



SAPIENZA
UNIVERSITÀ DI ROMA

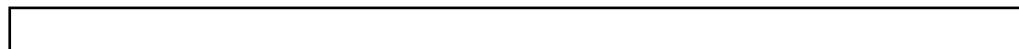


UNIONE EUROPEA
Fondo Sociale Europeo



REACT EU

AI-based solution methods for PDEs with application to Oncological Hyperthermia



Sapienza Università di Roma

National PhD Program in Artificial Intelligence (XXXVII cycle)

Guglielmo Cappellini

ID number 1643022

Advisor

Prof. M. Vendittelli

Co-Advisors

Prof. A. Cristofaro

Prof. G. Trappolini

Academic Year 2024/2025



UNIONE EUROPEA
Fondo Sociale Europeo



REACT EU

La borsa di dottorato è stata cofinanziata con risorse del
Programma Operativo Nazionale Ricerca e Innovazione 2014-2020, risorse FSE REACT-EU
Azione IV.4 “Dottorati e contratti di ricerca su tematiche dell’innovazione”
e Azione IV.5 “Dottorati su tematiche Green”

Thesis defended on May 30th, 2025
in front of a Board of Examiners composed by:

Prof. A. Chella (chairman)
Prof. S. Costantini
Prof. M. Matteucci

Reviewers:

Prof. H. Crezee, Amsterdam UMC.
Prof. N. Tonellotto, University of Pisa.

AI-based solution methods for PDEs with application to Oncological Hyperthermia

PhD thesis. Sapienza University of Rome

© 2025 Guglielmo Cappellini. All rights reserved

This thesis has been typeset by L^AT_EX and the Sapthesis class.

Author's email: cappellini@diag.uniroma1.it

*Dedicated to
my family.*

Abstract

This thesis investigates the observability of temperature distribution within the Gross Tumor Volume during superficial hyperthermia treatment, an adjuvant therapy employed alongside radiotherapy or chemotherapy for relapses and treatment-resistant tumors.

A critical clinical challenge is addressed: the real-time assessment of temperature at the target site—essential for treatment feedback. Current protocols rely on invasive thermocouples, causing significant patient discomfort while providing only discrete, imprecisely localized measurements. This challenge is further compounded by uncertainties in patient-specific properties, particularly blood perfusion rates that fluctuate during tissue heating. We aim at predicting the temperature distribution inside a mono-dimensional domain represented by an ideal needle that enters perpendicularly from the surface of the patient, relying only on superficial boundary measurements. Given the uncertainties in the properties of the patient, this study is framed as a state estimation problem of the governing physical law of heat transfer in biological systems—Pennes' Bio-Heat Equation—rather than a direct solution of the equation itself.

The methodology is based on a multiple-model adaptive estimation framework using an adaptive observer design. The validation process has highlighted some limitations in traditional numerical simulation approaches, that served as a prompt to implement the estimator using Deep Learning techniques. Specifically, Physics-Informed Learning is leveraged, an approach particularly suited to scenarios with known governing equations but limited training data. This Physics-Informed approach offers three key advantages: elimination of domain discretization requirements, enabling continuous spatial-temporal probing; real-time prediction of the solution to partial differential equation; and scalability to higher-dimensional input spaces without the constraints of conventional numerical methods.

The final section extends our methodology to wave equation solutions for the deformation of soft material, with applications in mixed reality surgical training environments. This interdisciplinary research integrates applied mathematics, deep learning, and control engineering to address complex clinical challenges.

Keywords: Physics-informed learning, Adaptive state estimation, Oncological hyperthermia, Bio-heat transfer, Mixed reality

Contents

| | | |
|----------|--------------------------------------------------------------------------|-----------|
| 1 | Introduction | 1 |
| 1.1 | Context and objectives | 1 |
| 1.2 | Thesis outline | 2 |
| 1.3 | Thesis Contributions | 2 |
| 1.4 | Notation | 3 |
| 2 | Oncological Hyperthermia | 5 |
| 2.1 | Biological rationale | 5 |
| 2.2 | Devices and procedures | 6 |
| 2.2.1 | Methods of heating | 6 |
| 2.2.2 | Equipment for superficial hyperthermia | 7 |
| 2.2.3 | Metrics of interest in HT routine | 8 |
| 2.3 | Thermal modeling | 8 |
| 2.3.1 | Pennes' Bio-heat equation | 9 |
| 2.3.2 | Effect of discrete blood vessels | 14 |
| 2.3.3 | Tissue property and perfusion uncertainties in HT | 16 |
| 2.4 | Current limitations of Hyperthermia treatment | 17 |
| 2.5 | Potential of Hyperthermia in cancer treatment | 17 |
| 3 | Machine Learning | 19 |
| 3.1 | Introduction | 20 |
| 3.2 | Mathematical preliminaries | 21 |
| 3.2.1 | Gradients and Jacobians | 21 |
| 3.2.2 | Gradient descent | 22 |
| 3.3 | Datasets and losses | 23 |
| 3.4 | Linear models | 24 |
| 3.5 | Feedforward neural networks | 24 |
| 3.6 | Backpropagation | 25 |
| 3.7 | Hyper-parameters | 27 |
| 3.7.1 | Activation functions | 27 |
| 3.7.2 | Optimizers | 27 |
| 3.7.3 | Weights initialization | 28 |
| 3.8 | Artificial Intelligence for Health | 29 |
| 3.8.1 | Physics-Informed Neural Networks | 30 |
| 3.9 | Improving the training and establishing a validation benchmark | 32 |
| 3.9.1 | Tools and packages | 32 |

| | | |
|----------|-------------------------------------------------------------------------------|-----------|
| 3.9.2 | Hyper-parameters optimization | 32 |
| 3.9.3 | Other techniques | 33 |
| 4 | A State Observer of PBHE | 35 |
| 4.1 | Boundary control of PDEs | 36 |
| 4.1.1 | Stability analysis | 36 |
| 4.1.2 | Nonlinearities in bioheat transfer | 37 |
| 4.1.3 | Proposed approach: multiple-model adaptive observer | 38 |
| 4.2 | Problem formulation | 38 |
| 4.3 | Observer structure | 39 |
| 4.3.1 | Nominal case (known perfusion) | 39 |
| 4.3.2 | Unknown perfusion | 41 |
| 4.3.3 | Discussion | 43 |
| 4.4 | Multiple-model adaptive scheme | 45 |
| 4.5 | Conclusion | 47 |
| 5 | PINNs-based implementation of the multiple-model observer | 49 |
| 5.1 | Direct problem | 50 |
| 5.1.1 | Neural Bio-Heat System (NBHS) | 51 |
| 5.2 | Inverse problem | 55 |
| 5.3 | Adaptive estimation of PBHE | 57 |
| 5.3.1 | Neural Bio-Heat Observer (NBHO) | 57 |
| 5.3.2 | Model | 58 |
| 5.3.3 | Results | 59 |
| 5.3.4 | Overcoming MATLAB limitations | 62 |
| 5.4 | Conclusion | 63 |
| 6 | Application of Physics-informed multiple-model observer to a real case | 65 |
| 6.1 | Materials and methods | 66 |
| 6.1.1 | Preliminary considerations | 66 |
| 6.1.2 | Muscle-equivalent phantom and position of the probes | 68 |
| 6.1.3 | Applicator characteristics, heating procedure | 69 |
| 6.2 | Main hypothesis and model | 70 |
| 6.3 | Overview of the measurements | 72 |
| 6.4 | Results | 76 |
| 6.4.1 | Inverse problem | 76 |
| 6.4.2 | Adaptive estimation | 77 |
| 6.4.3 | Considerations on in-depth ground truth measurements | 80 |
| 6.5 | Conclusion | 83 |
| 7 | A novel application: the wave equation | 85 |
| 7.1 | Introduction | 85 |
| 7.2 | The wave equation for vibrating strings | 87 |
| 7.2.1 | The ideal string | 87 |
| 7.2.2 | Introducing stiffness effects | 87 |
| 7.2.3 | Scaling | 88 |

| | | |
|----------|--------------------------------------------------------|------------|
| 7.3 | PINNs for simulation of deformable materials | 89 |
| 7.4 | Experiments | 90 |
| 7.4.1 | Plucked string | 90 |
| 7.5 | Discussion | 92 |
| 7.6 | Further developments | 93 |
| 7.6.1 | Struck string and membrane | 93 |
| 7.6.2 | Mathematical formulation of the PINC problem | 93 |
| 7.7 | Conclusions | 94 |
| 8 | Conclusions | 95 |
| 8.1 | Contributions | 95 |
| 8.2 | Future works | 96 |
| A | HPO of PINNs | 97 |
| A.1 | Choice of sampling points | 97 |
| A.2 | Optimization of PINN architecture | 100 |
| A.3 | Choice of output injection gain for the NBHO | 101 |
| B | Companion figures | 103 |
| B.1 | Cooling 1 | 104 |
| B.2 | Cooling 2 | 107 |
| C | Abstract in Italiano | 111 |
| | Acronyms | 111 |
| | Glossary | 115 |
| | Bibliography | 121 |

Chapter 1

Introduction

1.1 Context and objectives

Hyperthermia (HT) consists of heating tumors in the range from 40 to 43 [°C], and can be applied as a sensitizer to Radiotherapy (RT) and/or Chemotherapy (CH) [14]. In fact, heating of tissues enables some biological mechanisms that, in combination with conventional oncological treatment, has been proven effective for several malignancies e.g. melanomas, sarcomas and bladder tumors, recurrent breast cancer, cervix uteri and rectum tumors [76, 59, 71, 49, 33, 7].

In general, HT treatments last for one hour and the goal temperature is 43 [°C]. The effectiveness is critically related to achieving adequate therapeutic tumor temperatures in a reproducible manner across treatments, but a safe application also requires avoiding excessive heating outside the target region. For loco-regional HT treatment of tumors, Electromagnetic (EM) heating is applied using phased-array systems that use constructive interference to focus the power deposition at a specified target location [38]. The EM power deposition is expressed using the Specific Absorption Rate (SAR) [W kg^{-1}]. In the superficial HT, tumors up to 4 [cm] depth are treated with a single EM applicator. Between the patient surface and the applicator, a water bolus is positioned to avoid skin burns and enhance antenna matching, together with a Thermocouples (TC) matrix named Thermal Monitoring Sheet (TMS) for measuring skin temperature (See Figure 1.1).

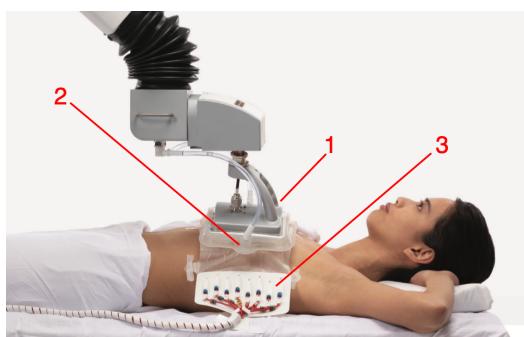


Figure 1.1. Superficial HT treatment with ALBA ON400D (From [52]). Three components can be identified: 1) the applicator that delivers EM power; 2) the water bolus to ensure good matching and avoid skin burns; 3) the TMS to monitor superficial temperature.

Currently, HT exhibits some limitations that complicate the integration into the oncology workflow. Treatments are conducted with limited feedback information, because the acquisition of target temperature necessitates the insertion of catheters into the patient's body or using natural cavities. This approach is often avoided since it yields discrete temperature readings that are difficult to localize, and induces significant discomfort in patients. Moreover, some characteristics of patients are uncertain, not directly measurable, and can vary in time, even during the same treatment, affecting the temperature distribution at the target.

For improving HT, the achievement of feedback information from noninvasive, boundary measurements would be beneficial. This implies the development of tools able to solve the equation that models the bio-heat transfer incorporating boundary conditions in real-time and in the presence of unknown parameters. Most of the existing approaches for fast simulation of bio-heat transfer are based on conventional numerical methods such as Finite-Difference Time-Domain (FDTD), forming a global nonlinear system of equations for temperature solution. These methods focus mainly on numerical accuracy, convergence, and stability, rather than minimizing computation time, hence are not suitable for our purpose. We propose to achieve the online adaptive estimation of bio-heat transfer employing a control engineering technique named multiple-model adaptive observer with deep learning-based implementation.

1.2 Thesis outline

The thesis is organized as follows: in Chapter 2 we provide an introduction to HT, showing the biological rationale, the most utilised devices and procedures, the mathematical model, current limitations and perspectives. Chapter 3 aims at providing insights on Machine Learning (ML), with some tools, common Neural Network (NN) architectures, and a focus on the application of ML for health. Chapter 4 deals with the mathematical formulation of a Multiple-model Adaptive Observer (MM-Obs), able to incorporate boundary measurements and predict the temperature distribution in the entire domain and in presence of unknown coefficients. In Chapter 5, numerical validation of the MM-Obs is performed and in Chapter 6 the application of the proposed model on an experiment mimicking a HT treatment is given. The last chapter exhibits a novel application, the solution of the wave equation with application to predict the deformation of tissues with application to training and planning of medical procedures in mixed reality.

1.3 Thesis Contributions

The work conducted throughout this thesis led to the following publications:

- Cristofaro, A.; **Cappellini, G.**; Staffetti, E.; Trappolini, G.; Vendittelli, M. “*Adaptive Estimation of the Pennes' Bio-Heat Equation - I: Observer Design.*” In: 2023 62nd IEEE Conference on Decision and Control (CDC), 2023, pp. 1931–1936.

