2023 62nd IEEE Conference on Decision and Control (CDC)
December 13-15, 2023. Marina Bay Sands, Singapore

Adaptive Estimation of the Pennes' Bio-Heat Equation - II: a NN-based implementation for real-time applications

G. Cappellini*, G. Trappolini*, E. Staffetti†, A. Cristofaro*, M. Vendittelli*

***Abstract*— This is the companion paper of a two-part work on the observation of the heat transfer phenomenon in biological tissues. In particular, we are interested in real-time estimation of the temperature in the interior of a spatial domain of interest using measurements at its boundary. The prevailing model for heat transfer in biological tissues, pioneered by Pennes [1], relies on a parabolic reaction-diffusion partial differential equation. However, neither the observation problem has been fully explored nor have the available solutions proved suitable for real-time applications. In the companion paper [2], we propose the design of an observer whose formal properties, however, cannot be easily reflected in its practical performance, due to computational issues arising with the use of common numerical solvers. The difficulties are mostly related to the integration of a system of coupled PDEs/ODE, required by the algorithm. In this paper, we propose an alternative implementation of the observer that makes use of deep neural networks for predicting the PDEs state, thus avoiding the online integration. Preliminary results show that this approach is very effective in solving the considered problem and is amenable to extension to other classes of PDEs and to higher dimensions.**

I. Introduction

Biological tissue heating has been widely studied in the literature both for determining health hazards in case of exposure to electromagnetic fields [3], [4], [5], [6], and for devising therapeutic treatments [7], [8], [9]. Hyperthermia, in particular, is used in oncology to increase the effectiveness of radiotherapy and chemotherapy [10], [11].

Literature studies have shown that the clinical effectiveness of hyperthermia treatments is critically related to the ability to reach and maintain the desired temperatures at the target, with the right timing, for the desired duration, in a reproducible manner for all the subsequent treatments [12], [13]. These quality requirements are, however, hard to meet, particularly in tumours that are typically treated with superficial hyperthermia, e.g., breast, head and neck, and skin/extremities cancers. Recent advances [14] are paving the way to the automatic delivery of hyperthermia treatments. However, the absence of feedback control of the temperature rise at the target prevents an appropriate level of accuracy in the delivery of the needed thermal dose. To this end, it is necessary to get information on the temperature of the whole target volume. Currently, temperature measurements are invasive and only available at specific points. This work is a first step toward a non-invasive temperature estimation achieved through the design of an observer using only measurements at one boundary.

The observation problem of Partial Differential Equations (PDEs) is difficult and still open in general. For the class of parabolic reaction-diffusion equations with an unknown coefficient of the reaction term, which encompasses the bio-heat equation studied here, some estimation approaches have been proposed based on backstepping design [15], structured uncertainties [16], interval observers [17] and sparsity [18].

Following a novel pathway in the context of infinite-dimensional systems, the multiple-model (MM) observer proposed in the companion paper [2] estimates the prediction model that minimizes the measurement error. The convergence of the estimation error to a bounded set is formally proven and the bound is shown to depend on the granularity of the approximation, hence on the number of prediction models. In principle, the error can be made arbitrarily small by increasing the number of prediction models. This theoretical property encounters some practical difficulties: by increasing the number of prediction models, the implementation relying on traditional solvers, available, for example, in Matlab, incurs numerical instability, and the convergence rate cannot be pushed too far [2]. On the other hand, if the number of prediction models is kept small, the error bound is too high.

In this paper, we propose an alternative implementation of the observer introduced in [2] that makes use of deep learning for the prediction step. Deep learning is a machine learning method based on multi-layer artificial neural networks, usually referred to as deep neural networks. The solution of PDEs with deep learning is currently attracting interdisciplinary attention. Since a great amount of data, not always available, is usually required for training deep neural networks, the key idea in solving PDEs with deep learning is to use additional information obtained by enforcing physical laws at random points in the continuous space-time domain to train such networks. This approach is known as physics-informed learning. In [19], some of the prevailing trends in incorporating physics into machine learning are reviewed, and the current capabilities and limitations of physics-informed learning are described.

Physics-informed learning is implemented in [20] through the so-called Physics-Informed Neural Networks (PINNs) to solve forward and inverse problems involving nonlinear PDEs. We have used the Python library called DeepXDE for solving PDEs through PINNs [21].

The main contribution of the paper is the proposition of

* Dipartimento di Ingegneria Informatica, Automatica e Gestionale, Sapienza University of Rome, Italy
† Universidad Rey Juan Carlos, Madrid, Spain
Corresponding author: G. Cappellini, guglielmo.cappellini@uniroma1.it

This work was partially supported by PNRR MUR projects PE0000013-FAIR and IR0000013-SoBigData.it, Sapienza project "Modelling, control and simulation of systems governed by partial differential equations" (prot. RM12117A803B0D13), and project CADUCEO (No. F/180025/01-05/X43, funded by the Italian Ministry of Enterprises and Made in Italy)

a computational model of the observer in [2] making use of PINNs for the prediction step to overcome the intrinsic limitations of standard solvers. Beyond its practical utility, the approach proposed here represents a methodological solution for the extension of the observer to higher-dimensional domains and other classes of problems.

