Medical Engineering and Physics 123 (2024) 104092

Contents lists available at ScienceDirect

Medical Engineering and Physics

journal homepage: www.elsevier.com/locate/medengphy

Exploring the potential of Physics-Informed Neural Networks to extract vascularization data from DCE-MRI in the presence of diffusion

D. Sainz-DeMena, M.A. Pérez, J.M. García-Aznar *

Department of Mechanical Engineering, Aragon Institute for Engineering Research (I3A), University of Zaragoza, Zaragoza, Spain

ARTICLE INFO

Keywords:
Physics-Informed Neural Networks
DCE-MRI
Inverse problems
Extended Tofts model

ABSTRACT

Dynamic contrast-enhanced magnetic resonance imaging (DCE-MRI) is widely used to assess tissue vascularization, particularly in oncological applications. However, the most widely used pharmacokinetic (PK) models do not account for contrast agent (CA) diffusion between neighboring voxels, which can limit the accuracy of the results, especially in cases of heterogeneous tumors. To address this issue, previous works have proposed algorithms that incorporate diffusion phenomena into the formulation. However, these algorithms often face convergence problems due to the ill-posed nature of the problem.

In this work, we present a new approach to fitting DCE-MRI data that incorporates CA diffusion by using Physics-Informed Neural Networks (PINNs). PINNs can be trained to fit measured data obtained from DCE-MRI while ensuring the mass conservation equation from the PK model. We compare the performance of PINNs to previous algorithms on different 1D cases inspired by previous works from literature. Results show that PINNs retrieve vascularization parameters more accurately from diffusion-corrected tracer-kinetic models. Furthermore, we demonstrate the robustness of PINNs compared to other traditional algorithms when faced with noisy or incomplete data. Overall, our results suggest that PINNs can be a valuable tool for improving the accuracy of DCE-MRI data analysis, particularly in cases where CA diffusion plays a significant role.

1. Introduction

Dynamic contrast-enhanced magnetic resonance imaging (DCE-MRI) is a widely used imaging technique that provides information on tissue perfusion and permeability of different tissues. By analyzing the uptake and washout of a contrast agent (CA) in a tissue, DCE-MRI can help diagnose and monitor various pathologies, particularly tumors [2,47,26]. To extract quantitative information of these tissues, the voxel-wise CA concentration versus time curves need to be fitted to some of the pharmacokinetic (PK) models available [17]. Although there are multiple PK models, some of the most employed models are the standard Tofts model (STM) and the extended Tofts model (ETM) [39,38]. Both are compartmental models that incorporate two distinct compartments - the intravascular and extravascular-extracellular space (EES) - and account for the exchange of CA between them. Unlike the STM, the ETM considers that a fraction of the voxel is occupied by blood vessels. Therefore, the ETM incorporates an additional term that includes the contribution of the intravascular CA to the total CA concentration in the voxel.

The major limitation of both models is that they disregard the passive phenomena of CA transport, considering only the arrival of CA to the tissue through blood perfusion. Several authors have previously highlighted this limitation, proving that the assumption of no inter-voxel CA transport can introduce inaccuracies in the estimated parameters of the model [34,15,19,4,28,8,33]. Consequently, they have proposed alternative formulations that include these phenomena, focusing mainly on CA transport by diffusion.

This work is based on our previous publication [31], where we presented the formulation of the diffusion-corrected ETM (D-ETM) model, along with the implementation of a finite element (FE) based optimization algorithm to fit that model to CA concentration curves. In this previous work, we demonstrated the importance of CA diffusion, especially in necrotic regions in tumors, and showed the increased accuracy of the D-ETM model with respect to the ETM when retrieving the model parameters in the presence of CA diffusion. Nevertheless, due to the ill-posed nature of the problem, the algorithm faced convergence problems, getting caught in local minima in some cases.

[*] Corresponding author.
E-mail address: jmgaraz@unizar.es (J.M. García-Aznar).

https://doi.org/10.1016/j.medengphy.2023.104092
Received 6 April 2023; Received in revised form 23 November 2023; Accepted 16 December 2023

Available online 21 December 2023

Thanks to the great advances in Machine Learning (ML), and more precisely in Deep Learning (DL), in the last years several authors have developed different DL-based methods to retrieve PK model's parameters from DCE-MRI data. Such works mainly focus on the use of convolutional neural networks (CNN) [7,41,40] and recurrent neural networks (RNN), such as long short-term memory (LSTM) networks [48]. The main drawback of these architectures is that both are purely data-driven approaches. This means that these networks need to be trained on a sufficiently large dataset that includes different patients from distinct clinical centers where diverse protocols may have been applied. Besides, they do not only need the raw clinical data, but also the "ground-truth" for the parameters of the model being fitted. Although in the case of the STM or the ETM these parameter values can be extracted using a fast algorithm such as non-linear least-squares (NLLS) [37] fitting, if we want to include the diffusive term we need to incorporate other optimization algorithms, such as those proposed in [4,28,8,33,31]. This poses a major challenge, since those algorithms that are fast enough for this task make assumptions that limit their applicability to certain tissues [4,8] and those which can be widely applied to different tissues are too computationally expensive to fit such a large dataset and struggle to converge to the exact solution in some situations [28,33,31]. Thus, it appears that the problem we are addressing requires that the known physical laws that govern the processes of CA transport are included in the network architecture, steering away from purely data-driven approaches.

Recently, Physics Informed Neural Networks (PINNs) [29,16,3] have emerged as a promising alternative for solving PDEs and other inverse problems. PINNs combine the flexibility and scalability of neural networks with the physical constraints imposed by the underlying equations, allowing for efficient and accurate solutions even for highly nonlinear and ill-posed problems. Other authors have used this type of neural networks (NN) to fit tracer-kinetic models to DCE-MRI data [27,12], outperforming current NLLS method. Zapf et al. [46] used PINNs to estimate the diffusion coefficient governing the long term spread of molecules in the human brain from diffusion tensor imaging (DTI) MR data, showing its potential for this task. The results obtained in these works show that PINNs can successfully retrieve PK parameters and solve inverse problems of CA diffusion in biological tissues.

In this paper, we investigate the potential of using PINNs to fit diffusion-corrected pharmacokinetic models to synthetic DCE-MRI data, with the aim of establishing a robust framework for future analysis. To facilitate our exploration, we focus on 1D spatial domains while highlighting the broader implications of this approach for advancing DCE-MRI data analysis.

With this physics-driven NN architecture we aim to overcome the limitations of traditional solvers and achieve more robust parameter estimation for DCE-MRI analysis, showing the potential of PINNs to extract more accurate vascularization data from this type of MR sequences, even when faced with noisy and incomplete data.

2. Methods

This section begins with the introduction of the D-ETM formulation. Then, the fundamentals of PINNs will be explained, finishing with a description of the PINN implementation chosen for the D-ETM.

2.1. D-ETM formulation

The ETM is a compartmental model that, as stated previously, considers two different compartments (intravascular and EES compartments) and the exchange of CA between them. This model assumes the hypothesis of well-mixed compartments presented by Tofts [38], which states that no CA concentration gradients exist within the respective compartments (Eq. (1)):

$$C_t(\boldsymbol{x},t) = v_e(\boldsymbol{x})C_e(\boldsymbol{x},t) + v_p(\boldsymbol{x})C_p(t) \tag{1}$$

where $C_t(\boldsymbol{x},t)$ is the CA concentration in the voxel; $C_e(\boldsymbol{x},t)$ and $C_p(t)$ are the CA concentration in the EES and intravascular compartments, respectively; $v_e(\boldsymbol{x})$ and $v_p(\boldsymbol{x})$ are the volume fraction of each of these compartments with respect to the voxel and $\boldsymbol{x}$ is the spatial coordinates vector. The D-ETM formulation, as defined in [31], adds a diffusive term to the differential formulation of the ETM, obtaining Eq. (2):

$$\frac{\partial C_t(\boldsymbol{x},t)}{\partial t} = \boldsymbol{\nabla}\cdot\left(D_{eff}(\boldsymbol{x})\nabla C_t(\boldsymbol{x},t)\right) + \frac{K^{Trans}}{v_e}(\boldsymbol{x})\left(C_p(t)(v_e(\boldsymbol{x})+v_p(\boldsymbol{x})) - C_t(\boldsymbol{x},t)\right) + v_p(\boldsymbol{x})\frac{\mathrm{d}C_p(t)}{\mathrm{d}t} \tag{2}$$

where $K^{Trans}(\boldsymbol{x})$ is the extravasation rate between the intravascular and the EES compartments.