- **Cappellini, G.**; Trappolini, G.; Staffetti, E.; Cristofaro, A.; Vendittelli, M. “*Adaptive Estimation of the Pennes’ Bio-Heat Equation - II: A NN-Based Implementation for Real-Time Applications.*” In: 2023 62nd IEEE Conference on Decision and Control (CDC), 2023, pp. 5364–5369.
- **Cappellini, G.**; Cristofaro, A.; Staffetti, E.; Trappolini, G.; Vendittelli, M. “*Adaptive Estimation of the Pennes’ Bio-Heat Equation.*” In: 2023 I-RIM Conference, 2023, pp. 202–206.

The following works have been submitted and are currently under review:

- De Santis, E.; **Cappellini, G.**; Vendittelli, M. “*Real-time Simulation of Deformable Tissues Using PINNs.*” Submitted to RO-MAN 2025.
- **Cappellini, G.**; Cristofaro, A.; De Santis, E.; Staffetti, E.; Trappolini, G.; Vendittelli, M. “*Adaptive Estimation of the Pennes’ Bio-Heat Equation: Observer Design and PINNs Implementation.*” Submitted to IEEE Transactions on Control Systems Technology (TCST).

1.4 Notation

Given a function $f : \mathbb{R}^n \rightarrow \mathbb{R}$, we denote by $\partial_{z_i} f$ the partial derivative of $f(z_1, \dots, z_n)$ with respect to the variable z_i , $i = 1, \dots, n$. The notation $L^2[0, 1]$ stands for the space of functions $f : [0, 1] \rightarrow \mathbb{R}$ whose squared integral is finite and corresponds to the L^2 -norm, that is

$$\|f\|_{L^2} := \left(\int_0^1 |f(z)|^2 dz \right)^{\frac{1}{2}} < +\infty.$$

Chapter 2

Oncological Hyperthermia

The abnormal cellular growth that characterises every tumor tissue, causes the latter to exhibit particular characteristics such as hypoxia and an acidic environment. These characteristics can be used to kill tumor cells selectively. HT consists of locally increasing temperature and is used in oncology to treat locally advanced tumors, recurrent tumors, and hard-to-treat tumors. This is because, especially in locally advanced tumors or local recurrence, there is often limited efficacy of conventional treatments (RT and/or CH) and more effective treatment strategies, or their combinations, are required.

There is an increasing interest in enhancing the precision of these treatments and facilitating the smooth incorporation of HT into the oncology workflow. This endeavor can be realized through the optimization of Hyperthermia Treatment Planning (HTP) in an offline phase while achieving non-invasive feedback control of treatments in an online phase.

This chapter is organized as follows: the first section delineates the biological rationale of HT, explaining the biological mechanisms exploited by this oncological treatment. The second section presents an overview of the devices, protocols, and metrics employed in clinical settings. Subsequently, we proceed to demonstrate the physical law widely adopted for modelling Bio-Heat transfer in biological systems, namely, Pennes' Bio-Heat Equation (PBHE). Additionally, insights into the limitations of this model and potential advancements are provided. In the concluding part of this chapter, the potential and actual limitations of HT are discussed.

2.1 Biological rationale

The use of heat in curing diseases has a long history [55]. From a free translation of an aphorism by Hippocrates [57]:

"Those who cannot be cured by medicine can be cured by surgery. Those who cannot be cured by surgery can be cured by hyperthermia. Those who cannot be cured by hyperthermia are indeed incurable."

Biologically, HT has two different types of interactions with radiation [58]. Firstly, heat has a radiosensitizing effect. This is most prominent with a simultaneous application, but is of the same magnitude in both tumor and normal tissue and

will not improve the therapeutic ratio unless the tumor is heated to a higher temperature than the normal tissue. Secondly, HT exhibits a direct cytotoxic effect, and moderate heat treatment alone can almost selectively destroy tumor cells. Hyperthermic cytotoxicity is seen as direct heat killing of cells in a deprived micro-environment, characterized by insufficient blood supply with subsequent poor nutrition and increased acidity due to anaerobic metabolism and accumulation of lactic acid and other waste products. Because such cells are the most radioresistant, a smaller radiation dose is needed to control the remaining more radiosensitive cells.

The magnitude of both the sensitizing and the cytotoxic effects depends on the temperature and heating time. The clinical effectiveness of HT 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. Clinically, heating of superficial tumors (e.g., breast, neck nodes and malignant melanoma) has confirmed the biological rationale for using HT as an adjuvant to RT [14]. The thermal enhancement ratios are approximately 1.5 in several superficial tumor sites after external heating. From a practical point of view, true simultaneous treatment is almost impossible using external heating, and the major effect of the combined treatment will have to rely on hyperthermic cytotoxicity. This makes the design of clinical schedules less complicated since only a few heat fractions may be needed to achieve an optimal effect.

We focus our analysis on the application of HT on superficial and semi-deep tumors. To this end, two main parameters are used for the optimisation of the temperature distribution during superficial HT treatments: EM power and water bolus temperature [70]. The water bolus in microwave HT has two main functions: it cools the surface of the body to avoid burns or overheating of the skin and guarantees good contact between the applicator and the body surface, which implies good impedance matching of the applicator. The choice of the correct temperature of the bolus is a critical point not only to guarantee adequate superficial cooling but also to determine the depth of the HT treatment [16]. We decided to decouple the problem and consider the thermal problem, without considering the interaction process between the EM energy deposited from the microwave antenna and the biological tissue, which results in thermal energy deposition.

2.2 Devices and procedures

This section is dedicated to providing an overview of the techniques for the heating of biological tissues, the equipment necessary to conduct HT treatments, and useful metrics to monitor treatments.

2.2.1 Methods of heating

Achievement of an effective treatment requires high-quality heating equipment, precise thermal dosimetry, and adequate quality assurance. Several types of devices, antennas and heating or power delivery systems have been proposed and developed in recent decades. These vary considerably in technique, heating depth, ability to focus, and size of the heating focus [38]. Clinically used heating techniques involve

electromagnetic and ultrasonic heating, hyperthermic perfusion and conductive heating.

Depending on clinical objectives and available technology, thermal therapies can be subdivided into three broad categories: local, locoregional, or whole body heating.

Electromagnetic heating techniques apply a high frequency alternating sinusoidal EM field generated using one or more antennas. These EM fields cause dielectric heating by molecular dipole rotation/polarization/vibration in the [MHz] range and ionic conduction in the [kHz] range. Polar molecules (e.g. water) have an electric dipole moment and therefore these molecules align continuously with the alternating field. Electric forces cause rotating molecules to push, pull, and collide with other molecules, thereby distributing the energy to adjacent molecules and atoms, causing dielectric heating. In conduction, ions in tissue oscillate due to the forces exerted by the electric current. This current faces internal resistance because of the collisions of charged particles with neighboring molecules or atoms, causing dielectric heating. Tissue heating is dominated by ionic conduction in the extracellular fluid for lower frequencies (< 1 [MHz]) with the cell membranes acting as isolators, and by dipole polarization at higher frequencies (> 1 [MHz]). For frequencies > 1 [MHz] the cell membranes become permeable to the E-fields and the microscopic structure of tissues can be neglected [21].

2.2.2 Equipment for superficial hyperthermia

In superficial heating, the energy is deposited in a limited volume of tissue close to the heating device. It is therefore applied only to tumors extending up to $\sim 3\text{--}4$ [cm] below the skin surface.

A water bolus, with water circulating at a temperature typically around 40 [$^{\circ}\text{C}$], is used to couple the electromagnetic energy into the tissue. The selected exact water temperature influences the skin temperature and depends on the desired heating depth and whether the skin is target or not [2].

Significant progress has been made in further development of systems with better penetration depth at 915 and 434 [MHz]. Furthermore, superficial antennas with various effective field sizes allowing adequate treatment of both small and larger tumor areas have been developed. This has resulted in clear progress in the quality of superficial heating equipment [67].

The Yahta-4 system uses contact flexible microstrip applicator (CFMA; SRPC 'Istok', Fryazino, Russia) consisting of two coplanar active electrodes and a shield electrode, separated by a thin fluoroplastic substrate [24]. The applicator has an integrated water bolus and can be bent to follow the curvature of the body contour. Five different sizes are available to cover target regions of various dimensions. The smallest antenna, type 1H, has an aperture size of 7.2×19.7 [cm^2] and the largest, type 3H and 5H, have aperture sizes of 28.7×20.7 [cm^2] and 19.7×28.5 [cm^2], respectively, the difference being a 90°shift in field direction. Bending the applicator increases the effective heating depth [43].

The ALBA ON 4000 (ALBA Hyperthermia, Rome, Italy) is a 434 [MHz] system for superficial HT, which uses antennas similar to the CFMA, but with a fixed curvature. The system has an integrated thermocouple thermometry system for continuous monitoring of 4–64 temperature sensors.

2.2.3 Metrics of interest in HT routine

Some metrics of interest in HT treatments are:

- **T₉₀, T₅₀ or Tx**: the temperatures at which 90%, 50% or x% of the tumor temperature measurements exceed;
- **T_{max}**: the maximum temperature measured during treatment;
- **CEM43T90**: the cumulative equivalent minutes at 43 [°C] thermal dose calculated for the T90-temperature over the course of treatment. The use of CEM43 is based on the Arrhenius relationship, describing the time-temperature dependent cytotoxic effect of HT [15];
- **T_{rise}**: the average temperature increase above 37 [°C], multiplied by the ratio of the prescribed duration to the actual treatment duration [69].

2.3 Thermal modeling

Partial differential problems are usually expressed as a system of three components:

- a general linear or nonlinear differential operator applied to the variable of interest u :

$$\mathcal{N}(x, u, t) = 0 \quad x \in \Omega, t \in [0, T];$$

- an Initial Condition (IC) over the whole computational domain Ω :

$$u(x, 0) = \tilde{d}(x) \quad x \in \Omega;$$

- one or more Boundary Conditions (BCs) on the variable u or on its normal derivative at a boundary $\partial\Omega$. Different kinds of BCs can be formulated:

1st kind, Dirichlet:

$$u(x, t) = \tilde{g}(x, t) \quad x \in \partial\Omega, t \in [0, T];$$

2nd kind, Neumann:

$$\partial_x u|_{\partial\Omega} = \tilde{g}(x, t) \quad x \in \partial\Omega, t \in [0, T];$$

3rd kind, Robin:

$$\partial_x u|_{\partial\Omega} = \tilde{g}(u, x, t) \quad x \in \partial\Omega, t \in [0, T];$$

2.3.1 Pennes' Bio-heat equation

Pennes' Bio-Heat Equation (PBHE) is a parabolic Partial Differential Equation (PDE) consisting of a standard heat equation plus a linear reaction term. The control and estimation of this equation are complicated by the presence of unknown coefficients in both thermal and EM properties of the patient (see 2.3.3). Perfusion is connected to vascular dilatation due to the increase in temperature, it is difficult to assess and can increase when tissues are heated, but it is bounded in a certain range for each type of tissue. Understandably, this increase affects the temperature distribution in tissues, causing cold tracks or a more generalized cooling effect depending on the size and geometry of vessels. The antenna's heating pattern and the water bolus's cooling efficacy are connected to the characteristics of the heating devices, that need an adequate characterization.

According to Pennes [61], heat transfer in a biological system with a homogeneous medium, expressed in cartesian coordinates, can be written as:

$$\varrho c \partial_t T = k \partial_{xx} T - \varrho_b c_b w_b (T - T_a) + \mathcal{Q}, \quad (2.1)$$

defined on the domain $\Omega \in [0, L_0] \times [0, +\infty)$, with L_0 [m] the length of the domain. The term on the left side is the change in thermal energy storage, where ϱ is the density [kg m^{-3}] and c is the specific heat [$\text{J kg}^{-1} \text{K}^{-1}$] of the biological tissue. The first term on the right side is the net transfer of thermal energy due to temperature gradients, and k is the thermal conductivity [$\text{W m}^{-1} \text{K}^{-1}$] of the biological tissue. The second term on the right side is the blood perfusion term, where ϱ_b is the density of blood [kg m^{-3}], c_b is the specific heat of blood [$\text{J kg}^{-1} \text{K}^{-1}$], and w_b is the blood perfusion rate [s^{-1}], i.e., the volumetric blood flow rate per tissue volume [78], which is variable; T_a is the arterial blood temperature [$^\circ\text{C}$], that we assume to be constant and equal to 37 [$^\circ\text{C}$] [56]. The term \mathcal{Q} is the heat source [W m^{-3}]. It comprehends the metabolic heat generation rate q_{met} , considered constant per unit volume and unit time and equal to 4200 [W m^{-3}], and the power \mathcal{Q}_E deposited by an electric field E in tissue with electric conductivity σ [S m^{-1}] [44]:

$$\mathcal{Q} = q_{\text{met}} + \mathcal{Q}_E = q_{\text{met}} + \frac{1}{2} \sigma |E|^2 \quad [\text{W m}^{-3}].$$

For simplicity, given the choice made to decouple the thermal problem from the EM one, the present study will consider Specific Absorption Rate SAR [W kg^{-1}] as heat source: $\mathcal{Q} = \varrho \text{SAR}$ [W m^{-3}].

