolate the rest, a feature unavailable in conventional simulations and data-driven neural networks. More importantly, the PINN simulations mentioned above were achieved using physical laws in the form of PDEs and boundary conditions only, but possess the flexibility to include experimental and/or simulation data during training to further increase the accuracy of the prediction [58]. The extension to embracing experimental data is especially valuable in the context of voltammetry.

The relative simplicity of implementation comes with the cost of increased computational time. While 3D simulation of chronoamperometry at a cube electrode can achieve an accuracy of 3.3%, simulation took significantly longer than the FDM simulation even with better hardware as shown in Table 4. PINN also suffers decreasing marginal return of additional training on level of accuracy. To achieve a level of accuracy required for analysis of experiments, PINN simulations do not work out-of-box but necessitates a fair amount of treatment and convergence study with hyperparameters, which may offset the time saved from implementation. Therefore, for new users to avoid known difficulties, one requires best practices as initial inputs, an intended key contribution of this paper.

The eight case studies reported above span situations from time-dependent one dimensional to three dimensional simulations and cover both chronoamperometry and cyclic voltammetry. These challenges serve the dual purpose of expanding the applications of PINN to more complicated electroanalytical problems and to offer best practices using PINN. First, we stressed the importance of dimensionless parameters in both FDM and PINN simulation to avoid diminishing/-exploding gradient problems. Then, the non-zero conditioning time and transformation of PDEs were introduced to increase the accuracy of chronoamperometry simulation. Sequence-to-sequence training was used to avoid inaccurate solution for cyclic voltammetry at low scan rates (long time duration) and we recommend using 10 temporal inter-

vals. The adaptive weights algorithm was used to accelerate convergence. In addition, we analyzed the effect of batch size on the accuracy of 2D simulation of chronoamperometry at a microband electrode, and recommend to start at a batch size of 0.2% of dataset size. In 3D simulations, overlapping domain decomposition and/or learning rate scheduler were recommended to facilitate electroanalytical levels of accuracy. Lastly, transformation from cylindrical coordinates to Cartesian coordinates was effective avoiding the difficulty of 2D simulation in cylindrical coordinates when R approaches zero. Of course, these best practices may be combined to further optimize the system.

PINN simulations, compared with traditional FDM or FEM, are still in their infancy. With the best practices suggested above, we hope to encourage more electrochemical scientists to explore the possibility of PINN in their physical systems, including Multiphysics simulations, assimilation of even noisy experimental data and to increase the accessibility of such simulations. Further development can be expected to include ohmic effects [59,60] and slow electron transfer kinetics, and to further consider coupled homogeneous kinetics [20].

CRedit authorship contribution statement

Haotian Chen: Conceptualization, Methodology, Software, Writing – original draft, Visualization. **Christopher Batchelor-McAuley:** Software, Writing – review & editing. **Enno Kätelhön:** Methodology, Writing – review & editing. **Joseph Elliott:** Methodology, Writing – review & editing. **Richard G. Compton:** Conceptualization, Methodology, Writing – original draft, Writing – review & editing, Supervision, Project administration.

Data availability

<https://github.com/nmerovingian/PINN-Voltammetry-Best-Practices>

Declaration of Competing Interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Joseph Elliott reports financial support was provided by Syngenta and EPSRC.

Acknowledgements

HC, CBM, JE and RGC would like to acknowledge the use of the University of Oxford Advanced Research Computing (ARC) facility in carrying out this work. <http://dx.doi.org/10.5281/zenodo.22558>. Joseph Elliott would like to acknowledge the support provided to him by Syngenta and EPSRC via the EPSRC Industrial CASE award: EP/V519741/1. Haotian Chen thanks Lady Margaret Hall for a 2022/2023 graduate scholarship.

Appendix A. Supplementary data

Supplementary data to this article can be found online at <https://doi.org/10.1016/j.jelechem.2022.116918>.