The paper is organized as follows. Sect. II summarizes the facts used in the subsequent developments on Penns' bio-heat equation and the observer proposed in [2]. Sect. II describes the neural networks used to solve the PDEs of interest, while Sect. IV proposes simulation results validating the approach. A brief conclusion is drawn in Sect. V.

II. Observer design

Temperature measurements during hyperthermia treatments are currently obtained through the use of invasive probes inserted into the patient's body at specific locations. To enable real-time temperature control with a minimally invasive approach, we propose to estimate the temperature in the whole target volume using superficial measurements. In this section, we detail the working conditions within which the presented contribution has been developed. To cope with the complexity of the hyperthermia treatment that involves electromagnetic and heat transfer phenomena, the problem is attacked in a modular way by first considering heat transfer in human tissues, assuming the thermal flow at the boundary as an input variable. The cooling effect of the water bolus [3] used in hyperthermia treatments is out of the scope of the current work, but our approach allows the extension of the model and analysis to be included in future work.

The heating phenomenon in biological tissues is usually modelled using Pennes' equation [1]

$$\rho c \partial_t T = k_{\text{eff}} \partial_{xx} T - \rho_b c_b \omega_b (T - T_a) + \mathcal{Q},\ x \in \Omega,\ t \in [0, t_f], \tag{1}$$

where: $\partial_t = \frac{\partial}{\partial t}$, $\partial_{xx} = \frac{\partial^2}{\partial^2_{xx}}$; T is the temperature and it is a function of space x and time t; ρ and c are: the density and specific heat of the tissue respectively; k_{eff} is its thermal effective conductivity; ρ_b and c_b are the density and specific heat of blood; ω_b is the blood perfusion rate, i.e., the volumetric blood flow rate per tissue volume [7]; T_a is the arterial blood temperature, constant at 37 °C [22]. The parameters ρ, c, ρ_b, c_b and k_{eff} can be considered constant for each individual, ω_b is uncertain and can vary within a certain range that is specific for each tissue.

The left-hand side of Eq. (1) expresses the variation of the thermal energy storage. The first term on the right-hand side is the net transfer of thermal energy due to temperature gradients, the second is the blood perfusion term with its uncertain coefficient ω_b. The term $\mathcal{Q}$ is an internal heat source comprehending the metabolic heat generation rate and, in the case of hyperthermia treatments, the power due to the radiating electric field [23].

Most of the existing methods for fast simulation of this equation are based on conventional numerical methods (e.g., Finite-Difference Time-Domain method), forming a global nonlinear system of equations for temperature solution. These methods are mainly focused on numerical accuracy, convergence, and stability rather than computation time, and hence are only suitable for pre-treatment predictive analysis.

In this work, we propose a NN-based implementation of the observer presented in [2] that requires the solution of (1) in real-time. To achieve this objective we use a system of deep neural networks trained to simulate the system and the prediction models. Each prediction model is characterized by a value of the perfusion coefficient ω_b varying in an admissible interval that includes also the value of the perfusion coefficient of the system. The closer the value of the perfusion coefficient of the prediction model to that of the system, the smaller the prediction error. Once trained, these networks are interrogated simultaneously to generate the error that feeds the MM observer developed in [2] and summarized in Sect. II-B.

A. Scaling of Pennes' equation

To avoid numerical instability of PINNs it is useful to determine a dimensionless, scaled version of the Eq. (1). To this end, operate a temperature translation $T' = T - T_a$ and define a dimensionless temperature as $\theta = \frac{T'}{T_M - T_a}$, where $T_M = 45$ °C is assumed as the maximum tolerable skin temperature [24]. Introduce also the characteristic length $L_0 = 5$ cm, which is the typical heating range of a superficial hyperthermia antenna applicator. Space coordinates are transformed as $X = \frac{x}{L_0}$, yielding:

$$\partial_{xx} T' = \partial_{XX} T' \left(\partial_x X\right)^2 = \partial_{XX} T' \frac{1}{L_0^2}. \tag{2}$$

From the definition of θ one gets:

$$T' = \theta (T_M - T_a), \tag{3}$$

$$\partial_t T' = \partial_\theta T' \partial_t \theta = (T_M - T_a) \partial_t \theta. \tag{4}$$

Likewise:

$$\begin{aligned} \partial_{\mathrm{X}} T' &= \partial_\theta T' \partial_{\mathrm{X}} \theta = (T_M - T_a) \partial_{\mathrm{X}} \theta, \\ \partial_{XX} T' &= (T_M - T_a) \partial_{XX} \theta. \end{aligned} \tag{5}$$