This formulation is based on the concept of effective diffusivity applied to biological tissues. Essentially, we are assuming that the transport of particles through any biological tissue can be viewed as the transport of particles through a porous medium [24,25,23]. Given the similarity between porosity and v_e, as both measure the volume fraction of "empty" space [36], the effective diffusivity coefficient can be defined as:

$$D_{eff} = \frac{2D}{3-v_e} \tag{3}$$

where D is the diffusivity coefficient of CA in free medium.

2.2. Physics-Informed Neural Networks

In recent years, the field of biomedical engineering has significantly increased in the use of deep learning techniques for a wide range of applications, from medical image analysis [32] to drug discovery [5]. Deep learning algorithms have shown great promise in improving the accuracy and efficiency of tasks such as disease diagnosis, prognostication, and treatment planning [22]. However, many of these approaches rely on large amounts of labeled data, which can be challenging to obtain in biomedical settings [21]. This is where Physics-Informed Neural Networks (PINNs) have emerged as a promising alternative, leveraging the underlying physics of the problem to reduce the reliance on labeled data and improve model generalization [6].

PINNs incorporate prior physical knowledge of the problem into the neural network architecture, making them more efficient and accurate than traditional data-driven DL approaches. PINNs can include partial differential equations (PDEs) to encode the governing physics of the problem, and then use neural networks to approximate the solution to the PDEs.

This combination of physics-based constraints and data-driven learning makes PINNs particularly effective for problems with limited data and complex physical phenomena. In the following sections, we will explain the basic concepts of PINNs and how we used them to solve our specific problem.

2.2.1. Fundamentals

PINNs are based on two main concepts: the Universal Approximation Theorem [13] and Automatic Differentiation (AD) [11].

The Universal Approximation Theorem states that any arbitrary function, no matter its complexity, can be approximated by a NN with only one hidden layer and a finite number of neurons.

Automatic differentiation is a technique for efficiently computing the derivatives of a function specified by a computer program. It works by recursively applying the chain rule of calculus to elementary operations such as addition, multiplication, and elementary functions like exponentials and trigonometric functions. This allows us to compute exact derivatives to machine precision without the need for symbolic manipulation or numerical approximations. In other words, we can obtain the derivatives of a function with the same precision as the function itself.

Therefore, one can train a NN to express solutions of time-dependent linear and non-linear PDEs from a set of inputs. Given a PDE of the form:

$$\frac{\partial u}{\partial t} + \mathcal{N}[u; P] = 0, \quad \boldsymbol{x} \in \Omega \subset \mathbb{R}^D, \quad t \in [0, T] \tag{4}$$

where $\mathcal{N}[u; P]$ is a differential operator with parameters P acting on the hidden solution $u(\boldsymbol{x}, t)$. We can approximate the solution of the PDE $u(\boldsymbol{x}, t)$ with a NN such that $u(\boldsymbol{x}, t) \approx NN(\boldsymbol{x}, t; \theta)$, being θ the NN parameters and $(\boldsymbol{x}, t)$ the input variables.

The process of NN training comprises a set of training data, a loss function $\mathcal{L}$ that measures the fitness performance of the NN with respect to the objective, and an optimizer that adjusts the NN parameters to minimize $\mathcal{L}$. In the case of PINNs, we incorporate the PDE as a constraint in $\mathcal{L}$ to ensure that the solution obtained by the NN satisfies the physical laws described by the PDE on a certain set of collocation points $\{\boldsymbol{x}_i, t_i\}_{i=1}^{N_f} \in \Omega \times [0, T]$. In the case of inverse problems, where $u(\boldsymbol{x}, t)$ is known on some set of training points $\{\boldsymbol{x}_i, t_i, u_i\}_{i=1}^{N_u} \in \Omega \times [0, T]$, this loss function is composed by two main terms:

$$\mathcal{L}(\theta, P) = w_d \mathcal{L}_d + w_r \mathcal{L}_r \tag{5}$$

where:

$$\mathcal{L}_d = \frac{1}{N_u} \sum_{i=1}^{N_u} \left(u(\boldsymbol{x}_i, t_i) - NN(\boldsymbol{x}_i, t_i; \theta) \right)^2 \tag{6}$$

$$\mathcal{L}_r = \frac{1}{N_f} \sum_{i=1}^{N_f} \left(\frac{\partial NN(\boldsymbol{x}, t; \theta)}{\partial t} + \mathcal{N}[NN(\boldsymbol{x}, t; \theta); P] \right)^2 \tag{7}$$

and N_u and N_f are the number of training points and collocation points, respectively. The first term of the loss function (Eq. (6)) is the mean square error (MSE) between the predicted solution and the ground truth solution at the sampled training points. The second term (Eq. (7)) is the MSE of the PDE residual at the collocation points. To compute this term, the differential operators included in the PDE are computed using automatic differentiation. Both terms are scaled by different weighting factors, w_d and w_r that control the relative importance of the data and PDE losses. Additionally, one may add other terms to the loss function related to boundary or initial conditions with their correspondent weighting factors, although they are not mandatory for inverse problems [30].

By minimizing this loss function, the NN learns to approximate the solution of the PDE at the training points while it satisfies the physical laws described by the PDE at the collocation points. For inverse problems this minimization process not only updates the NN weights and biases but also retrieves the unknown PDE parameters P.

2.2.2. Fitting the D-ETM using PINNs

After presenting the general concepts of PINNs, we now shift our focus to the specific implementation used to fit the D-ETM to synthetic DCE-MRI data.

Our implementation is based on previous works [29,14,12] that solved similar problems. Therefore, we opted for a densely connected forward neural network (FNN) architecture consisting of 8 hidden layers and 100 neurons per layer and the hyperbolic tangent (tanh) as the activation function. A normalization layer was included before the hidden layers to normalize the spatial and temporal coordinates to the [-1, 1] range, as it is recognized as a safeguard against vanishing or exploding gradients, as well as a stabilizing factor for the training procedure [29]. The network parameters θ were initialized using Glorot initialization [9] while the D-ETM parameters P were given random values within the physiological range (K^{Trans} between 0.05 and 0.5 min^{-1}; v_e between 0.3 and 1.0 and v_p between 0.01 and 0.3) [31,28]. The original formulation of the loss function presented in Eq. (5) is modified to include two additional terms:

$$\mathcal{L}(\theta, P) = w_d \mathcal{L}_d + w_r \mathcal{L}_r + w_{IC} \mathcal{L}_{IC} + w_C \mathcal{L}_C \tag{8}$$

where $\mathcal{L}_{IC}$ represents the initial conditions, $\mathcal{L}_C$ is a soft constraint for the PDE parameters and w_{IC} and w_C are their respective weighting factors. These additional terms are defined as:

$$\mathcal{L}_{IC} = \frac{1}{N_{IC}} \sum_{i=1}^{N_{IC}} (NN(\boldsymbol{x}_i, 0; \theta))^2 \tag{9}$$

$$\mathcal{L}_C = \sum_{i=1}^{N_p} \left(g(K_i^{Trans}, v_{ei}, v_{p_i}) \right) \tag{10}$$

where N_{IC} is the number of points used to evaluate the initial condition $\left(C_t(\boldsymbol{x}, t) = 0\right)$ and N_p is the number of points where the PDE parameters are evaluated. Given that the three D-ETM parameters (K^{Trans}, v_e and v_p) are spatial distributions, N_p should be equal or greater than the spatial discretization of the data points to achieve sufficient precision. The soft constraint presented in Eq. (10) is defined as:

$$g(K_i^{Trans}, v_{ei}, v_{p_i}) = max(K_i^{Trans}, 0) - K_i^{Trans} + max(v_{ei}, 0) - v_{ei}$$
$$+ max(v_{p_i}, 0) - v_{p_i} + max(v_{ei} + v_{p_i}, 1) - 1 \tag{11}$$

Eq. (11) sets the lower bound for all three PDE parameters to 0, while the upper bound for the sum of $v_e + v_p$ is set to 1. This constraint ensures that the sum of the intravascular space and the EES does not exceed the total voxel volume. These limits ensure that the parameters obtained are physically plausible. The weighting factors were all set to 1 except w_r, which was given a value of 1000. Based on the different tests conducted, this combination of weights leads to the best results. The loss function defined in Eq. (8) is minimized using the Adam optimizer [18] with a constant learning rate of 0.001 during 60,000 epochs. A schematic overview of this PINN implementation is presented in Fig. 1.