The original heuristic expression proposed in [29] describes the SAR distribution for radiative applicators in homogeneous, three-dimensional tissues and is based on the hypothesis that absorption is described by a linear term in the exponent in the x-direction and a quadratic term in the z- and y-directions. The formula describes how SAR varies in time with the power delivered by the hyperthermic system corrected for the reflected power, and in space depending on the applicator characteristics:

$$\text{SAR}(x, y, z, t) = \beta \exp[-az^2 - by^2 - \nu(x - x_0)] P(t) \quad [\text{W kg}^{-1}], \quad (2.2)$$

where $P(t)$ [W] is the delivered power corrected for the reflected fraction. The definition of a, b and c is given in Eq. (2.3). They are related to the European

Society for Hyperthermic Oncology (ESHO) parameters Effective Field Size (EFS), which indicates the area of effective heating for a superficial applicator, and to the Effective Heating Depth (EHD) of the applicator, equal to the Penetration Depth [m] (PD) as defined in [42]:

$$a = \frac{4 \ln 2}{EFS_x^2} \quad [\text{m}^{-2}], \quad b = \frac{4 \ln 2}{EFS_y^2} \quad [\text{m}^{-2}], \quad \nu = \frac{\ln 2}{PD - 10^{-2}x_0} \quad [\text{m}^{-1}]. \quad (2.3)$$

β [-] is a free parameter, whose value can be determined by a fitting procedure, and the constant x_0 is set to 0.004 [m].

In this study, the mono-dimensional formulation of Eq. (2.2) will be considered, evaluating the function along the line $(y, z) = (0, 0)$, which represents the ideal needle inserted perpendicular to the patient's surface from the applicator.

Boundary conditions

BCs for PBHE are constraints on temperature and its normal derivative. In the following, the physical meaning of each kind of boundary condition is reported.

To impose a Dirichlet condition (1st kind) means that the system is cooling or heating at a certain temperature of the wall $T_w(t)$ [°C]:

$$T|_{\partial\Omega} = T_w(t).$$

Recalling the law of Fourier, the heat flow Q [W] through a plate is proportional to plate thickness Δx [m], plate surface A [m^2] and temperature difference over the plate ΔT [°C]. The proportionality is made equality by the definition of the thermal conductivity k [$\text{W m}^{-1} \text{K}^{-1}$]

$$Q = -kA \frac{\Delta T}{\Delta x} \quad [\text{W}]. \quad (2.4)$$

If we consider the heat flow per unit surface q_w [W m^{-2}] (also defined as heat flux) for a plate of infinitesimal thickness dx , Eq. (2.4) transforms to:

$$q_w = -k \partial_x T. \quad (2.5)$$

From Eq. (2.5) it is clear that imposing a Neumann condition (2nd kind) is equivalent to cool or heat by a certain $q_w(t)$:

$$-k \partial_x T \Big|_{\partial\Omega} = q_w(t).$$

A Robin BC (3rd kind) represents heating or cooling by convective heat transfer, which means heat exchange between the domain boundary and fluid at a certain temperature of the fluid $T_f(t)$ [°C]:

$$-k \partial_x T \Big|_{\partial\Omega} = h (T_f(t) - T|_{\partial\Omega}),$$

where h [$\text{W m}^{-2} \text{K}^{-1}$] is the convection coefficient between the domain boundary and the fluid.

Steady-state solutions

PBHE (2.1) includes the blood perfusion term, whose function is to maintain the steady-state temperature of the biological tissue T_B [°C]:

$$T_B = T_a + \frac{q_{\text{met}}}{\varrho_b c_b w_b}.$$

Therefore, Eq. (2.1) has steady-state solutions. The characteristic length of bioheat transfer δ_B [m] is defined as

$$\delta_B = \sqrt{\frac{k}{\varrho_b c_b w_b}}.$$

In the case of the 1-D Cartesian coordinate, the expression for the steady-state solutions depends on the kind of BC:

$$\begin{aligned} \text{1st kind: } T_{\text{steady}}(x) &= T_B + (T_w - T_B)e^{-\frac{x}{\delta_B}}; \\ \text{2nd kind: } T_{\text{steady}}(x) &= T_B + \frac{q_w \delta_B}{k} e^{-\frac{x}{\delta_B}}; \\ \text{3rd kind: } T_{\text{steady}}(x) &= T_B + \frac{T_f - T_B}{1 + \frac{k}{h \delta_B}} e^{-\frac{x}{\delta_B}}. \end{aligned} \quad (2.6)$$

Problem formulation

Let us consider the monodimensional formulation of Eq. (2.1), the domain being $\Omega = [0, L_0] \times [0, +\infty)$. We consider an ideal needle that penetrates perpendicularly from the surface of the patient, governed by the following PDE:

$$\varrho c \partial_t T = k \partial_{xx} T - \varrho_b c_b w_b (T - T_a) + \varrho \text{SAR}.$$

The IC is:

$$T(x, 0) = T_0(x) \quad x \in [0, L_0].$$

For the superficial end ($x = 0$), a Robin-type BC is chosen to model the convective heat transfer between fluid and the surface of the patient:

$$\partial_x T|_{x=0} = -\frac{h}{k} (v(t) - T(0, t)) \quad t \geq 0,$$

The temperature of the fluid (v) is a control input and it is adjusted together with the deposited power (SAR) to obtain a peak temperature inside the domain, where the target is localized. For the in-depth end ($x = L_0$), with L_0 chosen sufficiently large to be outside of the heating range of the applicator, a Dirichlet BC is imposed:

$$T(L_0, t) = T_a \quad t \geq 0.$$

The following measurements are available:

$$\tilde{y}_1(t) = T(L_0, t), \quad \tilde{y}_2(t) = T(0, t), \quad \tilde{y}_3(t) = v(t).$$

As depicted in Fig. 2.1, these measurements constitute the in-depth temperature, the superficial temperature, and the temperature of the water bolus. In clinical scenarios, these are obtained through the use of TC. While y_2 and y_3 are non-invasive measurements, acquiring y_1 requires the use of a catheter through which the TC is inserted, causing patient discomfort.

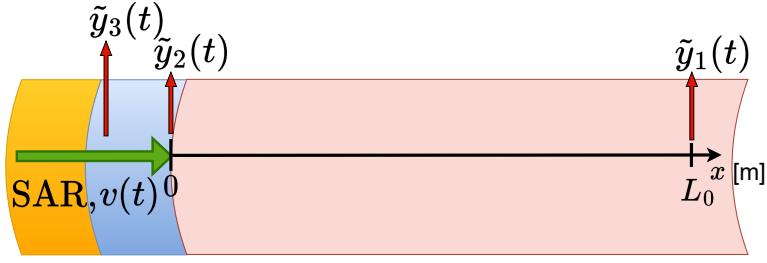


Figure 2.1. A generic scheme for the mono-dimensional PBHE: the domain represents a virtual needle of length L_0 that enters the patient orthogonally from the surface in which heat SAR is applied ($x = 0$). Skin burns are prevented by controlling the water bolus temperature v . The boundary measurements of the system \tilde{y}_3 , \tilde{y}_2 , and \tilde{y}_1 are available.

Scaling

To achieve the robustness and stability of Physics-Informed Neural Networks (PINNs) that will be employed to solve the PBHE, it is recommended to scale input and outputs to $\mathcal{O}(1)$. In the following, we report the scaling procedure, inspired by [56]. First, we introduce a temperature translation $T' = T - T_a$:

$$\varrho c \partial_t T' = k \partial_{xx} T' - \varrho_b c_b w_b T' + \mathcal{Q}.$$

Then, we define the dimensionless temperature $\theta = \frac{T'}{\Delta T}$, with $\Delta T = T_{\max} - T_{\min}$. T_{\max} can be set to 45 [°C], i.e., the maximum temperature reached in HT treatments. Substituting $T' = \theta \cdot \Delta T$:

$$\varrho c \partial_t \theta = k \partial_{xx} \theta - \varrho_b c_b w_b \theta + \frac{\mathcal{Q}}{\Delta T}. \quad (2.7)$$

Choosing a time span t^* [s], time and space coordinates are scaled:

$$\tau = \frac{t}{t^*} [-], \quad X = \frac{x}{L_0} [-].$$

Partial derivatives are transformed as follows:

$$\partial_x \theta = \frac{1}{L_0} \partial_X \theta, \quad \partial_{xx} \theta = \frac{1}{L_0^2} \partial_{XX} \theta, \quad \partial_t \theta = \frac{1}{t^*} \partial_\tau \theta.$$

Substituting inside Eq. (2.7):

$$\frac{\varrho c}{t^*} \partial_\tau \theta = \frac{k}{L_0^2} \partial_{XX} \theta - \varrho_b c_b w_b \theta + \frac{\mathcal{Q}}{\Delta T},$$

defined for $(X, \tau) \in [0, 1] \times [0, +\infty)$. Introducing the scaled boundary measurements $y_i = \tilde{y}_i / \Delta T$, $i = 1, 2, 3$, and the scaling coefficients:

$$\begin{aligned} a_1 &= \frac{L_0^2}{k} \frac{\varrho c}{t^*}, & a_2 &= L_0^2 \frac{\varrho_b c_b}{k}, & a_3 &= \frac{\varrho L_0^2}{k \Delta T} \beta \exp(\nu z_0), \\ a_4 &= \nu L_0, & a_5 &= \frac{L_0 h}{k}, \end{aligned} \quad (2.8)$$

the problem formulation read as:

$$a_1 \partial_\tau \theta = \partial_X X \theta - w_b a_2 \theta + \bar{P}(\tau) a_3 \exp(-a_4 X), \quad (2.9)$$

where $\bar{P}(\tau)$ is the net delivered power scaled in time. Introducing the scaled control input $\bar{v}(\tau)$, the BCs are transformed as:

$$\begin{aligned} \partial_X \theta|_{X=0} &= -a_5(\bar{v}(\tau) - \theta(0, \tau)), \quad \tau \geq 0, \\ \theta(1, \tau) &= 0, \quad \tau \geq 0. \end{aligned}$$

A possible IC for the system is represented by the polynomial

$$\theta_0(X) = b_1 X^3 + b_2 X^2 + b_3 X + b_4 \quad X \in [0, 1]. \quad (2.10)$$

The constants b_1 , b_2 , b_3 and b_4 are defined following the compatibility with BCs and imposing the peak temperature θ_{gt} at X_{gt} :

$$\begin{aligned} \theta_0(1) &= y_1(0), \\ \theta_0(0) &= y_2(0), \\ \partial_X \theta_0(0) &= -a_5(\bar{v}(0) - \theta(0, 0)), \\ \theta_0(X_{\text{gt}}) &= \theta_{\text{gt}}. \end{aligned}$$

Substituting inside Eq. (2.10):

$$\begin{aligned} b_1 + b_2 + b_3 + b_4 &= y_1(0), \\ b_4 &= y_2(0), \\ b_3 &= -a_5(y_3(0) - y_2(0)), \\ b_1 X_{\text{gt}}^3 + b_2 X_{\text{gt}}^2 + b_3 X_{\text{gt}} + b_4 &= \theta_{\text{gt}}. \end{aligned} \quad (2.11)$$

The system of linear equations in Eq. (2.11) can be solved using any linear solver.

Recalling the definition of the characteristic length of bioheat transfer δ_B [m] in Eq. (2.3.1), Fourier and Biot numbers can be defined, respectively, as

$$Fo = \frac{k}{\rho c} \frac{t}{\delta_B^2}, \quad Bi = \frac{h \delta_B}{k}.$$

Substituting a_1 and a_5 from Eq. (2.8):

$$Fo = \frac{\tau}{a_1} \frac{L_0^2}{\delta_B^2}, \quad Bi = a_5 \frac{\delta_B}{L_0}.$$

The Fourier number is the dimensionless characteristic time for the occurrence of a diffusion transient over the characteristic length scale δ_B . It characterizes the rate of heat conduction within a solid object or material over time. It helps understand how quickly temperature changes propagate through a material when subjected to varying thermal conditions.

The Biot number describes relative rates of heat transfer inside a solid medium compared to heat transfer at the solid's external surface. It gives insight into whether the heat conduction within a solid material can be considered negligible in relation to the heat convection transfer at its surface or if the two processes must be analyzed together.

2.3.2 Effect of discrete blood vessels

Heat transfer between solid media and flow in tubes (vessels) has great clinical relevance. Depending on the size of the heated volume, it is possible to describe the heat transfer contribution of all small vessels by an enhanced effective thermal conductivity (expected values 5-10 times the intrinsic thermal conductivity of the tissue), while we have to describe the large discrete vessels separately.

Definition 1 (Thermal Equilibrium Length) *The length over which the temperature difference between the blood and the tissue will be reduced by a factor e.*

The basic assumption of PBHE is that blood enters the local tissue volume at the arterial temperature T_a (37 [°C]) and leaves this volume at the local tissue temperature. The main objection against this conventional bioheat transfer theory is that it neglects the following aspects:

1. the heat transport related to the mass transport of blood;
2. the actual temperature of the blood entering the local tissue volume;
3. the individual cooling of discrete large vessels;
4. the fundamental importance of the entire venous vessel network by assuming an infinite thermal equilibrium length for all venous vessels.

The reason for this conventional theory remaining the basis of most thermal modelling lies in its mathematical simplicity and its intelligent form. The combination of the heat production term and the heat sink term, which are both spatially variable, together with the fact that the solid tissue heat transfer contribution by conduction is relatively unimportant, makes possible the fit of almost every measured temperature distribution. However, this is just a mathematical fit; it fails in its description of heat transfer processes in vascularized tissues.