Substituting Eq. (5) in Eq. (2) and the result in Eq. (1) together with Eq. (4) and Eq. (3), introducing thermal diffusivity $\alpha = \frac{\rho c}{k_{\text{eff}}}$ and rearranging the equation we obtain:

$$\partial_t \theta = \frac{1}{\alpha L_0^2} \partial_{XX} \theta - L_0^2 \frac{\rho_b c_b \omega_b}{k_{\text{eff}}} \theta + \mathcal{Q} \frac{L_0^2}{k_{\text{eff}} (T_M - T_a)}.$$

Introduce a change in the time variable $\tau = \frac{t}{\tau_f}$, with $\tau_f = 1800s$, the typical time span of a treatment. Hence: $\partial_t \theta = \frac{1}{\tau_f} \partial_\tau \theta$. The adimensional bioheat equation reads as

$$\partial_\tau \theta = a_1 \partial_{XX} \theta - a_2 W \theta + \mathcal{Q} \frac{L_0^2}{k_{\text{eff}}}, \tag{6}$$

where $a_1 = \frac{\tau_f}{\alpha L_0^2}$, $a_2 = \frac{c_b L_0^2}{k_{\text{eff}}}$, and $W = \rho_b\, \omega_b$.

B. Multiple-model observer

Eq. (6) is a parabolic reaction-diffusion PDE that can be written in the form

$$\partial_\tau \theta = \sigma \partial_{XX} \theta - \omega \theta + \Psi,$$

where $\sigma = a_1$ is a known constant, $\omega = a_2 W$ is an uncertain perfusion coefficient, with a_2 known constant and W an uncertain perfusion rate, and $\Psi = \mathcal{Q}\frac{L_0^2}{k_{\text{eff}}}$ is a known heat source.

In the companion paper [2], we have proved exponential convergence of the L^2-norm of the error $\|\theta(\cdot,\tau)-\hat{\theta}(\cdot,\tau)\|_{L^2}$, where $\hat{\theta}$ is an estimate of θ provided by the observer

$$\partial_\tau \hat{\theta} = \sigma \partial_{XX} \hat{\theta} - \omega \hat{\theta} + \Psi, \tag{7}$$

defined for $(X,\tau) \in [0,1]\times[0,1]$ with boundary conditions

$$\begin{aligned} \hat{\theta}(0,t) &= C_0, \\ \partial_X \hat{\theta}(1,\tau) &= v(\tau) + \alpha(y(\tau) - \hat{\theta}(1,\tau)), \end{aligned} \tag{8}$$

where $\alpha > 0$ is the output injection gain.

In particular, the error norm converges to zero if the perfusion coefficient ω is equal to that of the system, described by Eq. (6), $\omega_r = a_2 W$, otherwise, it is ultimately bounded by the square of the difference between ω in Eq. (7) and the actual value ω_r. This result has suggested the use of a MM observer to reduce the uncertainty on the perfusion coefficient ω so as to get better estimation results.

Considering that, although variable, the perfusion coefficient takes value in a bounded range, we can select a set of fixed values $\omega_j \geq 0$, $j = 1, ..., N$, for some $N \in \mathbb{N}$, with:

$$\begin{gathered} \omega_1 < \omega_2 < \cdots < \omega_N, \\ \omega_1 \leq \omega_{\min}, \quad \omega_{\max} \leq \omega_N. \end{gathered}$$

Accordingly, we can define a family of observers $\hat{\theta}^{(j)}(X,\tau)$ with the structure (7) and with the choice $\omega = \omega_j$, that is:

$$\begin{aligned} \partial_\tau \hat{\theta}^{(j)} &= \sigma \partial_{XX} \hat{\theta}^{(j)} - \omega_j \hat{\theta}^{(j)} + \Psi, \\ \hat{\theta}^{(j)}(0,\tau) &= 0, \\ \partial_{XX} \hat{\theta}^{(j)}(1,\tau) &= v(\tau) + \alpha(y(\tau) - \hat{\theta}^{(j)}(1,\tau)). \end{aligned} \tag{9}$$

As proved in [2], for each observer within this family, the corresponding estimation error can be bounded as follows

$$\|\theta(\cdot,\tau) - \hat{\theta}^{(j)}(\cdot,\tau)\|_{L^2}^2 \leq M\,|\omega_j - \omega|^2, \tag{10}$$

where M depends on a maximum value of the temperature[1], on the system parameter a_1 and on output injection gain α.