References

- [1] M. Raissi, P. Perdikaris, G.E. Karniadakis, Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations, *J. Comput. Phys.* 378 (2019) 686–707.
- [2] G.E. Karniadakis, I.G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, L. Yang, Physics-informed machine learning, *Nat. Rev. Phys.* 3 (6) (2021) 422–440.
- [3] R.G. Nascimento, K. Fricke, F.A. Viana, A tutorial on solving ordinary differential equations using Python and hybrid physics-informed neural network, *Eng. Appl. Artif. Intell.* 96 (2020) 103996.
- [4] S. Cai, Z. Mao, Z. Wang, M. Yin, G.E. Karniadakis, Physics-informed neural networks (PINNs) for fluid mechanics: a review, *Acta Mech. Sin.* 37 (12) (2021) 1727–1738.

- [5] A.D. Jagtap, Z. Mao, N. Adams, G.E. Karniadakis, Physics-informed neural networks for inverse problems in supersonic flows, arXiv preprint arXiv:2202.11821 (2022).
- [6] S. Luo, M. Vellakal, S. Koric, V. Kindratenko, J. Cui, *Parameter Identification of RANS Turbulence Model Using Physics-Embedded Neural Network*, Springer International Publishing, Cham, High Performance Computing, 2020, pp. 137–149.
- [7] C.F. Gasmi, H. Tchelepi, Physics informed deep learning for flow and transport in porous media, arXiv preprint arXiv:2104.02629 (2021).
- [8] L. Deng, Y. Pan, Application of physics-informed neural networks for self-similar and transient solutions of spontaneous imbibition, *J. Pet. Sci. Eng.* 203 (2021) 108644.
- [9] A. Arzani, J.-X. Wang, R.M. D'Souza, Uncovering near-wall blood flow from sparse data with physics-informed neural networks, *Phys. Fluids* 33 (7) (2021) 071905.
- [10] J. Kim, C. Lee, Prediction of turbulent heat transfer using convolutional neural networks, *J. Fluid Mech.* 882 (2020) A18.
- [11] J. Hong, J. Liu, Rapid estimation of permeability from digital rock using 3D convolutional neural network, *Comput. Geosci.* 24 (4) (2020) 1523–1539.
- [12] C. Cheng, G.-T. Zhang, Deep Learning Method Based on Physics Informed Neural Network with Resnet Block for Solving Fluid Flow Problems, *Water* 13 (4) (2021) 423.
- [13] A. Subramaniam, M.L. Wong, R.D. Borker, S. Nimmagadda, S.K. Lele, Turbulence enrichment using physics-informed generative adversarial networks, arXiv preprint arXiv:2003.01907 (2020).
- [14] W. Ji, W. Qiu, Z. Shi, S. Pan, S. Deng, Stiff-pinn: Physics-informed neural network for stiff chemical kinetics, *J. Phys. Chem. A* 125 (36) (2021) 8098–8106.
- [15] G.S. Gusmão, A.P. Retnanto, S.C. Da Cunha, A.J. Medford, Kinetics-Informed Neural Networks, *Catal. Today*, 2022.
- [16] S. Moon, W. Zhong, S. Yang, J. Lim, W.Y. Kim, PIGNet: a physics-informed deep learning model toward generalized drug–target interaction predictions, *Chem. Sci.* 13 (13) (2022) 3661–3673.
- [17] M.S.F. Bangi, K. Kao, J.-S.-I. Kwon, Physics-informed neural networks for hybrid modeling of lab-scale batch fermentation for β -carotene production using *Saccharomyces cerevisiae*, *Chem. Eng. Res. Des.* 179 (2022) 415–423.
- [18] S.I. Ngo, Y.-I. Lim, Solution and Parameter Identification of a Fixed-Bed Reactor Model for Catalytic CO₂ Methanation Using Physics-Informed Neural Networks, *Catalysts* 11 (11) (2021) 1304.
- [19] H. Chen, E. Kätelhön, R.G. Compton, Predicting Voltammetry Using Physics-Informed Neural Networks, *J. Phys. Chem. Lett.* 13 (2022) 536–543.
- [20] H. Chen, E. Kätelhön, R.G. Compton, The application of physics-informed neural networks to hydrodynamic voltammetry, *Analyst* 147 (2022) 1881–1891.