Summarizing, the bottlenecks in bioheat transfer are: the thermal equilibrium length of venous vessels in comparison with their actual physiological length; the enhancement of the effective thermal conductivity; the presence of large vessels.

Effective thermal conductivity

In the literature, fat and muscle conductivity are usually given as 0.2 and 0.5 – 0.6 [$\text{W m}^{-1} \text{K}^{-1}$] respectively, but these values apply to excised non-perfused tissue. The blood flow in the smaller (counter-current) vessels, diameters 0.1 – 0.5 [mm], contributes greatly to the effective tissue conductivity. In general, effective thermal conductivity k_{eff} [$\text{W m}^{-1} \text{K}^{-1}$], is a tensor, dependent on factors such as the velocity of the flow v [cm s^{-1}], direction and vascular density [10]. For a high vascular density, k_{eff} can be considered uniform locally; for isotropic flow k_{eff} reduces to a scalar. For example, $k_{\text{eff}} = 2$ [$\text{W m}^{-1} \text{K}^{-1}$] in the direction of a 300 [μm] diameter counter-current pair (one pair per [cm^2]), against 0.5 – 0.6 [$\text{W m}^{-1} \text{K}^{-1}$] without blood flow [74]. Similarly, numerical simulations of arrays of discrete vessels with a diameter of 0.48 [mm], incorporated in blocks of tissue (one vessel per 2.25 [mm^2])

showed a k_{eff} of $7.2 \text{ [W m}^{-1} \text{ K}^{-1}]$ [11]. For the simple one-dimensional case where two blood vessels with radius $a \text{ [m]}$ and spacing $l \text{ [m]}$ have the same direction as the temperature gradient, the effective thermal conductivity is defined as follows:

$$k_{\text{eff}} = k \left(1 + \frac{\pi^2}{4S k^2} n a^2 k_b^2 Pe^2 \right),$$

Where $S = \frac{\pi}{\cosh^{-1}\left(\frac{l}{2a}\right)}$ [-] is a shape factor, $k_b \text{ [W m}^{-1} \text{ K}^{-1}]$ is the thermal conductivity of blood due to conduction, $Pe = PrRe = 2\varrho_b c_b \frac{av}{k_b}$ [-] is the Peclet number.

Heating of discrete blood vessels

In the following, we report a solution given in [41, Section 3.2] and some considerations for further developments. The model consists of a cylinder of tissue of radius $R_2 \text{ [m]}$ around a vessel of radius $R_1 \text{ [m]}$ and length $L \text{ [m]}$. A constant higher temperature $T_{R2} \text{ [}^\circ\text{C]}$ is maintained on the outer surface of the tissue cylinder.

The temperature distribution in the tissue and the temperature of the fluid are calculated as a function of the position $y \in [-L/2, L/2] \text{ [m]}$ along the vessel. We both have to consider the heating of the fluid in the vessel and the temperature distribution in the surrounding tissue.

First, we introduce the mean fluid temperature $\langle T \rangle \text{ [}^\circ\text{C]}$. This mean fluid temperature is defined such that $\langle T \rangle$ is proportional to the amount of heat transported by the fluid:

$$\langle T \rangle \cdot \langle v \rangle \cdot \pi R_1^2 = \int_0^{R_1} 2\pi r v(r) T(r) dr,$$

where $\langle v \rangle$ is the mean flow velocity and $v(r)$ the flow velocity at radius r . The flow of the fluid through the vessel is assumed to be laminar. The heat flux through the vessel wall at a point x is given by:

$$Q'' = h(y)(T_{\text{wall}}(y) - \langle T_{\text{fluid}}(y) \rangle) \quad [\text{W m}^{-2}]. \quad (2.12)$$

Because there is no discontinuity in the temperature distribution along the vessel, the heat transfer coefficient $h(y) \text{ [W m}^{-2} \text{ K}^{-1}]$ is given by the following equation, with $D = 2R_1$:

$$h = 3.66 \frac{k}{D} \quad [\text{W m}^{-2}].$$

Assuming that there is no axial heat flow through the domain and that there is no absorption of energy in the domain around the vessel, we can calculate the heat flow from the medium towards the vessel. The vessel is divided along its axis into small elements $\Delta y \text{ [m]}$, which are taken so small that the heat flux is constant over it. If we now consider the heat flow through the vessel wall at a point y over a length Δy , using equation (2.12), we find:

$$Q(y) = 2\pi R_1 h \Delta y (T_{\text{wall}}(y) - \langle T_{\text{fluid}}(y) \rangle) \quad [\text{W}]. \quad (2.13)$$

The heat flow through the tissue over the same length is given considering heat conduction through the wall of a cylindrical tube:

$$Q(y) = \frac{2\pi k \Delta y}{\ln\left(\frac{R_2}{R_1}\right)} (T_{R2} - T_{\text{wall}}(y)) \quad [\text{W}]. \quad (2.14)$$

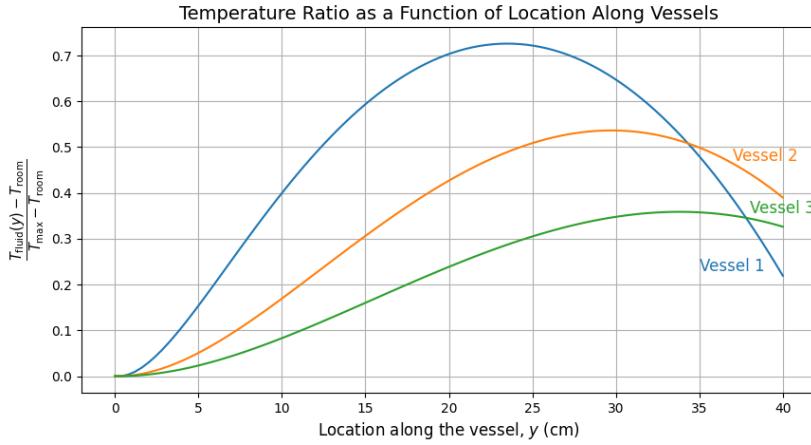


Figure 2.2. The temperature distribution along the axis for some vessels:

Vessel 1: $\langle v \rangle = 1.0 \text{ [cm s}^{-1}\text{]}, R_1 = 0.5 \text{ [mm]}$

Vessel 2: $\langle v \rangle = 1.5 \text{ [cm s}^{-1}\text{]}, R_1 = 1.0 \text{ [mm]}$

Vessel 3: $\langle v \rangle = 2.0 \text{ [cm s}^{-1}\text{]}, R_1 = 1.5 \text{ [mm]}.$

The heat flow through the vessel wall will heat the fluid. This correlation is quantified at a point x by the formula:

$$Q(y) = \pi R^2 \rho \langle v \rangle c (\langle T_{\text{fluid}}(y + \Delta y) \rangle - \langle T_{\text{fluid}}(y) \rangle) \quad [\text{W}]. \quad (2.15)$$

The heat flows expression in Eq. (2.13), (2.14) and (2.15) must be equal in the stationary situation, so if both $\langle T_{\text{fluid}}(y) \rangle$ and T_{R2} are known, $\langle T_{\text{fluid}}(y + \Delta y) \rangle$ and $T_{\text{wall}}(y)$ can be calculated. The temperature of the fluid at the vessel entry point $T_{\text{inlet}} = \langle T_{\text{fluid}}(-L/2) \rangle$ is known. As this entry temperature is known we can calculate the heating of the fluid and the temperature distribution around the vessel by a step-by-step calculation. The temperature distribution along the axis for some vessels is given in Figure 2.2. Our efforts went to model the case in which the temperature T_{R2} is not constant. Two approaches have been lined out, the first being the introduction of a fitting parameter $\epsilon \in (0, 1) [-]$ in Eq. (2.14):

$$Q(y) = \epsilon \cdot \frac{2\pi k \Delta y}{\ln(\frac{R_2}{R_1})} (T_{R2} - T_{\text{wall}}(y)) \quad [\text{W}].$$

Under the assumption that the temperature at the outlet T_{outlet} is known, the parameter ϵ can be tuned to obtain $\langle T_{\text{fluid}}(L/2) \rangle = T_{\text{outlet}}$.

The other approach may be to consider a profile for $T_{R2}(y)$, $y \in [-L/2, L/2]$. This may take the form of a straightforward parabola, peaking at $y = 0$, where $T_{R2}^0 = T_{\min}$:

$$T_{R2}(y) = T_{R2}^0 + \frac{4(T_{\min} - T_{R2}^0)}{L^2} \quad [\text{°C}],$$

or the steady-state solution to PBHE in Eq. (2.6).

2.3.3 Tissue property and perfusion uncertainties in HT

In Ref. [28], a Polynomial Chaos Expansion (PCE)-based HTP uncertainty quantification was developed and applied for locoregional HT. Uncertainties in thermal

conductivity and heat capacity were found to have negligible impact on the predicted temperature ($< 1.0e-10$ [°C]), while density and permittivity uncertainties had a small impact (< 0.3 [°C]). Uncertainties in electrical conductivity and perfusion can lead to large variations in predicted temperature. However, variations in muscle properties result in the largest impact at locations that could limit treatment quality, with a standard deviation up to almost 6 [°C], and 3.5 [°C], for perfusion and electrical conductivity, respectively.

The combined influence of all significant uncertainties leads to large variations with a standard deviation up to 9.0, 3.6, 3.7 and 4.1 [°C], for the pancreatic, prostate, rectal, and cervical cases, respectively.

2.4 Current limitations of Hyperthermia treatment

Currently, HT is not frequently used by radiation oncologists despite its proven efficacy benefits when combined with CH and RT. This limited adoption stems primarily from its non-optimal integration into the radiation therapy workflow.

The few patients who do receive HT treatments experience significant drawbacks, including long preparation times, extended immobilization during treatment sessions, and limited accuracy in treatment delivery. Conventional HT treatment is time-consuming and lacks quantitative, online assessment of accuracy, making reproducibility difficult and often leading to unsatisfactory outcomes.

Temperature monitoring presents particular challenges in clinical settings. Currently, temperature sensors can be placed on the skin surface to prevent overheating, but accurate in-depth temperature measurement requires invasive catheters with temperature probes running through the patient's tissue. While these provide more precise measurements along the catheter path, they cause considerable discomfort. Consequently, in-depth temperature monitoring is often avoided, compromising treatment optimization.

The consequence of these limitations is that patients undergoing RT and/or CH remain subjected to suboptimal outcomes and survival rates. A solution that provides non-invasive in-depth temperature information by solving the PBHE (see 2.3.1) with boundary conditions informed by superficial measurements would be highly beneficial for real-time treatment guidance.

2.5 Potential of Hyperthermia in cancer treatment

Each year, approximately 2.5 million people are diagnosed with breast cancer and head-and-neck cancer, requiring treatment with combinations of surgery, CH, and/or RT. Despite current treatments, 10-year recurrence rates remain high at approximately 30%.

These patients experience enormous physical burdens from the side effects associated with CH/RT (e.g., fatigue, hair loss, nausea), as well as mental distress related to uncertainty about recurrence and survival. This suffering extends to their loved ones and caregivers who indirectly share their burden.

For these patients, the use of HT remains very limited despite evidence showing a 1.5-factor improvement in complete response rates [14]. The wide adoption of

HT, especially in low- and middle-income countries, could be a game changer in the management of cancer therapy in the coming decades.

Recent technological advances show promise for addressing current limitations. An Artificial Intelligence (AI)-based system could provide better insights into temperature distribution, significantly improving treatment precision. Additionally, accuracy in pre-operative plan transfer and antenna positioning can be dramatically improved with robotic assistance (ROBHOT, [19]). With accurate knowledge of the temperature distribution in tissue during superficial HT treatments, clinicians could further optimize treatment outcomes while minimizing patient discomfort.

Chapter 3

Machine Learning

A common challenge in physics research is developing computational methods to solve problems that are intractable using classical algorithms. Recent advances in Deep Learning (DL) have significantly accelerated progress in this area through the application of deep neural networks (NNs).

At their core, NNs apply a sequence of operations to input data to generate predictions. These operations are structured into layers, each containing multiple interconnected neurons (or nodes). The connections between neurons are characterized by weights, while each neuron is associated with a bias term. During the learning process, the network iteratively adjusts these parameters to minimize prediction error.

DL has demonstrated considerable potential in solving PDEs through state estimation using NNs, providing a powerful alternative to traditional numerical methods. Numerous approaches have emerged in recent years, each proposing different core architectures for the underlying NNs. Although traditional DL models typically require large datasets to achieve accurate results, physics-informed learning presents a different paradigm. In this approach, prior knowledge of physical laws acts as an implicit regularization mechanism, reducing or even eliminating the need for extensive labeled training data. This capability is particularly valuable in engineering and scientific applications where physical laws are well-established but real-world data are scarce or difficult to obtain. PINNs are especially promising for inverse problems, where key parameters governing a system must be inferred rather than observed directly.

In healthcare, DL is already transforming clinical practice, aiding in medical image segmentation, quantitative analysis, and automated reporting. The exploration of AI-driven methods continues to open new frontiers for engineering applications, bridging data-driven and physics-based modelling approaches.