Following the approach proposed in [25], the idea is then to introduce an overall estimator

$$\hat{\theta}^{\dagger}(X,\tau) = p_1(\tau)\hat{\theta}^{(1)}(X,\tau) + \cdots + p_N(\tau)\hat{\theta}^{(N)}(X,\tau), \tag{11}$$

obtained as dynamic convex combination of the observers $\hat{\theta}^{(j)}(X,\tau)$, with weights $p_j(\tau)$ being updated according to

$$\dot{p}_j(\tau) = -\lambda \left(1 - \frac{e^{-\mu_j(\tau)}}{\sum_{\ell=1}^{N} p_\ell(\tau) e^{-\mu_\ell(\tau)}}\right) p_j(\tau) \quad j = 1, ..., N, \tag{12}$$

where $\mu_j(\tau) := |\theta(1,\tau) - \hat{\theta}^{(j)}(1,\tau)|$ is the (absolute) output error and $\lambda > 0$ is the adaptive gain.

The overall observer (11) corresponds to a weighted average of the individual observers, whose weights adapt based on the size of the associated output errors. In particular, the weight $p_j^\star$ associated with the observer $\hat{\theta}^{(j^\star)}$ that asymptotically shows a smaller error with respect to the others is expected to tend to 1, while the remaining ones tend to zero [2]. This means that the estimation (7) will tend to the prediction of the *best* observer among the considered family, with respect to the inherent output error function.

It is worth noticing that the finer the gridding of $[\omega_{\min}, \omega_{\max}]$ is (i.e., the larger the number N of models), the smaller the error bound (10) associated with the best observer $\hat{\theta}^{(j^\star)}$. The number N of models, however, cannot be made arbitrarily large without incurring numeric instabilities [2]. This practical limitation has motivated the computational model presented in this work.

III. NN-BASED OBSERVER IMPLEMENTATION

This section presents the implementation of the observer in [2] for a 1D spatial domain using a system of PINNs [20] to solve Pennes' Eq. (6) and the N observers Eqs. (9).

As described in Sect. II-B, the predictions of the observers are used to provide the estimation of the temperature (11), in the whole domain at each time instant τ, through a convex combination of weights with the dynamics (12).

The novel application of PINNs to this setting enabled the creation of a pipeline that is both effective in terms of accuracy and efficient in terms of overall computational time. All the PINNs used in our implementation share a similar structure and hyperparameters: a 4-layer perceptron, with each layer containing 20 hidden neurons; $\tanh$ activation function and parameters initialization according to the Glorot uniform method; training for 20000 epochs using stochastic gradient descent and, in particular, the "Adam" variant with a learning rate equal to 10^{-3}.

A. PINN for solving Pennes' equation

Fig. 1 (left) shows the Neural Bio-Heat System (NBHS) used to solve the PDE associated with the system (6). It has been trained with a specific value of the perfusion rate W and boundary and initial conditions of the type

$$\begin{aligned} \theta(0,\tau) &= 0, \\ \partial_X \theta(1,\tau) &= v(\tau), \\ \theta(X,0) &= \theta_0(X). \end{aligned} \tag{13}$$

Coherently with [2], we have set the internal heat source to zero, while the temperature on the left boundary of the 1D domain is assumed to be constant and equal to T_a, which corresponds to $\theta(0,\tau) = 0$.

The NBHS is implemented using the DeepXDE library [21]. A preliminary scaling of the input variables (x,t) is operated as illustrated in the previous section, while the differential constraints are represented by the PDE (6) with boundary and initial conditions (13). The loss function $\mathcal{L}$ measures the discrepancy between the neural network solution θ and the constraints represented by the PDE and the boundary conditions.

After the training, NBHS will take the two coordinates (X,τ) as input, and will return $\theta(X,\tau)$. This value will be then scaled and translated to obtain $T(X,\tau)$:

$$T(X,\tau) = T_a + \theta(X,\tau)\cdot(T_M - T_a).$$

[1]For hyperthermia treatments this can be set equal to 45° C.

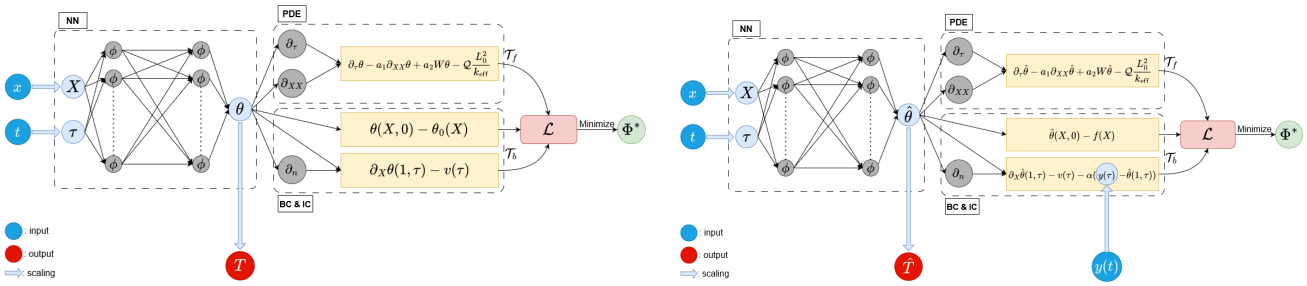

Fig. 1: Left: a scheme of the computational model used to solve Eq. (6) (NBHS). Right: the observer (9) (NBHO).