- [21] J. Kim, K. Lee, D. Lee, S.Y. Jin, N. Park, DPM: A Novel Training Method for Physics-Informed Neural Networks in Extrapolation, arXiv preprint arXiv:2012.02681 (2020).
- [22] X. Yang, S. Zafar, J.-X. Wang, H. Xiao, Predictive large-eddy-simulation wall modeling via physics-informed neural networks, *Phys. Rev. Fluids* 4 (3) (2019) 034602.
- [23] S. Das, S. Tesfamariam, State-of-the-Art Review of Design of Experiments for Physics-Informed Deep Learning, arXiv preprint arXiv:2202.06416 (2022).
- [24] N. Rahaman, A. Baratin, D. Arpit, F. Draxler, M. Lin, F. Hamprecht, Y. Bengio, A. Courville, On the Spectral Bias of Neural Networks, *Proceedings of the 36th International Conference on Machine Learning*, PMLR, 2019, pp. 5301–5310.
- [25] A. Krishnapriyan, A. Gholami, S. Zhe, R. Kirby, M.W. Mahoney, Characterizing possible failure modes in physics-informed neural networks, *Advances in Neural Information Processing Systems* 34 (2021).
- [26] S. Wang, Y. Teng, P. Perdikaris, Understanding and mitigating gradient flow pathologies in physics-informed neural networks, *SIAM. J. Sci. Comput.* 43(5) (2021) A3055–A3081.
- [27] F.M. Rohrhofer, S. Posch, B.C. Geiger, On the Pareto front of physics-informed neural networks, arXiv preprint arXiv:2105.00862 (2021).
- [28] D. Amini, E. Haghighat, R. Juanes, Physics-informed neural network solution of thermo-hydro-mechanical (THM) processes in porous media, arXiv preprint arXiv:2203.01514 (2022).
- [29] L. Li, Y. Li, Q. Du, T. Liu, Y. Xie, ReF-nets: Physics-informed neural network for Reynolds equation of gas bearing, *Comput. Methods Appl. Mech. Eng.* 391 (2022) 114524.
- [30] A. Einstein, On the motion of small particles suspended in liquids at rest required by the molecular-kinetic theory of heat, *J. Annalen der Physik* 17 (1905) 549–560.
- [31] A. Gil, J. Segura, N.M. Temme, Computing the real parabolic cylinder functions $U(a, x)$, $V(a, x)$, *ACM Trans. Math. Softw.* 32 (1) (2006) 70–101.
- [32] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, arXiv preprint arXiv:1412.6980 (2014).
- [33] J. Crank, *The mathematics of diffusion*, Oxford university press 1979.
- [34] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, TensorFlow: A system for large-scale machine learning, 12th symposium on operating systems design and implementation, 2016, pp. 265–283.
- [35] R.D. Falgout, U.m., Yang, hypre: A library of high performance preconditioners, in: *International Conference on Computational Science*, Springer, 2002, pp. 632–641.
- [36] R.D. Falgout, J.E. Jones, U.M. Yang, *The design and implementation of hypre, a library of parallel high performance preconditioners*, Springer, Numerical solution of partial differential equations on parallel computers, 2006, pp. 267–294.
- [37] F. Cottrell, Application of the Cottrell equation to chronoamperometry, *Z. Physik Chem* 42 (1902) 385.
- [38] N.P. Bansal, J.A. Plambeck, An aid to the interpretation of electrochemical data measured with spherical and cylindrical electrodes: corrections to the Cottrell equation, *Can. J. Chem.* 56 (2) (1978) 155–156.
- [39] A. Géron, *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems*, O'Reilly 2019.
- [40] D. Masters, C. Lusch, Revisiting small batch training for deep neural networks, arXiv preprint arXiv:1804.07612 (2018).
- [41] T. Kadeethum, T.M. Jørgensen, H.M. Nick, Physics-informed Neural Networks for Solving Inverse Problems of Nonlinear Biot's Equations: Batch Training, 54th US Rock Mechanics/Geomechanics Symposium, OnePetro, 2020.