This chapter is structured as follows: we begin with an introduction to ML, followed by a review of fundamental mathematical tools, including the gradient descent optimization algorithm. We then provide an overview of the core NN architectures and their key parameters. Next, we explore ML applications in healthcare before introducing PINNs. The final section presents a set of techniques for optimizing PINN training to improve performance and stability.

3.1 Introduction

ML is a field of AI that enables computational systems to learn from experience and improve their performance on a given task without being explicitly programmed. The origins of ML go back to the pioneering work of Arthur Samuel (1901–1990), who defined it as the

“programming of a digital computer to behave in a way which, if done by human beings or animals, would be described as involving the process of learning.”

More formally, Tom Mitchell provided a widely accepted definition:

“Machine Learning is the study of computer algorithms that improve automatically through experience.”

Mathematically, ML consists of developing models and algorithms capable of describing or replicating the behaviour of a system based on observed data. A learning model should be expressive enough to capture essential aspects of the problem while maintaining mathematical tractability.

As Mitchell further elaborated, a system is said to learn with respect to a task T , a performance metric P , and an experience E if its performance at T improves, as measured by P , following experience E .

The two primary tasks being performed by ML are:

- Classification: The goal is to determine the class to which a given element belongs. This is particularly relevant in applications such as image recognition, spam detection, and medical diagnosis.
- Regression: The objective is to estimate the value of a certain quantity based on input features. Regression tasks arise in areas such as financial forecasting, temperature prediction, and risk assessment.

The process of learning in ML consists of two key phases, the first being the training, and the second the prediction phase. In the training phase, the model learns from a set of available data by optimizing an objective function. This phase typically involves the application of optimization algorithms to minimize an error metric or maximize predictive accuracy. In the prediction phase, the trained model is used to make predictions on new, unseen instances. The ability to generalize well to new data is critical in determining the success of an ML model.

Given a collection of known instances, referred to as the training set, each containing a set of attributes (input variables), the objective is to construct a predictive model that maps inputs to outputs. The training set is used to identify patterns and underlying relationships, which the model then utilizes for making predictions.

To assess the quality and robustness of a learning model, a separate subset of data, known as the test set, is employed. The test set is not used during training and serves as a proxy for evaluating how well the model generalizes to future data. This validation technique is referred to as cross-validation.

3.2 Mathematical preliminaries

The present section delves deeper into the theoretical foundations, methodologies, and applications of ML, providing some tools for a better understanding of its role in solving complex problems.

3.2.1 Gradients and Jacobians

We give here some basic definitions and concepts concerning the differentiation of multi-valued functions. Recalling Section 1.4, we denote by $\partial_{z_i} f$ the partial derivative of $f(z_1, \dots, z_n)$ with respect to the variable z_i , $i = 1, \dots, n$. Some important properties of derivatives are:

- **Linearity:** the derivative of a sum is the sum of derivatives:

$$\partial[f(x) + g(x)] = f'(x) + g'(x);$$

- **Product rule:**

$$\partial[f(x)g(x)] = f'(x)g(x) + f(x)g'(x);$$

- **Chain rule:** the derivative of function composition is given by multiplying the corresponding derivatives:

$$\partial[f(g(x))] = f'(g(x))g'(x);$$

Definition 2 (Gradient) *the gradient of a function $y = f(\mathbf{x})$ is given by:*

$$\nabla f(\mathbf{x}) = \partial f(\mathbf{x}) = \begin{bmatrix} \partial_{x_1} f(\mathbf{x}) \\ \vdots \\ \partial_{x_d} f(\mathbf{x}) \end{bmatrix}.$$

We can introduce then the **directional derivative** when considering displacements in a general direction \mathbf{v} :

$$D_{\mathbf{v}} f(\mathbf{x}) = \lim_{h \rightarrow 0} \frac{f(\mathbf{x} + h\mathbf{v}) - f(\mathbf{x})}{h}.$$

We can decompose the vector \mathbf{v} in its components along each axis and express directional derivative by the dot product of the gradient with the displacement vector \mathbf{v} :

$$D_{\mathbf{v}} f(\mathbf{x}) = \langle \nabla f(\mathbf{x}), \mathbf{v} \rangle = \sum_i \partial_{x_i} f(\mathbf{x}) v_i.$$

Definition 3 (Jacobian) *The Jacobian matrix of a function $\mathbf{y} = f(\mathbf{x})$, $\mathbf{x} \in \mathbb{R}^d$, $\mathbf{y} \in \mathbb{R}^o$ is given by:*

$$\partial f(\mathbf{x}) = \begin{bmatrix} \frac{\partial y_1}{\partial x_1} & \frac{\partial y_1}{\partial x_2} & \cdots & \frac{\partial y_1}{\partial x_d} \\ \vdots & \ddots & \ddots & \vdots \\ \frac{\partial y_o}{\partial x_1} & \cdots & \cdots & \frac{\partial y_o}{\partial x_d} \end{bmatrix} \in \mathbb{R}^{o \times d}.$$

Gradients and Jacobians can be understood as linear functions tangent to a specific point. In particular, the gradient is the best “first-order approximation” in the following sense. For a point \mathbf{x}_0 , the best linear approximation in an infinitesimal neighborhood of $f(\mathbf{x}_0)$ is given by the Taylor’s theorem:

$$\tilde{f}(\mathbf{x}) = f(\mathbf{x}_0) + \langle \partial f(\mathbf{x}_0), \mathbf{x} - \mathbf{x}_0 \rangle.$$

3.2.2 Gradient descent

Consider the problem of iteratively minimizing a generic function $f(\mathbf{x})$. We can define the descent direction d^k : A direction d^k is said to be descent in \mathbf{x}^k whenever a $\tilde{\eta}^k$ exists such that

$$f(\mathbf{x}^k + \eta d^k) < f(\mathbf{x}^k) \quad \forall \eta \in (0, \tilde{\eta}^k).$$

For small η , hence locally, it is possible to decrease the objective function moving from \mathbf{x}^k along the direction d^k . Whenever it is possible to find such a direction, the point \mathbf{x}^k is not a local minimizer; indeed, there exists a point in the neighbourhood which improves the objective function.

Assume $\nabla f(\mathbf{x}^k) \neq 0$ then a descent direction exists in \mathbf{x}^k and any d such that

$$\nabla f(\mathbf{x}^k)^T d < 0.$$

In particular,

$$d^k = -\nabla f(\mathbf{x}^k),$$

is the steepest descent direction. Following this observation, and with the help of Ref. [65], we can recover the gradient method.

Definition 4 (Steepest gradient descent) *Given a differentiable function $f(\mathbf{x})$, a starting point \mathbf{x}_0 , and a step size sequence η , gradient descent proceeds as*

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \eta^{k+1} \nabla f(\mathbf{x}^k).$$

Some clarifications about the “minimizer” of a function are given here.

Definition 5 (Minimum) *A local minimum of $f(\mathbf{x})$ is a point \mathbf{x}^+ such that the following is true for some $\epsilon > 0$:*

$$f(\mathbf{x}) \leq f(\mathbf{x}') \quad \forall \mathbf{x}' : \quad \|\mathbf{x} - \mathbf{x}'\| < \epsilon.$$

The value of $f(\mathbf{x})$ is a minimum if we consider a sufficiently small neighborhood of \mathbf{x} . Intuitively, in such a point the slope of the tangent will be 0, and the gradient everywhere else in the neighborhood of \mathbf{x} will point upwards. We can formalize the first idea with the concept of stationary points:

Definition 6 (Stationary points) *A stationary point of $f(\mathbf{x})$ is a point \mathbf{x}^+ such that $\nabla f(\mathbf{x}^+) = 0$.*

Stationary points are not limited to minima: they can be maxima (the minima of $-f(\mathbf{x})$) or saddle points, which are inflection points where the curvature of the function is changing. In general, without any constraint on f , gradient descent can only be proven to converge to a generic stationary point depending on its initialization.

Definition 7 (Global minimum) *A global minimum of $f(\mathbf{x})$ is a point \mathbf{x}^* such that $f(\mathbf{x}^*) \leq f(\mathbf{x})$ for any possible input $f(\mathbf{x})$.*

The convergence of gradient descent to the global minimum will be achievable on convex functions.

Definition 8 (Convex function) A function $f(\mathbf{x})$ is convex if for any two points \mathbf{x}_1 and \mathbf{x}_2 and $\alpha \in [0, 1]$ we have:

$$f(\alpha\mathbf{x}_1 + (1 - \alpha)\mathbf{x}_2) \leq \alpha f(\mathbf{x}_1) + (1 - \alpha)f(\mathbf{x}_2). \quad (3.1)$$

The left-hand side in Eq. (3.1) is the value of f on any point inside the interval ranging from \mathbf{x}_1 to \mathbf{x}_2 , while the right-hand side is the corresponding value on a line connecting $f(\mathbf{x}_1)$ and $f(\mathbf{x}_2)$. If the function is always below the line joining any two points, it is convex (as an example, a parabola pointing upwards is convex). For a generic non-convex function, gradient descent converges to a stationary point. Nothing more can be said unless we look at higher-order derivatives (derivatives of the derivatives).

Accelerating gradient descent

The negative gradient describes the direction of the steepest descent, but only in an infinitesimally small neighborhood of the point. These directions can be extremely noisy, especially when dealing with large models. A variety of techniques have been developed to accelerate the convergence of the optimization algorithm by selecting better descent directions. With the “momentum” technique [77, Chapter 12], this can be done without requiring higher-order derivatives.

3.3 Datasets and losses

Definition 9 (Dataset) A supervised dataset \mathcal{T}_n of size n is a set of n pairs $\mathcal{T}_n = \{(x_i, y_i)\}_{i=1}^n$, where each (x_i, y_i) is an example of an input-output relationship we want to model.

Definition 10 (Loss Function) Given a desired target y and the predicted value $y_{\text{NN}} = f(x)$ form a model f , a loss function $\mathcal{L}(y, y_{\text{NN}}) \in \mathbb{R}$ is a scalar, differentiable function whose value correlates with the performance of the model, i.e., $\mathcal{L}(y, y_{\text{NN}}^1) < \mathcal{L}(y, y_{\text{NN}}^2)$ means that the prediction y_{NN}^1 is better than the prediction y_{NN}^2 when considering the reference value (target) y .

We will see shortly that our models can always be parameterized by a set of tensors Υ , called parameters of the model. Minimization of the loss function is done by searching for the optimal value of these parameters via numerical optimization. Hence, given a dataset \mathcal{T}_n , a loss function \mathcal{L} , and a model space f , we can train our model via gradient descent. For example, given a dataset $\{(x_i, y_i)\}_{i=1}^n$ for regression, we can minimize the Least Squares (LS) loss:

$$\Upsilon^* = \arg \min_{\Upsilon} \frac{1}{n} \sum_{i=1}^n (y_i - f(x_i))^2,$$

while for classification, we can wrap the output in a sigmoid function and minimize a cross-entropy loss:

$$L = -\frac{1}{n} \sum_{i=1}^n [y_i \log f(x_i) + (1 - y_i) \log(1 - f(x_i))].$$

3.4 Linear models

Building a model is done by choosing the correct sequence of differentiable blocks. In this section, we introduce the simplest block, linear models, which assume that inputs act additively on the output via a weighted average.

Definition 11 (Linear models) A linear model on an input \mathbf{x} is defined as:

$$f(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + b,$$

where $\mathbf{w} \in \mathbb{R}^c$ and $b \in \mathbb{R}$ are trainable parameters.

Definition 12 (Least-squares) The least-squares optimization problem is given by:

$$\mathbf{w}^*, b^* = \arg \min_{\mathbf{w}, b} \frac{1}{n} \sum_{i=1}^n (y_i - \mathbf{w}^T \mathbf{x}_i - b)^2.$$

Linear models are limited by their assumption of a linear relationship between inputs and outputs, which prevents them from capturing complex patterns in data. They also struggle with modeling interactions between features unless explicitly engineered.

3.5 Feedforward neural networks

To overcome the limitations of linear models, we introduce Feedforward Neural Networks (FNNs), also known as Multilayer Perceptrons (MLPs). These networks extend linear models by incorporating hidden layers and applying linear and nonlinear transformations to the inputs recursively, allowing them to learn intricate, nonlinear relationships automatically. This makes FNNs much more powerful for tasks involving complex data, such as image or speech recognition, where linear models would perform poorly.

Definition 13 (Fully-connected layer) For a batch of n vectors, each of size c , represented as a matrix $\mathbf{X} \in \mathbb{R}^{n \times c}$, a fully connected layer \mathcal{M} is defined as:

$$\mathcal{M} = \Phi(\mathbf{X}\mathbf{W} + \mathbf{b})$$

The parameters of the layer are the matrix $\mathbf{W} \in \mathbb{R}^{c \times c'}$ and the bias vector $\mathbf{b} \in \mathbb{R}^{c'}$, for a total of $(c' + 1)c$ parameters (assuming Φ does not have parameters).