Using a PINN, in this case, presents several advantages. In fact, this approach allows probing the system in a continuous manner and in real-time, meaning that the computation time is that of a forward pass of a deep neural network, which usually takes a few milliseconds on most hardware. Moreover, this approach can potentially scale to multiple input dimensions seamlessly. Alternatively, the use of a solver for each setting is necessary, with apparent drawbacks in terms of computational time.

B. Implementing a multiple-model observer with PINNs

Each of the N observers of the form (9) is implemented by means of a second PINN shown in Fig. 1 (right), which we name Neural Bio-Heat Observer (NBHO), trained with the same left boundary (i.e., $X = 0$) condition as NBHS. Initial and right-side boundary conditions are different from NBHS. In particular, the latter needs the PINN architecture to be augmented with an extra input (see Fig. 1, right), that is the solution of the system equation on the boundary (i.e., a measure of the temperature on the boundary): $y(\tau) = \theta(1, \tau)$. This structure makes it possible to tackle control problems. In particular, the right boundary condition can now be expressed as:

$$\partial_X \hat{\theta}(1, \tau) = v(\tau) + \alpha(y(\tau) - \hat{\theta}(1, \tau)),$$

i.e., on the right boundary, the system heat flux is corrected with the (adimensional) temperature estimation error at the same boundary. The choice of the output injection gain α is the result of an optimization procedure. Different values of $\alpha \in [0.5, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10]$ have been used to train several NBHO. The value $\alpha = 4$ used in the simulations minimizes the L^2 estimation error norm at $\tau = 1$.

In the MM estimator (11), each individual observer NBHO$_j$, described by Eq. (9), is implemented with a PINN replicating the NBHO structure. The difference between the N observers is in the different perfusion rate $W_j \in [0.45, 4]$, characterizing the muscle tissue[2], used for their training.

In accordance with [2], the results presented in the next section confirm that the adaptive weights tend to concentrate on the observer trained with the perfusion rate W_j, which is the closest to the actual value for the average perfusion.

IV. Simulations

In the following set of simulations, we first present the solution of Eq. (6) for a given set of initial and boundary conditions, then we validate the single observer trained with the same value of the perfusion rate of the system. To validate the MM observer, we first report the results comparable with those obtained with the Matlab implementation in [2], then a simulation showing convergence to the best estimator that was impossible to achieve with the Matlab implementation presented in [2].

Consider the adimensional bio-heat equation (6) for a muscular tissue without an internal source, i.e. setting $Q \equiv 0$, with coefficients taken accordingly to [22], as reported in the following table:

TABLE I: Parameters of the bio-heat equation.

Par.	$\varrho,\ \varrho_b$	$c,\ c_b$	κ_{eff}	T_a
Value	1050, 1043	3639, 3825	5	37
Units	$[kg/m^3]$	$[J/(kg{\cdot}K)]$	$[W/(m{\cdot}K)]$	$^\circ C$

The perfusion rate of the system is $W = 2.3\ [kg/(m^3{\cdot}s)]$, with an admissible range $[W_{\min}, W_{\max}] = [0.45, 4]$. Initial and boundary conditions (IC and BC, respectively) have been chosen as follows

$$\begin{aligned} \theta(X,0) &= \tfrac{q_0}{(T_M - T_a)} \tfrac{X^4}{4} + \tfrac{\beta}{(T_M - T_a)} X(X-1)^2 \quad \forall X \in [0,1], \\ \theta(0,\tau) &= 0 \quad \forall \tau \geq 0, \\ \partial_X \theta(1,\tau) &= \tfrac{q_0}{(T_M - T_a)} \quad \forall \tau \geq 0, \end{aligned}$$

with $q_0 = 16$ and $\beta = 15$. The rescaled solution of the equation is depicted in Fig. 2 (Left). Fig. 2 (Right), shows the difference between Matlab and PINNs prediction of the temperature (in °C) in the considered domain for a treatment duration of 1800 s.

It is worth noticing the difference in the initial temperature value across the whole domain, even though the same IC has been used in both the Matlab and PINNs computational models. This is because while in Matlab the IC is numerically imposed, the IC in PINNs acts as a soft constraint which is included in an optimization process. The degree to which this soft constraint is met can be adjusted based on network hyperparameters, including the number of training epochs and the weighting of the loss function. A similar discussion applies to the BC at $x = 1$. Conversely, the BC at $x = 0$ has been treated as a hard constraint by transforming the network's output [21], ensuring an exact match with the specified condition.