- [42] D.R. Harp, D. O'Malley, B. Yan, R. Pawar, On the feasibility of using physics-informed machine learning for underground reservoir pressure management, *Expert Syst. Appl.* 178 (2021) 115006.
- [43] L.N. Smith, A disciplined approach to neural network hyper-parameters: Part 1–learning rate, batch size, momentum, and weight decay, arXiv preprint arXiv:1803.09820 (2018).
- [44] P.M. Radiuk, Impact of training set batch size on the performance of convolutional neural networks for diverse datasets, (2017).
- [45] S.L. Smith, P.-J. Kindermans, C. Ying, Q.V. Le, Don't decay the learning rate, increase the batch size, arXiv preprint arXiv:1711.00489 (2017).
- [46] K. Aoki, K. Tokuda, H. Matsuda, Theory of Chronoamperometric Curves for a Short Time at Microband Electrodes, *Journal of Electroanalytical Chemistry* 54(12) (1986) 1010–1017.
- [47] K. Aoki, K. Tokuda, H. Matsuda, Theory of chronoamperometric curves at microband electrodes, *J. Electroanal. Chem. Interfacial Electrochem.* 225 (1) (1987) 19–32.
- [48] Y. Zhang, H. Qu, C. Chen, D. Metaxas, Taming the noisy gradient: train deep neural networks with small batch sizes, *The Twenty-Eighth International Joint Conference on Artificial Intelligence (IJCAI)*, 2019.
- [49] N.S. Keskar, D. Mudigere, J. Nocedal, M. Smelyanskiy, P.T.P. Tang, On large-batch training for deep learning: Generalization gap and sharp minima, arXiv preprint arXiv:1609.04836 (2016).
- [50] C. Batchelor-McAuley, R.G. Compton, Diffusion to a cube: A 3D implicit finite difference method, *J. Electroanal. Chem.* 877 (2020) 114607.
- [51] B.F. Smith, Domain decomposition methods for partial differential equations, *Springer, Parallel Numerical Algorithms*, 1997, pp. 225–243.
- [52] A.D. Jagtap, E. Kharazmi, G.E. Karniadakis, Conservative physics-informed neural networks on discrete domains for conservation laws: Applications to forward and inverse problems, *Comput. Methods Appl. Mech. Eng.* 365 (2020) 113028.
- [53] E. Kharazmi, Z. Zhang, G.E.M. Karniadakis, hp-VPINNs: Variational physics-informed neural networks with domain decomposition, *Comput. Methods Appl. Mech. Eng.* 374 (2021) 113547.
- [54] K. Shukla, A.D. Jagtap, G.E. Karniadakis, Parallel physics-informed neural networks via domain decomposition, *J. Comput. Phys.* 447 (2021) 110683.
- [55] R.G. Compton, E. Laborda, E. Kaetelhoe, K.R. Ward, *Understanding voltammetry: simulation of electrode processes*, 2nd ed., World Scientific London, 2020.
- [56] D. Shoup, A. Szabo, Chronoamperometric current at finite disk electrodes, *J. Electroanal. Chem. Interfacial Electrochem.* 140 (2) (1982) 237–245.
- [57] S.R. Belding, E.I. Rogers, R.G. Compton, Potential step chronoamperometry at microdisc electrodes: effect of finite electrode kinetics, *J. Phys. Chem. C* 113 (10) (2009) 4202–4207.
- [58] Q. He, D. Barajas-Solano, G. Tartakovsky, A.M. Tartakovsky, Physics-informed neural networks for multiphysics data assimilation with application to subsurface transport, *Adv. Water Resour.* 141 (2020) 103610.
- [59] A.M. Bond, S.W. Feldberg, Analysis of simulated reversible cyclic voltammetric responses for a charged redox species in the absence of added electrolyte, *J. Phys. Chem. B* 102 (49) (1998) 9966–9974.
- [60] I.B. Goldberg, A.J. Bard, Resistive effects in thin electrochemical cells: digital simulations of current and potential steps in thin layer electrochemical cells, *J. Electroanal. Chem. Interfacial Electrochem.* 38 (2) (1972) 313–322.