Let $\mathcal{M}^L(x) : \mathbb{R}^{d_{\text{in}}} \rightarrow \mathbb{R}^{d_{\text{out}}}$ be an L -layer NN, or an $L - 1$ -hidden layer NN, with N_l neurons in the l th layer ($N_0 = d_{\text{in}}$, $N_L = d_{\text{out}}$). Let us denote the weight matrix and bias vector in the l th layer by $\mathbf{W}^l \in \mathbb{R}^{N_l \times N_{l-1}}$ and $\mathbf{b}^l \in \mathbb{R}^{N_l}$. We define $\Upsilon = \{\mathbf{W}^l, \mathbf{b}^l\}_{1 \leq l \leq L}$ as the set of all weights matrices and bias vectors in the NN. Given a nonlinear activation function Φ , which is applied elementwise, the MLP is recursively defined as follows:

$$\text{Input layer: } \mathcal{M}^0(\mathbf{x}) = \mathbf{x} \in \mathbb{R}^{d_{\text{in}}}, \quad (3.2)$$

$$\text{Hidden layers: } \mathcal{M}^\ell(\mathbf{x}) = \Phi(\mathbf{W}^\ell \mathcal{M}^{\ell-1}(\mathbf{x}) + \mathbf{b}^\ell) \in \mathbb{R}^{N_\ell}, \quad 1 \leq \ell \leq N, \quad (3.3)$$

$$\text{Output layer: } \mathcal{M}^L(\mathbf{x}) = \mathbf{W}^L \mathcal{M}^{L-1}(\mathbf{x}) + \mathbf{b}^L \in \mathbb{R}^{d_{\text{out}}}. \quad (3.4)$$

The introduction of nonlinear activations prevents the network from collapsing into a linear transformation. If we were to stack multiple linear layers without any nonlinearity, the entire model would still behave as a single linear transformation due to the composition of linear functions. By interleaving linear transformations with nonlinear functions, NNs gain the ability to approximate complex, highly nonlinear functions, making them suitable for tasks such as image recognition, natural language processing, and time-series prediction.

The Universal Approximation Theorem, as formulated by Cybenko [13], states that a feedforward NN with a single hidden layer containing a finite number of neurons and a non-linear activation function can approximate any continuous function on a compact subset of \mathbb{R}^n to any desired degree of accuracy.

Theorem 1 (Universal Approximation Theorem, Cybenko 1989) *Let $h(x)$ be a continuous function defined on a compact subset $S \subset \mathbb{R}^d$. For a sufficiently large p , there exists an $f(x)$ as in Eq. (3.2) with p hidden units and sigmoid activation functions, such that for any $\epsilon > 0$:*

$$|h(x) - f(x)| < \epsilon, \forall x \in S$$

This holds for any non-constant, bounded, continuous Φ .

A NN of this form is a universal approximator. However, this theorem does not tell us about the feasibility of a given problem (e.g., how large should p be?). Because the NN can be highly non-convex, its optimization problem has multiple local minimum and/or saddle points.

Unlike a linear model, a NN has several design choices on which we have freedom. For instance, NN's depth (number of hidden layers $L - 1$) and width (number of neurons per layer N) influence its expressiveness. Deeper networks can capture hierarchical representations of data, which is a key advantage over shallow models. However, they also introduce challenges such as vanishing gradients (See 3.7.1), which can be mitigated with other design choices.

Some of them are the activation functions Φ , the choice of optimization algorithm and the initialization of the weights Υ_0 . We call these Hyper-parameters (HPs) to differentiate them from the parameters (weights) of the network. Choosing the correct set of HP is called the Hyper-parameters optimization (HPO) problem.

3.6 Backpropagation

The training of a NN is an unconstrained, highly non-convex problem. Furthermore, the minimization of loss function \mathcal{L} can be hard to achieve when the size of dataset \mathcal{T} is very large. To this end, we sample a subset $\mathcal{T}_t \subset \mathcal{T}$ from the dataset, which we call a mini-batch. Gradient descent applied on mini-batches of data is an example of Stochastic Gradient Descent (SGD). We aim to minimize the error function through an iterative method: where $d^k \in \mathbb{R}^d$ is the direction of movement and $\eta^k \in \mathbb{R}_+$ is the step size, also known as learning rate. Methods are defined by choosing:

- direction $d^k \in \mathbb{R}^d$
- step size $\eta^k > 0$

Algorithm 1 Stochastic Gradient Descent Algorithm.

```

1: Define an initial guess  $\Upsilon_0$  (e.g., random)
2: for  $k = 0, 1, 2, 3, \dots$  do
3:    $\Upsilon^{k+1} \leftarrow \Upsilon^k + \eta^k d^k$ 
4: end for
5: return  $\Upsilon^k$ 

```

The gradient method for MLP is often referred to as Backpropagation (BP) [64] which is a specialized technique of Automatic Differentiation (AD).

Definition 14 (Automatic differentiation) *AD refers to the task of simultaneously and efficiently computing all weight Jacobians of the MLP given knowledge of the computational graph and all individual input and weight Jacobians:*

$$AD(MLP(\mathbf{x})) = \{\partial_{\mathbf{w}_i} y\}_{i=1}^l.$$

Considering that the NN represents a compositional function, then BP applies the chain rule repeatedly to compute the derivatives. Combining the derivatives of the constituent operations by the chain rule gives the derivative of the overall composition. This algorithmic technique evaluates the gradient $\nabla \mathcal{L}(\Upsilon)$ that exploits the structure of the network.

This is related to the fact that the calculus of each $\partial_{\Upsilon_h} \mathcal{L}_p$ requires the forward propagation of the input and the backward propagation of the error appearing in the objective function. Each objective function and gradient evaluation with respect to all $w, b \in \Upsilon$ implies the propagation forward and backwards over all the layers:

1. Forward propagation of the input to evaluate $\mathcal{L}(\Upsilon)$;
2. Backward propagation of the error to evaluate $\nabla \mathcal{L}(\Upsilon)$.

AD capabilities are well-implemented in most DL frameworks such as TensorFlow [1] and PyTorch [60], and it allows us to avoid tedious derivations or numerical discretization while computing derivatives of all orders in space-time[8]. To demonstrate AD, we consider an FNN of only one hidden layer with two inputs x_1 and x_2 , and one output, y :

$$\begin{aligned} v &= -2x_1 + 3x_2 + 0.5 \\ h &= \tanh v \\ y &= 2h - 1. \end{aligned}$$

The forward pass and backward pass of AD for computing the partial derivatives $\partial_{x_1} y$ and $\partial_{x_2} y$ at $(x_1, x_2) = (2, 1)$ is shown in Table 3.1.

Table 3.1. Example of AD to compute the partial derivatives $\partial_{x_1}y$ and $\partial_{x_2}y$ at $(x_1, x_2) = (2, 1)$.

| Forward pass | Backward pass |
|---------------------------------|--------------------------------------------------------------------------------------------------|
| $x_1 = 2$ | $\partial_y y = 1$ |
| $x_2 = 1$ | |
| $v = -2x_1 + 3x_2 + 0.5 = -0.5$ | $\partial_h y = \partial_h(2h - 1) = 2$ |
| $h = \tanh v \approx -0.462$ | $\partial_v y = \partial_h y \partial_v h = \partial_h y \operatorname{sech}^2(v) \approx 1.573$ |
| $y = 2h - 1 = -1.924$ | $\partial_{x_1} y = \partial_v y \partial_{x_1} v = \partial_v y \times (-2) = -3.146$ |
| | $\partial_{x_2} y = \partial_v y \partial_{x_2} v = \partial_v y \times 3 = 4.719$ |

3.7 Hyper-parameters

3.7.1 Activation functions

The choice of activation function Φ significantly impacts the performance of a NN. Among the most common used activation functions Φ :

- S-Shape: which is a continuous differentiable monotonically increasing function such that

$$\lim_{t \rightarrow -\infty} \Phi(t) = -1, \quad \lim_{t \rightarrow \infty} \Phi(t) = 1.$$

Two examples of S-Shape activation functions are sigmoid: $\Phi(t) = \frac{1}{1+e^{-t}}$ and hyperbolic tangent: $\Phi(t) = \frac{e^t - e^{-t}}{e^t + e^{-t}}$;

- Linear Units $\Phi(t) = t$;
- Rectified Linear Units (ReLU) $\Phi(t) = \max(0, t)$ and other variants (like Leaky ReLU) are used to mitigate issues like dying neurons in standard ReLU.

Whenever a value goes through an activation function, during BP we multiply by the derivative of the function. The more layers we choose, the more of these multiplications will occur. As a result:

- If $\Phi'(\cdot) < 1$ always, the gradient will go to zero exponentially fast in the number of layers (vanishing gradient);
- If $\Phi'(\cdot) > 1$ always, the gradient will explode exponentially fast in the number of layers (exploding gradient).

3.7.2 Optimizers

Two optimizers that are widely used in DL are Adaptive Moment Estimation (Adam) [34] and Limited-memory Broyden–Fletcher–Goldfarb–Shanno (L-BFGS) [46]. While Adam is a first-order optimizer, L-BFGS is a quasi-Newton method, a second-order optimization technique. Each algorithm possesses distinctive characteristics, strengths, and limitations, rendering them appropriate for disparate scenarios, including their deployment in PINNs.

The core concept of Adam is to integrate the advantages of two existing algorithms, namely AdaGrad [17] and RMSProp [62], through the implementation of adaptive learning rates for each parameter. Adam employs estimates of both the first moment (mean) and the second moment (uncentered variance) of the gradients to adaptively adjust the learning rate for each parameter. This enables Adam to demonstrate greater resilience to noisy gradients, which are a common occurrence in DL tasks.

Table 3.2. Comparison of Adam and L-BFGS Optimizers.

| | Adam Optimizer | L-BFGS Optimizer |
|-------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Pros | <ul style="list-style-type: none"> - Increased robustness to noisy gradients - Adaptive learning rates - Efficient for large datasets - Handles sparse gradients well | <ul style="list-style-type: none"> - High precision in convergence - Rapid convergence for well-behaved problems - Effective for smaller datasets - Better for problems with smooth surfaces |
| Cons | <ul style="list-style-type: none"> - Sensitivity to HP - Suboptimal convergence - Difficulty with saddle points | <ul style="list-style-type: none"> - High computational cost - Not ideal for very large models or datasets - Less adaptable to noisy environments - Complexity in implementation |

L-BFGS is a quasi-Newton optimisation method that approximates the second derivatives (Hessian matrix) to guide parameter updates. In contrast to first-order methods, L-BFGS utilises past gradients to approximate the Hessian matrix, thereby facilitating an enhanced comprehension of the curvature of the loss surface. This second-order information enables L-BFGS to make more informed updates, which may ultimately result in a more precise convergence. Furthermore, L-BFGS can achieve faster convergence due to its use of second-order information, which enables it to reach precise solutions with fewer iterations. This is particularly advantageous in PINNs, where training is typically conducted on smaller datasets with high-dimensional inputs, making full-batch training feasible. In such scenarios, L-BFGS is particularly effective as it utilizes the entire dataset in each iteration, leading to efficient and accurate updates. As a drawback, L-BFGS can rapidly converge to a local minimum (for the residual) without exiting.

For this reason, the Adam optimizer is used first to avoid local minima, and then the solution is refined by L-BFGS [51].

3.7.3 Weights initialization

An initial distribution for weights and biases Υ_0 must be set when training a NN. These weights will be optimized during training to minimize the loss function \mathcal{L} . Usually, it is recommended to choose Υ_0 small and close to 0. By doing so, the optimization algorithm will be able to make greater updates at the start of the training phase.

The initialization method can be random or deterministic, where the first is often preferred, as it helps to break the symmetry between neurons and allows them to learn different features during training. On the other hand, deterministic initialization can lead to poor performance, as all neurons may learn the same features, resulting in a lack of diversity in the learned representations. Different initialization methods may be suitable for different problems and network architectures, as they can influence the required time to reach the optimized solution, and can also emphasize or lower the aforementioned problems of the gradients (vanishing or exploding gradient). Some initialization methods are given in the PyTorch module `nn.init`.

The most common are Xavier (or Glorot) [25], and Kaiming (or He) [31], recommended to use only with ReLU or Leaky ReLU activation functions. They are available in two variants, depending on the weights distribution that can be uniform (the weights are evenly distributed along the interval), or normal (the weights are evenly distributed around the mean).

3.8 Artificial Intelligence for Health

NN, by nature, are powerful function approximators, making them well-suited for learning complex physical relationships. Convolutional Neural Networks (CNNs) and U-nets are effective for handling high-dimensional spatial data and can learn spatial hierarchies. In fields such as image recognition and segmentation, NNs have already demonstrated their potential by enhancing automation, improving diagnostic accuracy, and reducing interpretation time [5]. A notable application of CNN is their use in mapping patients' electrical properties from Magnetic Resonance (MR) imaging [23].

In the context of solving a PDE, after the training phase, the network is expected to approximate the solution to the partial differential equation with a certain degree of accuracy, depending on the type of network and its parameters.

For time-series data, Recurrent Neural Networks (RNNs) and Long Short-Term Memory Networks (LSTMs) can model the temporal evolution of PDE solutions. Additionally, autoencoders are useful for dimensionality reduction and feature extraction, as they can reduce computational complexity and capture essential features of the state. Although originally designed for other tasks, these NN architectures have shown promising results in the field of PDE estimation. However, they need a consistent dataset of the solution for the supervised training.

Novel AI approaches allow us to train models without any dataset, but only with the knowledge of the physical principle, even with equations that do not have closed-form solutions. Knowledge of the principled physical law that governs the time-dependent dynamics of a system can act as a regularization agent that constrains the space of admissible solutions [63]. This is the case of Physics-informed Learning and PINNs [63] in particular. An alternative to PINNs is Kolmogorov-Arnold Networks (KANs), inspired by the Kolmogorov-Arnold representation theorem. KANs have a core architecture that slightly differs from MLPs: they replace traditional linear weights with spline-parametrized univariate functions, allowing them to dynamically learn activation patterns. This seemingly simple change enables KANs to outperform MLPs in terms of accuracy and interpretability [48].