Example 4.1: We begin by considering the case of known perfusion rate W. We consider an observer with output

[2]Note that $\omega_j = a_2 W_j$, with a_2 constant.

Fig. 2: Left: prediction of the solution of the bio-heat equation. Right: comparison between Matlab and PINNs prediction.

injection gain $\alpha = 4$ and initial condition:

$$\hat{\theta}(X,0) = \frac{q_0}{(T_M - T_a)} \frac{x^4}{4}. \tag{14}$$

The observer $\hat{\theta}(X,\tau)$ successfully achieves the reconstruction of the temperature $\theta(X,\tau)$ over the whole domain $[0,1]$. The rescaled temperature profile $T(x,t_f)$ along with the rescaled estimated temperature profile $\hat{T}(x,t_f)$ at the final time $t_f = 1800s$ are shown in Fig. 3, while Fig. 6 (left) shows the exponential decay of the L^2-norm of the prediction error $T(x,t) - \hat{T}(x,t)$.

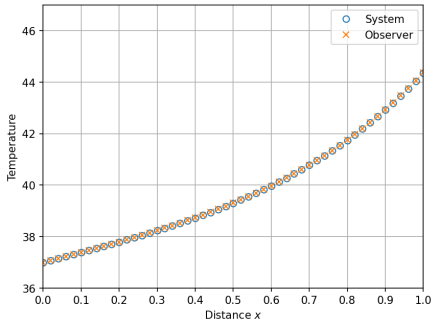

Fig. 3: Example 1: Solution vs. observer at $t = t_f$.

Example 4.2: Let us now assume that W is only known to belong to the interval $[W_{\min}, W_{\max}]$, and consider the MM approach synthesized in Sect. II-B, with $N = 8$ models computed through a uniform gridding $\{W_1, W_2, \ldots, W_8\}$ of the admissible variation interval of W, so that the closest value to the actual perfusion rate is W_5. Each model $\hat{\theta}^j(X,\tau)$, $j = 1, 2, ..., 8$, has been initialized as in (14). The results for two different values of the parameter λ in (12) are given: $\lambda \in \{5, 200\}$.

The comparison, after appropriate rescaling, of system solution $T(x,t)$, observers $\hat{T}^{(j)}(x,t)$ and MM adaptive observer $\hat{T}^{\dagger}(x,t)$ at the final time $t_f = 1800\,s$ is given in Fig. 4, along with an enlarged version for helping visualization of results. As clearly visible, the actual system is quite accurately reconstructed by the MM adaptive observer $\hat{T}^{\dagger}(x,t)$. The latter is built using the dynamic weights $p_j(x,t)$, whose time evolution is reported in Fig. 5 for the two different values of λ.

Finally, the evolution of the L^2-norm of the estimation error $e_T = T(x,t) - \hat{T}^{\dagger}(x,t)$ is depicted in Fig. 6 (right). Concerning the results obtained with Matlab, reported in the companion paper [2], the implementation based on PINNs allows the convergence of the observer with the same output

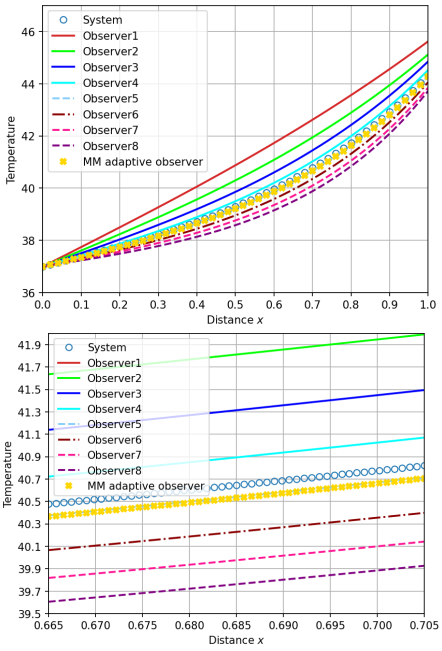

Fig. 4: Example 2: Solution vs. observers at $t = t_f$.

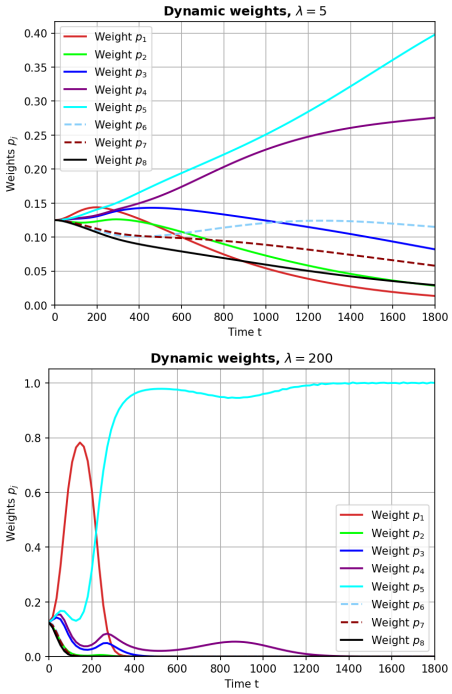

Fig. 5: Example 2: Dynamic weights, $\lambda = 5$ and $\lambda = 200$.