The number of training points was obtained from the resolution of the synthetic data, with a total of 360 points in the temporal domain (1 s resolution) and 60 points in the spatial domain (0.1 mm resolution) [31]. That resulted in a total of 21.600 training points. N_{IC} and N_p were set to 100 and 120, respectively. We found empirically that enhancing the spatial discretization of PDE parameters (N_p) with a factor of 2 with respect to the data resolution increased the accuracy. Finally, 10,000 collocation points were distributed over the whole domain using the Latin Hypercube Sampling (LHS) method [35].

The implementation shown so far includes most of the features described in the original work that laid the foundations for PINNs [29]. However, the results obtained with this implementation showed that, although the network was able to fit the data curves accurately, the error in the PDE parameters, particularly v_e, was too high (Appendix A). To retrieve more accurate PDE parameters, we introduced the residual-based adaptive refinement (RAR) method [20]. This method aims to improve the distribution of collocation points during training. After a certain number of epochs (E_{RAR}), the PDE residual is evaluated at a new set of collocation points randomly sampled. Then, these points are ranked by their mean $\mathcal{L}_r$ value. Finally, the top k points are added to the initial list of collocation points. This technique helps the NN focus on those regions where the PDE residual is higher, enhancing the gradients corresponding to the PDE parameters in that regions. Through successive tests, the RAR parameters were set to $E_{RAR} = 500$ and $k = 500$. The deep learning library Tensorflow 2 [1] is used to implement all the methods and optimizations described earlier.

It is important to note that all these hyperparameters were manually calibrated until we obtained satisfactory results. Moreover, it should be emphasized that before applying our methodology to more complex 2D cases, a proper tuning of the hyperparameters is necessary to understand and measure how each hyperparameter affect the results.

3. Examples of application

To assess the effectiveness of PINNs in accurately determining the D-ETM parameters in comparison to previous optimization algorithms

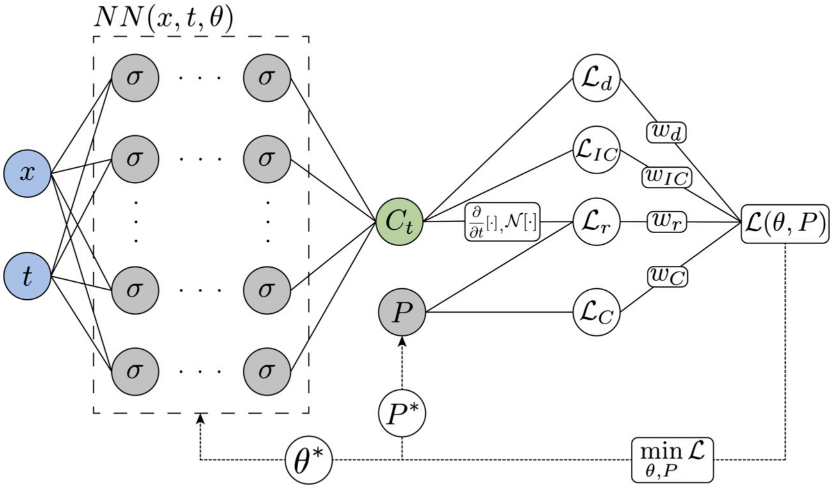

Fig. 1. Schematic representation of the PINN implementation developed to fit the D-ETM. PINNs take advantage of the computational efficiency of automatic differentiation to get the derivatives of C_t needed for the computation of $\mathcal{L}_r$. As in any other inverse problem, the optimizer not only updates the NN parameters (θ^*) but also the PDE parameters (P^*).

[31], two distinct *in silico* cases were devised. Both cases were based on a 1D spatial domain that corresponds to a cross-section of a circular tumor with two different regions: a necrotic core and a highly vascularized rim, similar to the benchmark case used in previous studies [28,8,31]. The objective of this benchmark case was to highlight the effect of diffusion in CA transport, therefore pointing out the limitations of the ETM. The different sets of synthetic CA concentration time courses were generated using the forward implementation of the D-ETM in ANSYS (Ansys Inc., TX, USA), as described in [31]. In all cases the diffusion coefficient for CA in free medium, D, was set to 2.6E-04 mm^2/s [19,10]; while the arterial input function (AIF) was the same as the one used in our previous work [31].

In the first case, the distribution of K^{Trans} and v_p is homogeneous through each of the regions, taking values of 0.3 min^{-1} and 0.1 in the vascularized rim and 0.05 min^{-1} and 0.01 along the necrotic core, respectively. The second case, on the other hand, is based on a heterogeneous distribution of these two vascularization parameters: K^{Trans} and v_p defined in the ranges [0.2, 0.3] min^{-1} and [0.07, 0.13], respectively, along the vascularized rim and K^{Trans} taking values between 0.02 and 0.07 min^{-1} and v_p ranging between 0.0 and 0.05 in the necrotic core. In both cases, v_e was set to 0.5 through the whole domain, to replicate the configuration presented in previous works [28,8,31]. We employed the absolute relative difference (ARD) metric to quantify the error in the fitted parameters for each model.

Additionally, each case was fitted with the two original models: the ETM [38] to and the D-ETM [31]. The latter was fitted using two different optimization methods: the FE-based algorithm presented in [31] (D-ETM FE) and the PINN approach presented in this work (D-ETM PINN). As stated before, in the D-ETM PINN the D-ETM parameters were initialized using random values within their physiological ranges. In the case of the D-ETM FE the output from the ETM fitting was used as initial seed. This was done to reduce the complexity of the minimization process, trying to avoid local minima.

3.1. Homogeneous distribution of parameters

Taking a look at the results obtained for the homogeneous case (Fig. 2 and Table 1), it is clear that this initial seed was not sufficient to prevent the D-ETM FE to converge to a local minimum in this case. In fact, it shows a greater error than the ETM for the K^{Trans} distribution (87% of nodes fitted by the ETM have an ARD error lower than 20%, while the D-ETM has only 73% below that threshold). This outcome was expected, since the limitations of this D-ETM FE when dealing with homogeneous distributions were first discovered in [31].

In comparison, this test proved the robustness of the D-ETM PINN when facing ill-posed problems, getting 99% of the nodes below the ARD threshold with a median ARD of only 0.68%, 3 times lower than the ETM. The Interquartile Range (IQR) also highlights the increased dispersion in the ARD distribution of the D-ETM FE mainly and, to a lesser extent, the ETM.

A key point of these results is the ability of PINNs to accurately adjust v_e values in necrotic zones, where K^{Trans} takes values close to zero. Previous models and algorithms [28,31] failed to retrieve accurately the distribution of v_e in those regions. This was caused by the vanishing effect observed in the gradient of C_t with respect to v_e, since this gradient was dependent on the K^{Trans} value. Therefore, when K^{Trans} tends to zero, the gradient does the same, causing the algorithm to converge to a local minimum. Thanks to the RAR method, PINNs are able to reduce this vanishing effect, overcoming the convergence issues and achieving great accuracy for v_e in necrotic regions.

3.2. Heterogeneous distribution of parameters

The second set of simulations corresponded to the same 1D domain of a circular tumor, but defining heterogeneous distributions of parameters. This case aims to resemble more closely to a real tumor where some degree of heterogeneity is present.

Results obtained are consistent with previous works [28,8,31]: the ETM tends to average the parameters distribution, failing to capture the heterogeneity shown in Fig. 3, while the two implementations of the D-ETM (FE and PINN) accurately depict this heterogeneity (Table 2).