3.8.1 Physics-Informed Neural Networks

Since a great amount of data, not always available, is usually required for training deep NNs, the key idea in solving PDEs with the Physics-Informed approach is to use additional information obtained by enforcing physical laws at random points in the continuous space-time domain to train such networks. Encoding such structured information into a learning algorithm results in amplifying the information content of the data the algorithm sees, enabling it to quickly steer itself towards the right solution. This approach is known as physics-informed learning. In PINNs, the core architecture typically used is a MLP, which consists of multiple layers of neurons where each layer is fully connected to the next. PINNs incorporate the underlying physics of the system, represented by PDEs, directly into the NN training process. We can think of PINNs as two interconnected networks (see Figure 5.2). The first network (trainable network) has trainable weights and biases and takes as input the coordinates of a point in the problem domain \mathbf{x} and predicts the approximate solution u_{NN} . The second network (residual network) computes the loss function based on the approximate solution of the first network and the PDE problem constraints. Derivatives of the solution with respect to the inputs are computed using AD.

Algorithm and theory

Consider the following partial differential equation parametrized by ψ for the solution $u(\mathbf{x})$ with $\mathbf{x} = (x_1, \dots, x_{d_{\text{in}}})$ defined on a domain $\Omega \in \mathbb{R}^{d_{\text{in}}}$:

$$\mathcal{N}(u, \mathbf{x}, \psi) = 0, \quad \mathbf{x} \in \Omega$$

and the BC:

$$\tilde{g}(u, \mathbf{x}) = 0, \quad \mathbf{x} \in \partial\Omega.$$

Where \mathcal{N} is a general linear or nonlinear differential operator; $\partial\Omega$ denote the boundary of the computational domain; $\tilde{g}(\mathbf{x})$ represent some BCs. We remark that this formulation can be easily generalized to higher-order PDEs since they can be written as systems of first-order PDEs. Following the framework of PINNs proposed in [63], $u(\mathbf{x})$ is approximated by a fully connected network, which takes the coordinates \mathbf{x} as inputs and outputs a vector $u_{\text{NN}}(\mathbf{x}, \Upsilon)$ with the same dimension as u . We recall that Υ is the set of all weight matrices and bias vectors in the NN u_{NN} .

In the next step, we need to select the scattered points \mathcal{T} which restrict the NN u_{NN} to satisfy the physics imposed by the PDE and BCs. The residual points \mathcal{T} are comprised of two sets, $\mathcal{T}_{\text{res}} \subset \Omega$ and $\mathcal{T}_{\text{b}} \subset \partial\Omega$. The Υ of the network Υ can be trained by minimizing a composite loss function taking the form

$$\mathcal{L}(\Upsilon; \mathcal{T}) = w_{\text{res}} \mathcal{L}_{\text{res}}(\Upsilon; \mathcal{T}_{\text{res}}) + w_{\text{bc}} \mathcal{L}_{\text{b}}(\Upsilon; \mathcal{T}_{\text{b}}), \quad (3.5)$$

where

$$\begin{aligned} \mathcal{L}_{\text{res}}(\Upsilon; \mathcal{T}_{\text{res}}) &= \frac{1}{|\mathcal{T}_{\text{res}}|} \sum_{\mathbf{x} \in \mathcal{T}_{\text{res}}} \left\| \mathcal{N}(u, \mathbf{x}, \psi) \right\|_2^2, \\ \mathcal{L}_{\text{b}}(\Upsilon; \mathcal{T}_{\text{b}}) &= \frac{1}{|\mathcal{T}_{\text{b}}|} \sum_{\mathbf{x} \in \mathcal{T}_{\text{b}}} \|g(u, \mathbf{x})\|_2^2, \end{aligned}$$

and w_{res} and w_{bc} are the loss weights. To compute the residuals for \mathcal{L}_{res} , derivatives of the outputs with respect to the inputs are required. Such computation is achieved in the PINN framework using AD in the DL code.

In addition, if we have further information on some points $\mathcal{T}_i \subset \Omega$:

$$\mathcal{I}(u, \mathbf{x}) = 0 \quad \text{for } \mathbf{x} \in \mathcal{T}_i,$$

an extra loss term indicating the mismatch between the predictions and the data can be taken into account:

$$\mathcal{L}_i(\Upsilon; \mathcal{T}_i) = \frac{1}{|\mathcal{T}_i|} \sum_{\mathbf{x} \in \mathcal{T}_i} \|\mathcal{I}(u_{\text{NN}}, \mathbf{x})\|_2^2.$$

This is useful for inverse problems, where some uncertain parameters ψ can be optimized during the training. From an implementation point of view, PINNs solve inverse problems as easily as forward problems. The only difference between solving forward and inverse problems is that we add an extra loss term to Eq. (3.5):

$$\mathcal{L}(\Upsilon, \psi; \mathcal{T}) = w_{\text{res}} \mathcal{L}_{\text{res}}(\Upsilon, \psi; \mathcal{T}_{\text{res}}) + w_{\text{bc}} \mathcal{L}_b(\Upsilon, \psi; \mathcal{T}_b) + w_i \mathcal{L}_i(\Upsilon, \psi; \mathcal{T}_b)$$

The last step is called training and consists of the procedure of searching for a good Υ by minimizing the loss $\mathcal{L}(\Upsilon; \mathcal{T})$. When Υ and ψ are optimized together, the solution is

$$\Upsilon^*, \psi^* = \arg \min_{\Upsilon, \psi} \mathcal{L}(\Upsilon, \psi; \mathcal{T}).$$

In the algorithm of PINN introduced above, we enforce soft constraints of BCs/IC through the loss \mathcal{L}_b . This approach can be used for complex domains and any type of BCs. On the other hand, it is possible to enforce hard constraints for simple cases [40], by transforming the output of the trainable network u_{NN} using an ansatz:

$$\tilde{u}_{\text{NN}}(\mathbf{x}) = \tilde{g}(\mathbf{x}) + h(\mathbf{x})u_{\text{NN}}(\mathbf{x}).$$

Approximation theory and error analysis

Unlike traditional numerical methods, for PINNs there is no guarantee of unique solutions because PINN solutions are obtained by solving nonconvex optimization problems, which in general do not have a unique solution. In practice, to achieve a good level of accuracy, we need to tune all the HPs, e.g., network size, learning rate, and the number of residual points. The required network size depends highly on the smoothness of the PDE solution. As an instance, Ref. [50] reports that a small network (e.g., a few layers and twenty neurons per layer) is sufficient for solving the 1D Poisson equation, but a deeper and wider network is required for the 1D Burgers equation to achieve a similar level of accuracy. In the same reference is also noted that PINNs may converge to different solutions from different network initial values, and thus a common strategy is to train PINNs from random initialization a few times (e.g., 10 independent runs) and choose the network with the smallest training loss as the final solution. An exhaustive analysis of approximation theory and error analysis is given in Ref. [53] and Ref. [54].

3.9 Improving the training and establishing a validation benchmark

PINNs exhibit notable training pathologies, as clearly pointed out in [72]. Also, their effectiveness can be hampered by poor choices made by users who lack DL expertise. The model's test accuracy depends significantly on architectural choices and training strategies. Fortunately, several best practices can substantially enhance both training efficiency and overall accuracy of PINNs. By establishing guiding principles, researchers can create strong baselines that advance the state-of-the-art in physics-informed learning. This standardized approach would ensure consistency across various applications and improve reproducibility.

3.9.1 Tools and packages

Several libraries facilitate the implementation of PINNs, both in commercial and research contexts. For instance, NVIDIA Modulus provides a framework tailored for physics-driven DL applications, while DeepXDE, an open-source Python library, has been widely adopted in academic and industrial research. Additionally, tools such as Weights & Biases enable systematic tracking of model parameters and performance metrics, while platforms like GitHub ensure version control and collaborative development. Optimization frameworks, including scikit-optimize, play a crucial role in refining model performance, and Hydra [75], a flexible configuration management tool, facilitates experimental reproducibility. We can visualize the slopes of functions using the matplotlib library. For the validation of PINN models, an integration of numerical solvers as MATLAB in the same workspace would be beneficial. To ensure the portability of such a workspace, one can consider a containerized environment setup that integrates essential libraries for PINN implementation, alternative DL architectures, and numerical solvers.

3.9.2 Hyper-parameters optimization

HPO is a systematic approach to identify the optimal set of HPs for a given model, enhancing its performance and generalization capabilities. By nature, PINNs exhibit several HPs such as the learning rate η , the width N and depth $L - 1$ of the network, the activation function Φ , and the weights ($w_{\text{res}}, w_{\text{bc}}$) for the loss function. The high dimensionality for the HP search space makes it difficult to find proper configurations. To remedy this concern, several search algorythms can be adopted.

The search algorythm

Grid search and random search are two fundamental methods for HPO. Grid search systematically explores a predefined set of hyperparameter values by evaluating all possible combinations, ensuring a thorough search but becoming computationally expensive as the number of parameters grows. On the other hand, random search selects HP configurations randomly within given ranges, which can be more efficient in high-dimensional spaces but lacks the structured exploration of grid search. Still, the first one suffers from the curse of dimensionality, and the other can be very

Table 3.3. Search space for the HPO.

| HP | Symbol | Range |
|---------------------|--------------|----------------------|
| Learning rate [-] | η | [1.00E-04, 5.00E-02] |
| Depth [-] | $L - 1$ | [1, 6] |
| Width [-] | N | [5, 250] |
| Activation function | Φ | {sin, sigmoid, tanh} |
| Initialization | Υ_0 | [He, Glorot] |

inefficient. To mitigate these drawbacks, a solution can be Gaussian processes-based Bayesian optimization.

HPO of PINNs is approached in [18] as a bi-level optimization problem: in the inner cycle, a PINN is trained with a set of HP γ , the optimal weights and biases Υ^* are retrieved. In the outer cycle, the optimal HP set γ^* is obtained by minimizing a loss function.

The search space is Γ , the Cartesian product of all HP ranges. An exhaustive search space is reported in Table 3.3. As shown by Ref. [53], PINNs requires sufficiently smooth activation functions. PINNs with ReLU and other non-smooth activation functions, such as ELU and SELU (Exponential and Scaled Exponential Linear Units) are not “consistent/convergent” methods: in the limit of an infinite training dataset a well-trained PINN with ReLU-like activation functions, the solution does not converge to the exact solution [54]. Every $\gamma \in \Gamma$ writes as:

$$\gamma = [\eta, L - 1, N, \Phi, \Upsilon_0].$$

The starting HP set is:

$$\gamma_0 = [1e - 3, 4, 50, "tanh", "Glorot normal"].$$

The optimization algorithm of the outer cycle is implemented with the `scikit-optimize` Python package. The acquisition function allows to define the next point at each iteration. Typically, we iterate the outer cycle for 100 times.

3.9.3 Other techniques

Here, we report some of the many techniques that can be applied to optimize the training procedure of PINNs.

Initial Weights Regularization (IWR) is a technique used to ensure that all terms in a loss function \mathcal{L} contribute equally at the beginning of training. This is achieved by normalizing each loss term so that they start from the same magnitude, preventing any single term from dominating the optimization process. By doing so, the model maintains a more balanced gradient flow across different components of the loss, leading to more stable and efficient convergence. This approach is particularly useful when combining different types of loss functions, as it mitigates issues arising from scale differences.

Resampling is a strategy to improve accuracy and stability during training. By updating the set of collocation points used to enforce the governing equations, the

network can better capture complex features such as sharp gradients, singularities, or regions of high error. An evolution to this technique is the Residual-based Adaptive Refinement (RAR), which ensures that training focuses on the most challenging areas, preventing underfitting in critical regions and enhancing overall convergence.

In [22], the dropout NN regularization technique is used to gain information regarding uncertainty propagation. In [4], warmstart (i.e., to initialize using the solution of the previous iterate) is exposed as a way to reduce the resource usage associated with the construction of performant DL systems. In [51], the importance of transfer learning and fine-tuning is stressed out.

Chapter 4

A State Observer of PBHE

This chapter presents formal approaches to the observation of bio-heat transfer from boundary measurements in the presence of uncertain system properties. The application is real-time temperature monitoring of superficial HT treatments, intending to provide essential feedback information consisting of temperature distribution at the target despite the uncertain reaction coefficient (blood perfusion rate w_b).

For parabolic reaction-diffusion equations with unknown reaction coefficients, such as PBHE, backstepping observer design [66, 3] appears to be the primary viable approach to adaptive estimation. However, backstepping observers rely on complex integral transformations that can limit real-time application efficiency. To address this limitation, we investigate an alternative framework based on multiple-model adaptive estimation, inspired by approaches described by [30] for finite-dimensional linear systems.

Our proposed multiple-model adaptive observer consists of a family of observers with access to temperature measurements at the system boundaries. Each observer evolves according to a copy of the PBHE but with different assigned perfusion values. The overall temperature estimation is calculated as a weighted average of each observer's prediction, with weights evolving according to an Ordinary Differential Equation (ODE) named the Dynamic Convex Weighting Generator (DCWG). We demonstrate that the observation error remains ultimately bounded and converges to the observer whose perfusion value most closely matches the actual value. This chapter advances both the solution of PDEs with unknown coefficients and state estimation techniques for temperature monitoring in HT treatments. To our knowledge, this represents the first application of multiple-model adaptive estimation in this specific context.