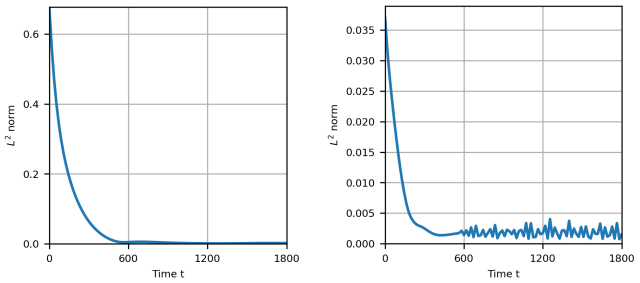

Fig. 6: L^2-norm of the estimation error. Left: Example 1. Right: Example 2.

injection gain $\alpha = 4$ ($\alpha = 32$ for unscaled variables) for both the cases of single and MM observer. With Matlab implementation, it was necessary to raise it by at least three orders of magnitude to avoid incurring numerical instability. The use of a smaller α allowed the achievement of a faster convergence of the dynamic weights in Python. Results obtained with $\lambda = 5$ and $\alpha = 4$ are similar to the ones obtained in Matlab with $\lambda = 200$ and $\alpha = 3 \cdot 10^4$. Smaller values of α in the Matlab implementation led to the progressive reduction of the solver integration step until reaching the minimum size and integration failure. With $\lambda = 200$ in Python implementation, the convergence of the weight $p_5(t)$ is practically reached after about 400 s. It is important to note that this result was impossible to obtain in Matlab using the same computer.

V. Conclusion

Adaptive estimation of the Pennes' Bio-Heat equation via Deep Neural Networks shows significant improvements with respect to the implementation in [2] that relies on the 1D PDE solver of Matlab. Results achieved in [2] are recovered with this novel approach with increased numerical stability which enabled convergence results that were impossible to achieve for the Matlab solver. The accuracy of the NN-based model also appears improved. Future work includes the optimization of these preliminary results and the extension to higher-dimensional domains.

References

[1] H. H. Pennes, "Analysis of tissue and arterial blood temperatures in the resting human forearm," *Journal of Applied Physiology*, vol. 1, no. 2, pp. 93–122, 1948.

[2] A. Cristofaro, G. Cappellini, E. Staffetti, G. Trappolini, and M. Venditelli, "Adaptive estimation of the Pennes' Bio-Heat equation - I: Observer design," in *Proceedings of the 62nd IEEE Conference on Decision and Control*, Singapore, December 13-15, 2023.

[3] K. R. Foster, H. N. Kritikos, and H. P. Schwan, "Effect of surface cooling and blood flow on the microwave heating of tissue," *IEEE Transactions on Biomedical Engineering*, vol. BME-25, no. 3, pp. 313–316, 1978.

[4] K. R. Foster, A. Lozano-Nieto, P. J. Riu, and T. S. Ely, "Heating of tissues by microwaves: A model analysis," *Bioelectromagnetics*, vol. 19, no. 7, pp. 420–428, 1998.

[5] K. R. Foster, M. C. Ziskin, and Q. Balzano, "Thermal response of human skin to microwave energy: A critical review," *Health Physics*, vol. 111, no. 6, pp. 528–541, 2016.

[6] P. Bernardi, M. Cavagnaro, S. Pisa, and E. Piuzzi, "SAR distribution and temperature increase in an anatomical model of the human eye exposed to the field radiated by the user antenna in a wireless LAN," *IEEE Transactions on Microwave Theory and Techniques*, vol. 46, no. 12, pp. 2074–2082, 1998.

[7] J. Zhang, G. A. Sandison, J. Y. Murthy, and L. X. Xu, "Numerical simulation for heat transfer in prostate cancer cryosurgery," *Journal of Biomechanical Engineering*, vol. 127, no. 2, pp. 279–294, 2005.

[8] S. Tanwar, L. Famhawite, and P. R. Verma, "Numerical simulation of bio-heat transfer for cryoablation of regularly shaped tumours in liver tissue using multiprobes," *Journal of Thermal Biology*, vol. 113, p. 103531, 2023.

[9] H. P. Kok, E. N. K. Cressman, W. Ceelen, C. L. Brace, R. Ivkov, H. Grüll, G. ter Haar, P. Wust, and J. Crezee, "Heating technology for malignant tumors: A review," *International Journal of Hyperthermia*, vol. 37, no. 1, pp. 711–741, 2020.