This effect is of particular significance in the case of the v_p, where only 46% of nodes fitted by the ETM show an ARD lower than 20%. This metric raises to 84% and 91% in the case of the PINN and FE implementations of the D-ETM. While the PINN method shows a slightly greater error for v_p, it outperforms the D-ETM FE in the case of K^{Trans}, with 94% of nodes below the ARD threshold versus 78% in the case of the D-ETM FE. The averaging effect shown by the ETM has a clear impact on the K^{Trans} error metrics, having only 52% of nodes with and ARD below 20%.

Regarding v_e, results are similar to the homogeneous case: both the ETM and the D-ETM FE fail to retrieve the v_e distribution. Although around three quarters of nodes are below the ARD threshold, Fig. 3 shows that in some nodes the ARD value is close to 100%. This can be explained by the vanishing effect explained previously, which prevents the D-ETM FE to converge to the solution. Again, the D-ETM PINN overcomes this issue and gets more than 98% of values below the ARD threshold, keeping the maximum ARD value below 25%.

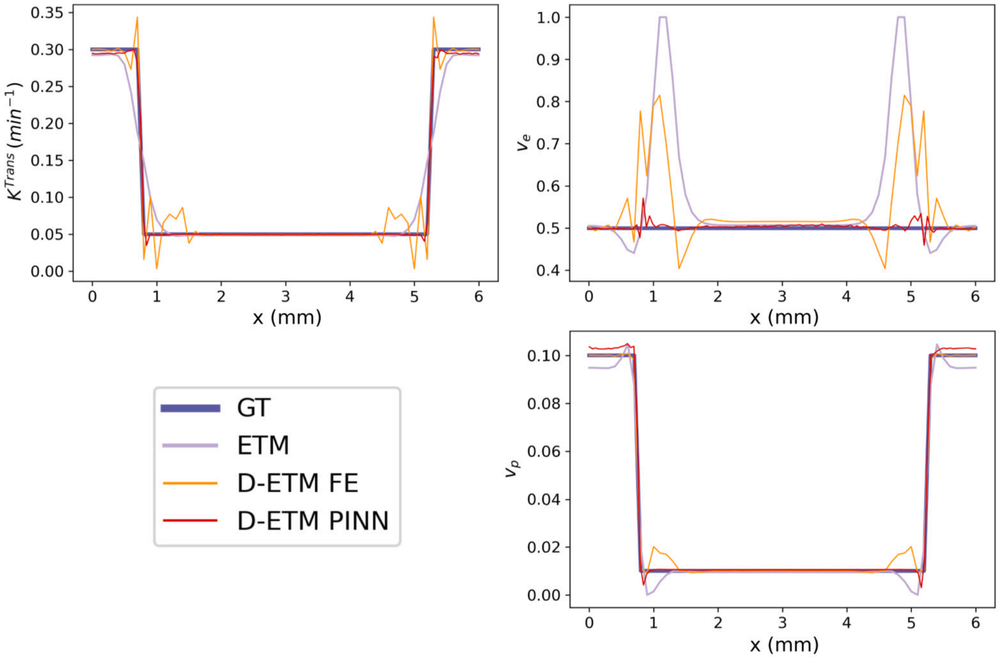

Fig. 2. Homogeneous case. Reference values and results of the D-ETM (with both methods, FE and PINN) and the ETM fitting. Results highlight the limitation of the ETM when faced with significant diffusion gradients, tending to average the K^{Trans} along that region. They also show the improved accuracy of the D-ETM PINN with respect to the D-ETM FE. While the latter converges to a local minimum, as in [31], the former retrieves accurately the distribution of parameters.

Table 1
Error metrics comparison between the D-ETM (FE and PINN methods) and the ETM for the homogeneous case. The metrics computed are the median and Interquartile Range (IQR) of the ARD distribution and the fraction of nodes whose ARD is below the defined threshold of 20%.

	D-ETM PINN			D-ETM FE			ETM		
	Median (%)	IQR (%)	Frac nodes ARD < 20%	Median (%)	IQR (%)	Frac nodes ARD < 20%	Median (%)	IQR (%)	Frac nodes ARD < 20%
K^{Trans}	0.68	1.46	99.18	2.43	22.11	73.77	2.20	3.61	86.88
v_e	0.50	0.45	100	3.26	9.21	83.61	1.62	10.62	83.61
v_p	4.70	1.53	97.54	1.18	5.11	86.88	5.90	0.8	86.88

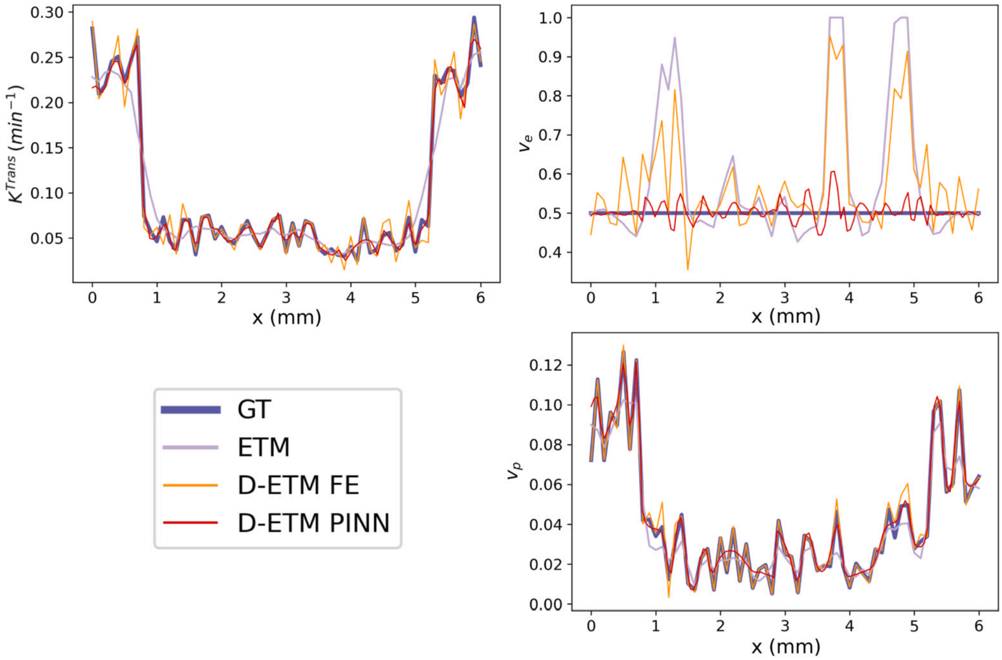

Fig. 3. Heterogeneous case. Reference values and results of the D-ETM (with both methods, FE and PINN) and the ETM fitting.

Table 2
Error metrics comparison between the D-ETM (FE and PINN methods) and the ETM for the heterogeneous case. The metrics computed are the median (m) and IQR of the ARD distribution and the fraction of nodes whose ARD is below the defined threshold of 20%. These metrics were computed from a set of 10 simulations with different heterogeneous distributions of parameters.

	D-ETM PINN			D-ETM FE			ETM		
	m (%)	IQR (%)	Frac nodes ARD < 20%	m (%)	IQR (%)	Frac nodes ARD < 20%	m (%)	IQR (%)	Frac nodes ARD < 20%
K^{Trans}	4.04	7.39	95.16	8.81	18.31	78.33	18.85	26.92	52.46
v_e	2.01	4.34	98.93	10.19	18.33	72.13	7.94	12.29	77.05
v_p	7.43	12.24	82.95	1.68	4.76	91.80	22.60	16.17	45.90

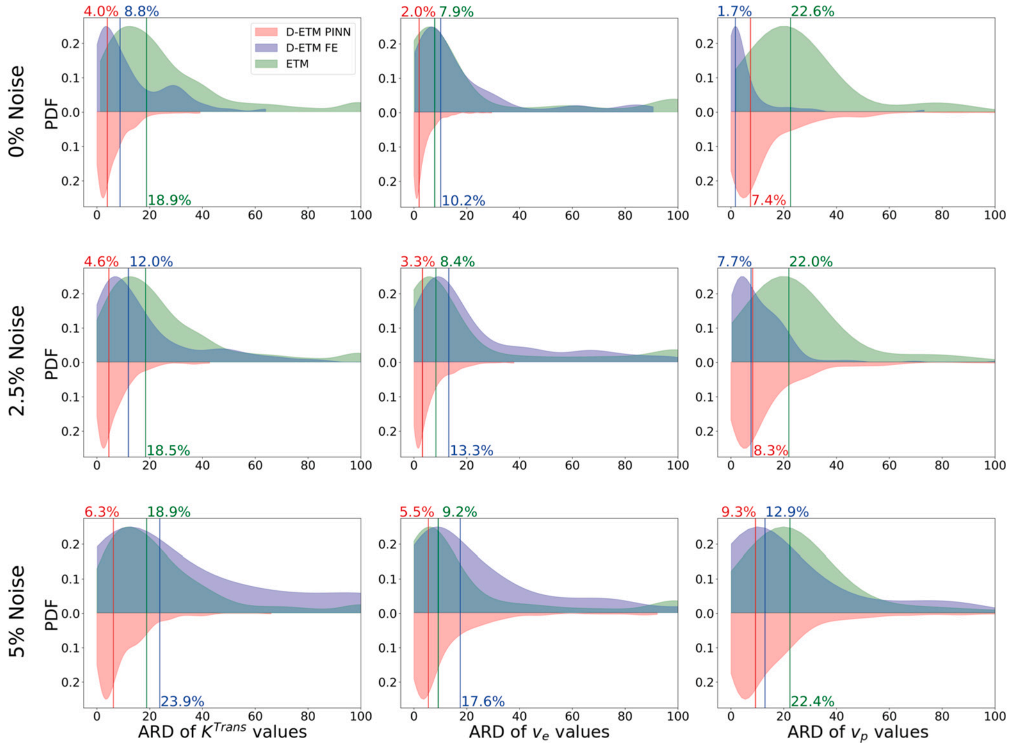

Fig. 4. Comparison of the ARD probability density function (PDF) of each of the parameters for each of the models and three different levels of noise: 0%, 2.5% and 5%. Vertical lines correspond to the median of the ARD distributions. The D-ETM PINN is much more robust to noise compared to the D-ETM FE, while the ETM does not seem affected by noise.

It is worth noting that despite this good error metrics, the D-ETM PINN shows some kind of averaging patterns in some regions of the spatial domain, especially for v_p. This is probably due to the different effect these parameters have on the cost function depending on the algorithm. While in the case of the D-ETM FE the C_t curves depended on the parameters value through a forward FE simulation (see [31] for further details), therefore increasing the impact of PDE parameters (mainly v_p) on the cost function; in the case of the D-ETM PINN the PDE parameters only impact part of the loss function ($\mathcal{L}_r$ and $\mathcal{L}_C$). Therefore, there can be small errors on the PDE parameters distribution while the total loss value $\mathcal{L}$ is minimized, since the data loss ($\mathcal{L}_d$) is being reduced by updating the NN parameters (θ).

4. Testing the robustness of the PINN approach against noisy and incomplete data

After demonstrating the increased accuracy of the D-ETM PINN with respect to the D-ETM FE, we test its robustness when faced with noise and incomplete temporal data.

4.1. Influence of noise

Initially, a set of 1D heterogeneous distributions of parameters similar to those presented in Fig. 3 were generated. Next, experimental noise was added to the generated C_t data curves using a Gaussian distribution with a standard deviation (SD) equal to a fraction (2.5%, and 5%) of the highest concentration value reached in the whole domain, similar to previous works [28,31].

The ARD distributions for each of the parameters and each model are shown in Fig. 4. These results show that the D-ETM PINN is much more robust to noise than the D-ETM FE, even in the case of v_p. When faced with medium levels of noise, both methods show similar ARD for this variable (Table 3). However, when the noise level reaches 5%, the D-ETM PINN is more accurate than the D-ETM FE (Table 4).

The results of the other two variables follow this same trend: the influence of noise is much lower in the case of the D-ETM PINN compared to the D-ETM FE. The ETM, however, seems to be unaffected by noise, reaching a similar accuracy to the D-ETM FE for high noise levels (5%).

Table 3
Error metrics comparison between the D-ETM (FE and PINN methods) and the ETM for the heterogeneous case corresponding to the 2.5% noise level. The metrics computed are the median (m) and IQR of the ARD distribution and the fraction of nodes whose ARD is below the defined threshold of 20%. These metrics were computed from a set of 10 simulations with different heterogeneous distributions of parameters.

	D-ETM PINN			D-ETM FE			ETM		
	m (%)	IQR (%)	Frac nodes ARD < 20%	m (%)	IQR (%)	Frac nodes ARD < 20%	m (%)	IQR (%)	Frac nodes ARD < 20%
K^{Trans}	4.62	7.42	93.85	12.04	21.43	64.20	18.46	25.88	54.09
v_e	3.31	6.59	96.39	13.31	22.69	69.67	8.36	17.61	74.92
v_p	8.30	12.33	82.13	7.66	12.47	87.71	21.98	17.80	42.63

Table 4
Error metrics comparison between the D-ETM (FE and PINN methods) and the ETM for the heterogeneous case corresponding to the 5% noise level. The metrics computed are the median and IQR of the ARD distribution and the fraction of nodes whose ARD is below the defined threshold of 20%. These metrics were computed from a set of 10 simulations with different heterogeneous distributions of parameters.

	D-ETM PINN			D-ETM FE			ETM		
	m (%)	IQR (%)	Frac nodes ARD < 20%	m (%)	IQR (%)	Frac nodes ARD < 20%	m (%)	IQR (%)	Frac nodes ARD < 20%
K^{Trans}	6.29	10.05	89.67	23.86	56.44	43.44	18.90	25.45	53.61
v_e	5.46	11.29	85.32	17.57	22.92	67.21	9.24	21.34	73.11
v_p	9.28	14.43	78.61	12.92	27.37	61.48	22.36	20.33	42.13

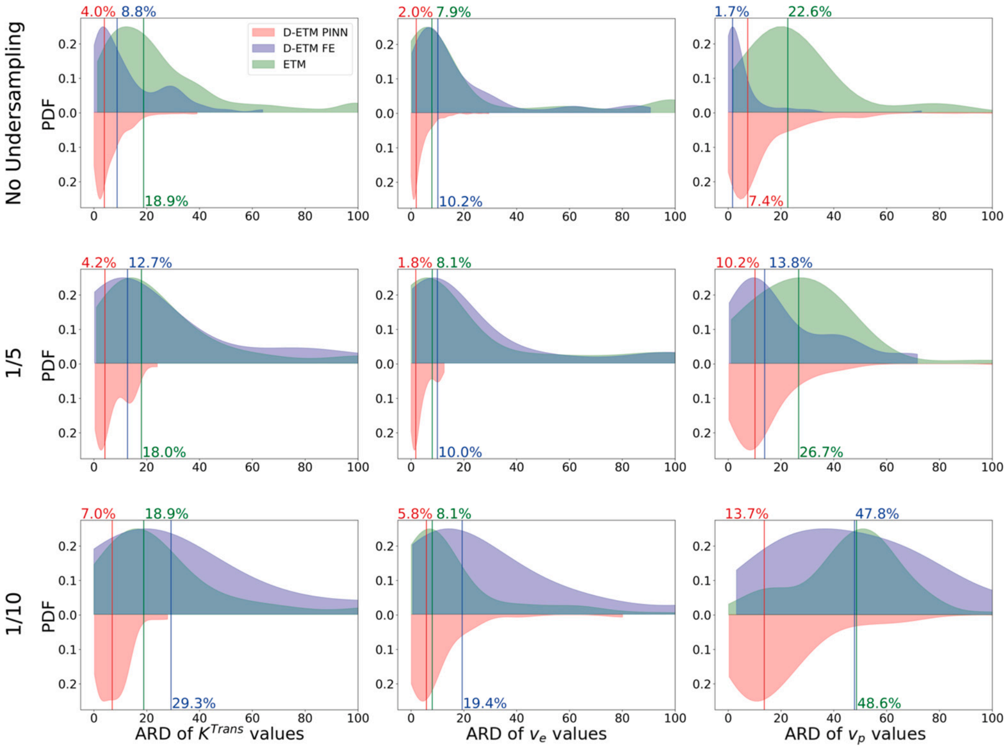

Fig. 5. Comparison of the ARD probability density function (PDF) of each of the parameters for each of the models over three different time resolutions: 1 s (No undersampling), 5 s (1/5) and 10 s (1/10). Vertical lines correspond to the median of the ARD distributions. The D-ETM PINN outperforms the other two, showing great robustness even in the worst scenario.