[10] N. Datta, S. G. Ordóñez, U. Gaipl, M. Paulides, H. Crezee, J. Gellermann, D. Marder, E. Puric, and S. Bodis, "Local hyperthermia combined with radiotherapy and-/or chemotherapy: Recent advances and promises for the future," *Cancer Treatment Reviews*, vol. 41, no. 9, pp. 742–753, 2015.

[11] P. B. Elming, B. S. Sørensen, A. L. Oei, N. A. Franken, J. Crezee, J. Overgaard, and M. R. Horsman, "Hyperthermia: The optimal treatment to overcome radiation resistant hypoxia," *Cancers*, vol. 11, no. 1, pp. 1–20, 2019.

[12] A. Bakker, J. van der Zee, G. van Tienhoven, H. P. Kok, C. R. N. Rasch, and H. Crezee, "Temperature and thermal dose during radiotherapy and hyperthermia for recurrent breast cancer are related to clinical outcome and thermal toxicity: A systematic review," *International Journal of Hyperthermia*, vol. 36, no. 1, pp. 1023–1038, 2019.

[13] T. Ohguri, Y. Harima, H. Imada, H. Sakurai, T. Ohno, Y. Hiraki, K. Tuji, M. Tanaka, and H. Terashima, "Relationships between thermal dose parameters and the efficacy of definitive chemoradiotherapy plus regional hyperthermia in the treatment of locally advanced cervical cancer: Data from a multicentre randomised clinical trial," *International Journal of Hyperthermia*, vol. 34, no. 4, pp. 461–468, 2018.

[14] M. Ferro, P. Pavoni, and M. Venditelli, "ROBHOT: A robot-assisted technology for superficial hyperthermia treatments," in *Video contributions to the 2023 IEEE International Conference on Robotics and Automation*, London, UK, May 29 - June 2, 2023.

[15] A. Smyshlyaev and M. Krstic, "Backstepping observers for a class of parabolic PDEs," *Systems & Control Letters*, vol. 54, no. 7, pp. 613–625, 2005.

[16] T. Ahmed-Ali, F. Giri, M. Krstic, F. Lamnabhi-Lagarrigue, and L. Burlion, "Adaptive observer for a class of parabolic PDEs," *IEEE Transactions on Automatic Control*, vol. 61, no. 10, pp. 3083–3090, 2016.

[17] T. Kharkovskaia, D. Efimov, E. Fridman, A. Polyakov, and J.-P. Richard, "On design of interval observers for parabolic PDEs," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 4045–4050, 2017.

[18] W. Hu and M. A. Demetriou, "Observer design with sparsity for parabolic PDEs," *IFAC-PapersOnLine*, vol. 52, no. 2, pp. 180–182, 2019.

[19] G. E. Karniadakis, I. G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, and L. Yang, "Physics-informed machine learning," *Nature Reviews Physics*, vol. 3, no. 6, pp. 422–440, 2021.

[20] M. Raissi, P. Perdikaris, and G. Karniadakis, "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," *Journal of Computational Physics*, vol. 378, pp. 686–707, 2019.

[21] L. Lu, X. Meng, Z. Mao, and G. E. Karniadakis, "DeepXDE: A deep learning library for solving differential equations," *SIAM Review*, vol. 63, no. 1, pp. 208–228, 2021.

[22] M. L. Van der Gaag, M. De Bruijne, T. Samaras, J. Van der Zee, and G. C. Van Rhoon, "Development of a guideline for the water bolus temperature in superficial hyperthermia," *International Journal of Hyperthermia*, vol. 22, no. 8, pp. 637–656, 2006.

[23] J. Lang, B. Erdmann, and M. Seebass, "Impact of nonlinear heat transfer on temperature control in regional hyperthermia," *IEEE Transactions on Biomedical Engineering*, vol. 46, no. 9, pp. 1129–1138, 1999.

[24] H. D. Trefná, H. Crezee, M. Schmidt, D. Marder, U. Lamprecht, M. Ehmann, J. Hartmann, J. Nadobny, J. Gellermann, N. Van Holthe, P. Ghadjar, N. Lomax, S. Abdel-Rahman, C. Bert, A. Bakker, M. D. Hurwitz, C. J. Diederich, P. R. Stauffer, and G. C. Van Rhoon, "Quality assurance guidelines for superficial hyperthermia clinical trials: I. clinical requirements," *International Journal of Hyperthermia*, vol. 33, no. 4, pp. 471–482, 2017.

[25] V. Hassani, A. P. Aguiar, A. M. Pascoal, and M. Athans, "Further results on plant parameter identification using continuous-time multiple-model adaptive estimators," in *Proceedings of the 48h IEEE Conference on Decision and Control held jointly with the 2009 28th Chinese Control Conference*, Shanghai, China, December 15-18, 2009.