4.2. Temporal undersampling

In this final subsection, we investigated the impact of incomplete data on the accuracy of the ETM, the D-ETM FE, and D-ETM PINN. To achieve this, we conducted a temporal undersampling analysis on both the concentration data curves and the arterial input function (AIF). Specifically, we considered two scenarios where the temporal resolution was reduced to 5 seconds and 10 seconds, respectively. The aim of this analysis was to simulate realistic situations in which the temporal resolution may deviate from the ideal resolution of 1 second used in our previous experiments.

Our results, presented in Fig. 5, revealed that the ETM was almost unaffected by the undersampling, except for the v_p variable, which was more sensitive to it. Conversely, the D-ETM FE performed well at a temporal resolution of 5 s, but its performance degraded when the temporal resolution was reduced to 10 s, obtaining higher errors than the ETM.

Table 5
Error metrics comparison between the D-ETM (FE and PINN methods) and the ETM for the case with a temporal resolution of 5 s. The metrics computed are the median (m) and IQR of the ARD distribution and the fraction of nodes whose ARD is below the defined threshold of 20%.

	D-ETM PINN			D-ETM FE			ETM		
	m (%)	IQR (%)	Frac nodes ARD $<20\%$	m (%)	IQR (%)	Frac nodes ARD $<20\%$	m (%)	IQR (%)	Frac nodes ARD $<20\%$
K^{Trans}	4.23	10.09	96.72	12.70	26.74	60.66	18.03	21.85	52.46
v_e	1.81	5.07	100	9.97	20.92	73.77	8.05	13.01	77.05
v_p	10.20	12.01	86.60	13.81	26.52	70.49	26.68	23.86	31.15

Table 6
Error metrics comparison between the D-ETM (FE and PINN methods) and the ETM for the case with a temporal resolution of 10 s. The metrics computed are the median (m) and IQR of the ARD distribution and the fraction of nodes whose ARD is below the defined threshold of 20%.

	D-ETM PINN			D-ETM FE			ETM		
	m (%)	IQR (%)	Frac nodes ARD $<20\%$	m (%)	IQR (%)	Frac nodes ARD $<20\%$	m (%)	IQR (%)	Frac nodes ARD $<20\%$
K^{Trans}	6.97	7.93	93.44	29.26	40.82	40.98	18.92	19.11	52.46
v_e	5.76	8.36	89.34	19.38	34.31	50.82	8.07	13.27	80.32
v_p	13.72	18.28	67.20	47.84	49.16	18.03	48.59	20.54	14.75

Interestingly, the D-ETM PINN demonstrated superior performance, even under the worst-case scenario of a 10 s temporal resolution. As shown in Table 5 and Table 6, the D-ETM PINN outperformed both the D-ETM FE and the ETM. With a 10 s resolution, 93% and 89% of nodes show an ARD lower than the 20% threshold for K^{Trans} and v_e, respectively. This was almost twice the proportion observed for the ETM and the D-ETM FE. Even in the case of the sensitive v_p variable, the D-ETM PINN maintained a relatively high proportion (67%) of values below the ARD threshold, in comparison to the D-ETM FE (18%) and the ETM (15%).

In summary, our findings suggest that the D-ETM PINN is more robust to incomplete data and performs better than both the ETM and D-ETM FE under these conditions. The ETM performs reasonably well except for the v_p variable, while the D-ETM FE shows good performance at a 5 s temporal resolution but struggles with further undersampling.

5. Discussion

DCE-MRI is a powerful imaging technique widely used in clinical practice, particularly in oncology, to assess the vascular properties of tissues. The ability to obtain functional information about tumors using DCE-MRI is useful for diagnosing, staging, and monitoring tumors' response to antiangiogenic therapies [2,47,26]. However, accurately retrieving physiological parameters from DCE-MRI is a challenging task due to the complexity of the underlying pharmacokinetic models. Traditional models, such as the standard and extended Tofts models, are widely used to estimate these parameters but are known to produce inaccurate results in regions where there is significant passive delivery of CA [34,15,19,4,28,8,33,31]. Other authors have proposed different approaches that include the diffusion of CA and have developed several methods to fit their models to DCE-MRI data. Nevertheless, these approaches either have a limited applicability due to the hypothesis considered or due to their convergence issues computational cost. Therefore, there is a critical need for new methods that can retrieve more accurate parameters from DCE-MRI data.

The main objective of this study was to explore the use of PINNs as an alternative to other traditional algorithms to fit one of the models that include the diffusion term: the D-ETM. To do so, we tested the performance of this PINN approach versus the FE-based optimization algorithm presented in [31]. Both methods were compared to the ETM to highlight the importance of diffusion in CA delivery.

We tested these approaches on a 1D domain resembling a slice of a circular tumor with a highly vascularized rim and a necrotic core. This highlights CA diffusion effects, creating large CA concentration gradients between regions.

Our results indicate that PINNs are a promising tool to solve the ill-posed inverse problems associated with the fitting of the D-ETM to DCE-MRI data. The PINN-based approach kept almost all nodes in the domain below the acceptable error threshold. Previous algorithms, such as the FE-based, failed to retrieve the v_e distribution in necrotic regions, due to a low influence of this parameter on the global solution. However, the use of PINNs along with the RAR method overcome this limitation, outperforming traditional algorithms. Even in the homogeneous case, where the FE algorithm converged to a local minimum, the PINN approach depicted accurately the distribution of all the parameters.

To further demonstrate the robustness of PINNs, we tested its performance in the presence of noisy and incomplete data. The results obtained show that the PINN was affected to a much lower extent than the FE algorithm, retrieving very accurate distributions of D-ETM parameters. Taking a look at the error distribution for each of the three approaches tested we could conclude that PINNs combine (and even increase) the precision of the FE algorithm with the robustness of the ETM against noisy and incomplete data.

Despite these promising results, there is still room for improvement. First, in this study we did not perform a comprehensive hyperparameter tuning, which may have resulted in suboptimal performance. Future studies should focus on optimizing the PINN hyperparameters to further improve its performance [43]. Second, there are other additional features of PINNs described in the literature that were not included in our study. These may include the use of gradient-enhanced PINNs [20], the inclusion of annealing algorithms to update each loss weight (w_i) or new NN architectures optimized for PINNs [42]. Incorporating these features in future studies may enhance the accuracy and robustness of the PINN approach while reducing the training time.

The main limitation of our methodology is the lengthy training time required for the PINN. Our experiments were conducted on a PC equipped with a NVIDIA RTX 3070 GPU, 32 GB RAM, and an Intel i7-11700K CPU, with an average training time of around 30 minutes. This is even slower than the current FE algorithm, which took an average of 20 minutes on the same PC. Meanwhile, the NLLS algorithm used for the ETM required only a few seconds to fit all nodes in the domain, so it cannot be directly compared to either of our methods. To address this limitation, proper calibration of the PINN may help to reduce the training time. Additionally, transfer-learning techniques [45,44] can be applied to further lower the computational cost.

Our study demonstrates the capability of PINNs to overcome convergence issues when fitting the D-ETM to DCE-MRI data, outperforming previous algorithms. Currently, this 1D implementation has a limited applicability on *in vivo* data as living tissues rarely have axisymmetric properties. Therefore, this work lays the foundation for further research that improves our implementation and optimizes it for its application to 2D cases, the first and necessary step before applying this methodology to *in vivo* cases.

Ethical approval

Not required.

Declaration of competing interest

The authors report no conflicts of interest.

Acknowledgements

This work has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 826494 (PRIMAGE). The work was also supported by the Ministry of Science, Education and Universities, Spain (FPU18/04541). This work is part of the Grants PID2020-113819RB-I00, PLEC 2021-007709 and PID2021-122409OB-C21 funded by MCIN/AEI/10.13039/501100011033/, by "ERDF A way of making Europe" and by the "European Union NextGenerationEU/PRTR". This work also received funding from the European Research Council (ICoMICS Adv grant agreement: 101018587).

Appendix A. Influence of RAR method on the results

In order to highlight the influence of the RAR method on the precision of the retrieved parameters, two PINN implementations, both with and without using the RAR method, were trained and fitted to some data curves. Specifically, the data corresponds to the homogeneous case presented in section 3.1. Fig. A.6 compares the reference curve to the fitted curves obtained with and without the RAR method. These results show that both implementations are able to fit very accurately the reference curves, with minimal differences between each implementation.

However, the distribution of fitted parameters presented in Fig. A.7 highlights the limitations of the *vanilla* PINN (i.e., the standard implementation proposed in the original paper by *Raissi et al.* [29]). This original implementation fails to accurately retrieve the vascularization parameters, specifically the v_e parameter. By introducing the RAR method into the PINN implementation, we help the NN focus on those regions where the loss term related to the PDE reaches higher values, increasing significantly the precision of the fitted v_e map.

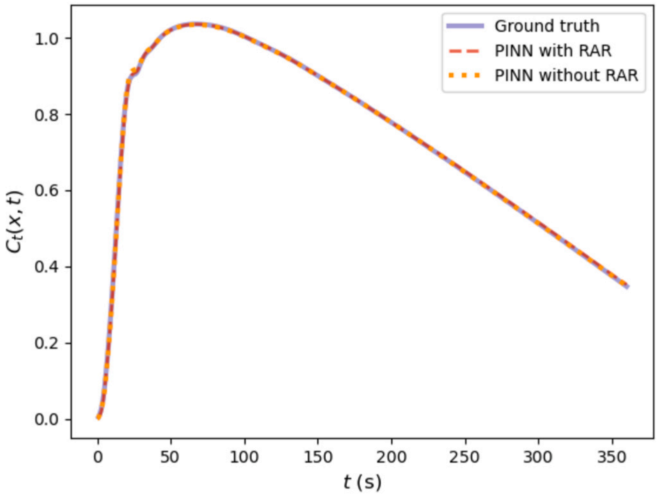

Fig. A.6. Comparison of the reference curves (ground truth) and the fitted curves obtained with the two PINN implementations, with and without the RAR method. The concentration values are expressed in mM. These curves correspond to one of the nodes of the homogeneous case presented in section 3.1. These results show the accuracy of both methods in fitting the curves.

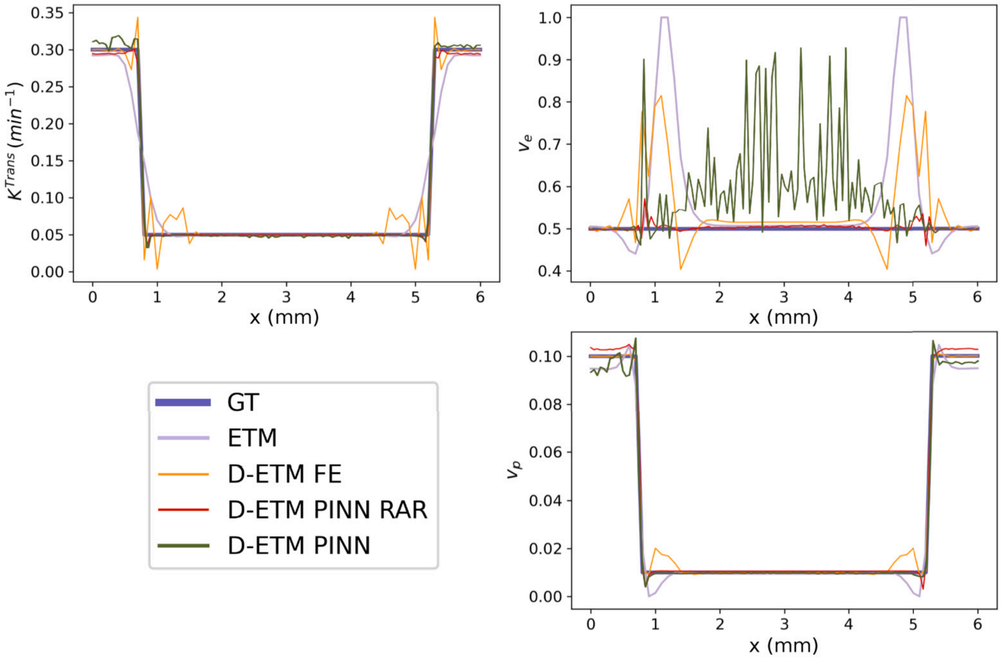

Fig. A.7. Vascularization parameters maps corresponding to the homogeneous case. Reference values and results of the D-ETM and the ETM fitting. The D-ETM was fitted using both the FE algorithm and the PINN approach, with and without the RAR method. Results show the lack of accuracy of the *vanilla* PINN in comparison with the implementation that included the RAR method.

References

[1] Abadi M, Barham P, Chen J, et al. Tensorflow: a system for large-scale machine learning. In: Proceedings of the 12th USENIX conference on operating systems design and implementation. USA: USENIX Association; 2016. p. 265–83.

[2] Barrett T, Brechbiel M, Bernardo M, Choyke PL. MRI of tumor angiogenesis. J Magn Reson Imaging 2007;26:235–49. https://doi.org/10.1002/jmri.20991. http://doi.wiley.com/10.1002/jmri.20991.

[3] Blechschmidt J, Ernst OG. Three ways to solve partial differential equations with neural networks — a review. GAMM-Mitt 2021;44:e202100006. https://doi.org/10.1002/GAMM.202100006. arXiv:2102.11802.

[4] Cantrell CG, Vakil P, Jeong Y, Ansari SA, Carroll TJ. Diffusion-compensated Tofts model suggests contrast leakage through aneurysm wall. Magn Reson Med 2017;78:2388–98. https://doi.org/10.1002/mrm.26607.

[5] Chen H, Engkvist O, Wang Y, Olivecrona M, Blaschke T. The rise of deep learning in drug discovery. Drug Discov Today 2018;23:1241–50. https://doi.org/10.1016/J.DRUDIS.2018.01.039.

[6] Cuomo S, Di Cola VS, Giampaolo F, Rozza G, Raissi M, Piccialli F. Scientific machine learning through physics–informed neural networks: where we are and what's next. J Sci Comput 2022;2022:1–62. https://doi.org/10.1007/S10915-022-01939-Z. arXiv:2201.05624.

[7] Fang K, Wang Z, Li Z, et al. Convolutional neural network for accelerating the computation of the extended Tofts model in dynamic contrast-enhanced magnetic resonance imaging. J Magn Reson Imaging 2021;53:1898–910. https://doi.org/10.1002/JMRI.27495.

[8] Fluckiger JU, Loveless ME, Barnes SL, Lepage M, Yankeelov TE. A diffusion-compensated model for the analysis of DCE-MRI data: theory, simulations, and experimental results. Phys Med Biol 2013;58:1983–98. https://doi.org/10.1088/0031-9155/58/6/1983.

[9] Glorot X, Bengio Y. Understanding the difficulty of training deep feedforward neural networks.

[10] Gordon MJ, Chu KC, Margaritis A, Martin AJ, Ross Ethier C, Rutt BK. Measurement of Gd-DTPA diffusion through PVA hydrogel using a novel magnetic resonance imaging method. Biotechnol Bioeng 1999;65:459–67. https://doi.org/10.1002/(SICI)1097-0290(19991120)65:4.

[11] Güne¸ A, Baydin G, Pearlmutter BA, Siskind JM. Automatic differentiation in machine learning: a survey. arXiv:1502.05767v4, 2018.

[12] van Herten RL, Chiribiri A, Breeuwer M, Veta M, Scannell CM. Physics-informed neural networks for myocardial perfusion MRI quantification. Med Image Anal 2022;78:102399. https://doi.org/10.1016/J.MEDIA.2022.102399. arXiv:2011.12844.

[13] Hornik K, Stinchcombe M, White H. Multilayer feedforward networks are universal approximators. Neural Netw 1989;2:359–66. https://doi.org/10.1016/0893-6080(89)90020-8.

[14] Iversen KF. Informed Neural Networks for Inverse Advection-Diffusion Problems.

[15] Jia G, O'Dell C, Heverhagen JT, et al. Colorectal liver metastases: contrast agent diffusion coefficient for quantification of contrast enhancement heterogeneity at MR imaging. Radiology 2008;248:901–9. https://doi.org/10.1148/radiol.2491071936.

[16] Karniadakis GE, Kevrekidis IG, Lu L, et al. Physics-informed machine learning. Nat Rev Phys 2021;3:422–40. https://doi.org/10.1038/s42254-021-00314-5.

[17] Khalifa F, Soliman A, El-Baz A, et al. Models and methods for analyzing DCE-MRI: a review. Med Phys 2014;41:124301. https://doi.org/10.1118/1.4898202.

[18] Kingma DP, Lei Ba J. Adam: a method for stochastic optimization. arXiv:1412.6980v9.

[19] Koh TS, Hartono S, Thng CH, et al. In vivo measurement of gadolinium diffusivity by dynamic contrast-enhanced MRI: a preclinical study of human xenografts. Magn Reson Med 2013;69:269–76. https://doi.org/10.1002/mrm.24246.

[20] Lu L, Meng X, Mao Z, Karniadakis GE. DeepXDE: a deep learning library for solving differential equations 2021;63:208–28. https://doi.org/10.1137/19M1274067. arXiv:1907.04502.

[21] Luca AR, Ursuleanu TF, Gheorghe L, Grigorovici R, Iancu S, Hlusneac M, et al. Impact of quality, type and volume of data used by deep learning models in the analysis of medical images. Inform Med Unlocked 2022;29:100911. https://doi.org/10.1016/J.IMU.2022.100911.

[22] Lundervold AS, Lundervold A. An overview of deep learning in medical imaging focusing on MRI. Z Med Phys 2019;29:102–27. https://doi.org/10.1016/J.ZEMEDI.2018.11.002. arXiv:1811.10052.

[23] Nicholson C. Diffusion and related transport mechanisms in brain tissue. Rep Prog Phys 2001;64:815–84. https://doi.org/10.1088/0034-4885/64/7/202.

[24] Nicholson C, Phillips JM. Ion diffusion modified by tortuosity and volume fraction in the extracellular microenvironment of the rat cerebellum. J Physiol 1981;321:225–57. https://doi.org/10.1113/JPHYSIOL.1981.SP013981.

[25] Nicholson C, Syková E. Extracellular space structure revealed by diffusion analysis. Trends Neurosci 1998;21:207–15. https://doi.org/10.1016/S0166-2236(98)01261-2.

[26] O'Connor JPB, Aboagye EO, Adams JE, et al. Imaging biomarker roadmap for cancer studies. Nature Publishing Group; 2017.

[27] Ottens T, Barbieri S, Orton MR, et al. Deep learning DCE-MRI parameter estimation: application in pancreatic cancer. Med Image Anal 2022;80:102512. https://doi.org/10.1016/J.MEDIA.2022.102512.

[28] Pellerin M, Yankeelov TE, Lepage M. Incorporating contrast agent diffusion into the analysis of DCE-MRI data. Magn Reson Med 2007;58:1124–34. https://doi.org/10.1002/mrm.21400.

[29] Raissi M, Perdikaris P, Karniadakis GE. Physics-informed neural networks: a deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. J Comput Phys 2019;378:686–707. https://doi.org/10.1016/J.JCP.2018.10.045.

[30] Raissi M, Yazdani A, Karniadakis GE. Hidden fluid mechanics: a Navier-Stokes informed deep learning framework for assimilating flow visualization data. arXiv:1808.04327v1, 2018.

[31] Sainz-DeMena D, Ye W, Pérez MÁ, García-Aznar JM. A finite element based optimization algorithm to include diffusion into the analysis of DCE-MRI. Eng Comput 2022;38:3849–65. https://doi.org/10.1007/S00366-022-01667-W.

[32] Shen D, Wu G, Suk HI. Deep learning in medical image analysis. Annu Rev Biomed Eng 2017;19:221. https://doi.org/10.1146/ANNUREV-BIOENG-071516-044442. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5479722/.

[33] Sinno N, Taylor E, Milosevic M, Jaffray DA, Coolens C. https://doi.org/10.1088/1361-6560/AC2205.

[34] Sourbron S. A tracer-kinetic field theory for medical imaging. IEEE Trans Med Imaging 2014;33:935–46. https://doi.org/10.1109/TMI.2014.2300450.

[35] Stein M. Large sample properties of simulations using Latin hypercube sampling. Technometrics 1987;29:143. https://doi.org/10.2307/1269769.

[36] Tao L, Nicholson C. Maximum geometrical hindrance to diffusion in brain extracellular space surrounding uniformly spaced convex cells. J Theor Biol 2004;229:59–68. https://doi.org/10.1016/J.JTBI.2004.03.003.

[37] Teunissen P. Nonlinear least-squares. Manuscr Geod 1990;15:137–50.

[38] Tofts PS, Brix G, Buckley DL, et al. Estimating kinetic parameters from dynamic contrast-enhanced T 1-weighted MRI of a diffusable tracer: standardized quantities and symbols. Technical Report. 1999.

[39] Tofts PS, Kermode AG. Measurement of the blood-brain barrier permeability and leakage space using dynamic MR imaging. 1. Fundamental concepts. Magn Reson Med 1991;17:357–67. https://doi.org/10.1002/mrm.1910170208.

[40] Ulas C, Das D, Thrippleton MJ, et al. Convolutional neural networks for direct inference of pharmacokinetic parameters: application to stroke dynamic contrast-enhanced MRI. Front Neurol 2019;9:1147. https://doi.org/10.3389/FNEUR.2018.01147.

[41] Ulas C, Tetteh G, Thrippleton MJ, et al. Direct estimation of pharmacokinetic parameters from DCE-MRI using deep CNN with forward physical model loss. In: Medical image computing and computer assisted intervention – MICCAI 2018. Cham: Springer International Publishing; 2018. p. 39–47.

[42] Wang S, Teng Y, Perdikaris P. Understanding and mitigating gradient flow pathologies in physics-informed neural networks. SIAM J Sci Comput 2021;43:A3055–81. https://doi.org/10.1137/20M1318043.

[43] Wang Y, Han X, Chang CY, et al. Auto-PINN: understanding and optimizing physics-informed neural architecture. arXiv:2205.13748, 2022.

[44] Weiss K, Khoshgoftaar TM, Wang DD. A survey of transfer learning. J Big Data 2016;3:1–40. https://doi.org/10.1186/S40537-016-0043-6/TABLES/6.

[45] Xu C, Cao BT, Yuan Y, Meschke G. Transfer learning based physics-informed neural networks for solving inverse problems in engineering structures under different loading scenarios. Comput Methods Appl Mech Eng 2023;405:115852. https://doi.org/10.1016/J.CMA.2022.115852. arXiv:2205.07731.

[46] Zapf B, Haubner J, Kuchta M, Ringstad G, Eide PK, Mardal KA. Supp. material: investigating molecular transport in the human brain from MRI with physics-informed neural networks. Sci Rep 2022;12:1–11. https://doi.org/10.1038/s41598-022-19157-w. arXiv:2205.02592.

[47] Zormpas-Petridis K, Jerome NP, Blackledge MD, et al. MRI imaging of the hemodynamic vasculature of neuroblastoma predicts response to antiangiogenic treatment. Cancer Res 2019;79:2978–91.

[48] Zou J, Balter JM, Cao Y. Estimation of pharmacokinetic parameters from DCE-MRI by extracting long and short time-dependent features using an LSTM network. Med Phys 2020;47:3447. https://doi.org/10.1002/MP.14222.