The chapter is organized as follows: after an introduction to the boundary control of PDEs, we formulate the specific problem for the superficial HT application. Next, we design and provide a convergence proof for a single observer in the nominal case, and in the case of an unknown reaction coefficient, the perfusion w_b . Finally, we introduce the concept of multiple-model observer for the adaptive control of PBHE in the case of unknown but bounded perfusion.

4.1 Boundary control of PDEs

The boundary control of PDEs finds application in several engineering fields. These include fluid flows in aerodynamics and propulsion applications; plasmas in lasers, fusion reactors, and hypersonic vehicles; liquid metals in cooling systems for nuclear reactors and computers, as well as in welding and metal casting processes; acoustic waves and water waves in irrigation systems. It is also applied to flexible structures in civil engineering, aircraft wings, and helicopter rotors, astronomical telescopes, and in nanotechnology devices such as the atomic force microscope. Additional applications involve electromagnetic waves and quantum mechanical systems, waves and “ripple” instabilities in thin film manufacturing and flame dynamics, as well as chemical processes in process industries and internal combustion engines. Control objectives for PDE systems include performance improvement for stable systems and optimal control, stabilization of unstable plants, trajectory tracking, and trajectory generation/motion planning.

Despite “in domain” control, where actuation penetrates inside the domain of the PDE system, boundary control is physically more realistic because actuation and sensing are usually non-intrusive. In thermal problems, the choices of boundary controls can be dedicated to the actuation of temperature or heat flux, or a mix of both. The mathematical equivalents of these choices of boundary control are, respectively, Dirichlet control (actuate value of a function at the boundary), Neumann control (actuate slope of a function at the boundary), and Robin control (mix of both). In our application, the boundary control is dedicated to the regulation of the water bolus temperature for the Robin superficial BC.

In the next sections, some basics of stability for ODEs, and their extension to PDEs, are given. Additionally, we provide insights about the specific use case, bio-heat transfer, and an overview of the proposed approach.

4.1.1 Stability analysis

Definition 15 (Exponential stability) *An ODE*

$$\dot{z} = Az, \quad z \in \mathbb{R}^n$$

is exponentially stable at $z = 0$ if there exist constants $M > 0$ (overshoot coefficient) and $\alpha > 0$ (decay rate) such that

$$\|z(t)\| \leq M e^{-\alpha t} \|z(0)\|,$$

for all $t \geq 0$, where $\|\cdot\|$ denotes any norm in \mathbb{R}^n (e.g., the 2-norm).

A sufficient condition for exponential stability is that all eigenvalues of the matrix A have negative real parts. While this spectral condition is mathematically straightforward, it is often impractical to use directly, especially in the context of nonlinear systems, time-varying systems, or systems described by PDEs.

To address these limitations, Lyapunov’s method provides a more constructive and widely applicable approach. Instead of relying solely on the system matrix, the method introduces an energy-like scalar function—called a Lyapunov function—which helps assess the stability of the system by tracking its evolution over time.

Definition 16 (Lyapunov equation) For every positive definite matrix Q , there exists a unique positive definite and symmetric matrix P satisfying

$$PA + A^T P = -Q. \quad (4.1)$$

A function of the form $V(x) = x^T Px$ is a natural candidate for a Lyapunov function. If P satisfies (4.1), then $V(x)$ is positive definite, and its time derivative along trajectories of the system is

$$\dot{V}(x) = \frac{d}{dt}(x^T Px) = x^T(PA + A^T P)x = -x^T Qx,$$

which is negative definite. This implies that $V(x)$ is strictly decreasing along nonzero trajectories, capturing the idea that the system’s “energy” is dissipating over time. Hence, the Lyapunov function serves as a powerful tool for proving stability without explicitly solving the system or computing eigenvalues.

For PDEs, an (infinite-dimensional) operator like (4.1) is hard to solve. Hence, we often can’t use standard Lyapunov functions. Instead, system norms are employed to measure stability. In finite-dimensional Euclidean spaces, all norms are equivalent. However, in infinite-dimensional settings of PDEs, the state space is typically a function space, and the associated norms are function norms rather than standard vector norms. Unlike in finite dimensions, norms on function spaces are generally not equivalent. As a result, when analyzing PDEs, different norms can yield different perspectives on stability and behavior, making the definition and analysis of stability considerably more nuanced. In this study, the \mathcal{L}^2 -norm of the error between observation variable and observer dynamics will be considered to assess the observability.

4.1.2 Nonlinearities in bioheat transfer

At present, boundary control methods for nonlinear PDEs remain limited. While several results exist for nonlinear PDEs that are neutrally stable with nonlinearities playing no destabilizing role, advanced control designs for broad classes of open-loop unstable nonlinear PDEs, where sophisticated non-quadratic control Lyapunov functions are needed for closed-loop stability, are largely undeveloped. In bioheat applications, nonlinearities might arise from temperature-dependent tissue properties, non-constant perfusion rates, or metabolic heat generation that varies nonlinearly with temperature, making these control techniques especially relevant for accurate modeling and effective control.

To avoid the nonlinear problem of having w_b as a function of the state variable, this study considers a constant perfusion. A similar theoretical formulation can be achieved in the case of an unknown time-varying reaction coefficient w_b . A more accurate way to model perfusion is to consider it as a temperature-dependent parameter (See [16, Par 2.3]), consisting of a nominal value plus an enhancement factor that changes qualitatively with temperature and quantitatively according to tissue type. The nominal perfusion value varies dramatically across tissue types, ranging from 10^{-5} [s^{-1}] to 10^{-3} [s^{-1}] for muscle, and can be enhanced up to tenfold as temperature increases, depending on the specific tissue involved.

4.1.3 Proposed approach: multiple-model adaptive observer

Unstable PDEs with unknown parameters can be effectively controlled using parameter estimators supplied by identifiers alongside state estimators supplied by adaptive observers. This approach is particularly suitable for this application, since perfusion rates vary significantly across different tissues and physiological conditions. In the present thesis, an ensemble of observers is employed, each one designed following a systematic procedure that has been reported in Algorithm 2.

Algorithm 2 Approach to the design of the adaptive observer.

- 1: Given the observation variable u , formulate observer dynamics \hat{u} providing its estimation;
 - 2: Define the observation error: $e_u = u - \hat{u}$;
 - 3: Analyze the error dynamics and construct a Lyapunov functional candidate:
 $V(t) = \frac{1}{2} \int_0^1 e_u^2 dx$;
 - 4: Prove exponential decay: $\|e_u(t)\|_{\mathcal{L}^2} \rightarrow 0$.
-

In Section 4.3.1, we provide an evaluation for the nominal case where perfusion w_b is known; then, in Section 4.3.2, a sensitivity analysis for cases where perfusion is unknown and the observer considers a guess \tilde{w}_b , is provided. In this case, we demonstrate that the \mathcal{L}^2 -norm of the observation error is ultimately bounded by a factor that depends on the difference between actual perfusion w_b and the guess of the observer \tilde{w}_b . From this result, and given the boundedness nature of the parameter w_b , in Section 4.4 we will introduce the MM-Obs. This estimation framework employs a weighted ensemble of observers with different perfusion values, dynamically adjusting their contributions based on observed error magnitudes. This adaptive strategy consists of an extension of [30] to continuous-time settings and it is especially suitable for practical implementations where parameter uncertainty is unavoidable, such as in biomedical applications involving heat transfer in perfused tissues.

4.2 Problem formulation

Recalling the normalized PBHE describing the heat propagation within a tissue in one dimension:

$$a_1 \partial_\tau \theta = \partial_{XX} \theta - w_b a_2 \theta + \bar{P}(\tau) a_3 \exp(-a_4 X),$$

with $(X, \tau) \in [0, 1] \times [0, 1]$, consider that the temperature of the fluid above the domain is a controlled input $\bar{v}(\tau)$ and that the right endpoint is sufficiently deep to be localized outside of the heating range of the applicator. Then, the BCs can be formulated as:

$$\begin{aligned} \partial_X \theta|_{X=0} &= -a_5(\bar{v}(\tau) - \theta(0, \tau)), \quad \tau \geq 0, \\ \theta(1, \tau) &= 0, \quad \tau \geq 0. \end{aligned}$$

The IC is represented by a third-order polynomial satisfying the compatibility with the BCs as explained in Sect.4.4.3, and adding a further condition on an internal peak temperature $\theta_0(X_{gt}) = gt(0)$:

$$\theta_0(X) = b_1 X^3 + b_2 X^2 + b_3 X + b_4 \quad X \in [0, 1].$$

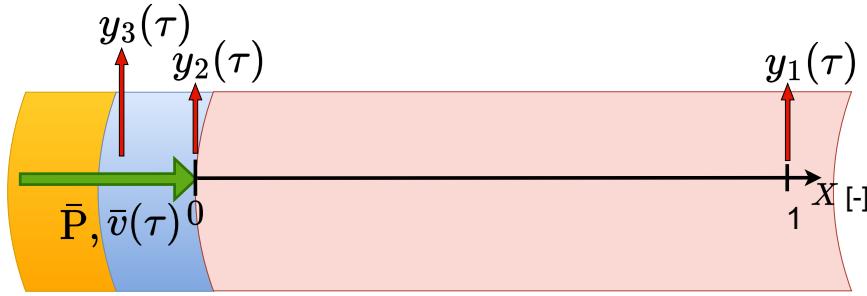


Figure 4.1. Scheme of the considered scaled setup. Consider an ideal needle that enters perpendicularly from the patient's surface, the left boundary being the surface and the right one being the in-depth bound. Heat is given the system with EM power \bar{P} , while water bolus at the controlled temperature v prevents skin burns. The boundary measurements from the system are y_3 , y_2 and y_1 . If the domain is sufficiently extended (~ 7 [cm]), the in-depth bound can be considered outside of the heating range, hence the measurement $y_1=0$.

In the setup described in Figure 4.1, we assume to know the temperature at the right endpoint $y_1(\tau)$, equal to 0 since no temperature rise is experienced outside of the heating range of the antenna, and to measure (non-invasively) the superficial temperature at the left endpoint y_2 , and the temperature of the fluid $y_3=\bar{v}(\tau)$. Hence, the superficial heat flux $g_0(\tau) = -a_5(y_3(\tau) - y_2(\tau))$ is known (measured output or forcing input). The functions y_1 , y_2 , y_3 , and θ_0 are assumed to be continuously differentiable, as this guarantees to deal with strong (smooth) solutions to the PDE.

The goal is to estimate in real-time the temperature $\theta(X, \tau)$ and the uncertain w_b , given the measurements y_1 , y_2 , y_3 , and the underlying physical law (PBHE).

4.3 Observer structure

In the following, we will design the observer following the steps in Algorithm 2. Specifically, the objective is to observe bio-heat transfer in a mono-dimensional domain by the knowledge of boundary temperatures y_1 , y_2 , and y_3 . An evaluation is given for the nominal case, in which the perfusion w_b of the system is known, together with a sensitivity analysis for the case in which the perfusion is not known.

4.3.1 Nominal case (known perfusion)

Let us preliminarily assume that $w_b > 0$ is constant and known, and address the observer design problem. To construct an observer $\hat{\theta}(X, \tau)$, consider first a copy of the PBHE dynamics

$$a_1 \partial_\tau \hat{\theta} = \partial_{XX} \hat{\theta} - w_b a_2 \hat{\theta} + \bar{P}(\tau) a_3 \exp(-a_4 X) \quad (4.2)$$

with BCs

$$\begin{aligned} \partial_X \hat{\theta} \Big|_{X=0}(\tau) &= g_0(\tau) + K(\hat{y}(\tau) - y_2(\tau)), \\ \hat{\theta}(1, \tau) &= y_1(\tau), \end{aligned} \quad (4.3)$$

where $K > 0$ is the output injection gain and $\hat{\theta}(\tau) = \hat{\theta}(0, \tau)$ is the observer's superficial temperature estimate. The compatibility conditions for the initialization of the observer $\hat{\theta}_0(X)$ read then as

$$\partial_X \hat{\theta}|_{X=0}(0) = \partial_X \theta|_{X=0}(0) + K(\hat{y}(0) - y_2(0)) \quad \wedge \quad \hat{\theta}_0^{(j)}(0) = y_1(0),$$

i.e.,

$$\partial_X \hat{\theta}|_{X=0}(0) = g_0(0) + K(\hat{y}(0) - y_2(0)) \quad \wedge \quad \hat{\theta}_0^{(j)}(1) = y_1(0). \quad (4.4)$$

Remark 4.4.3 provides a family of functions satisfying the above conditions. In light of Algorithm 2, now we define and analyze the observation error dynamics, proving its exponential decay.

Setting $e_\theta(X, \tau) := \theta(X, \tau) - \hat{\theta}(X, \tau)$ and $e_\theta^0(X) := e_\theta(X, 0)$, the error dynamics is governed by the PDE

$$\begin{aligned} a_1 \partial_\tau e_\theta &= \partial_{XX} e_\theta - w_b a_2 e_\theta, \\ \partial_X e_\theta(0, \tau) &= K e_\theta(0, \tau), \\ e_\theta(1, \tau) &= 0. \end{aligned} \quad (4.5)$$

From Eq. (2.8), $\frac{a_2}{a_1} = t^*$, so the first equation in (4.5) can be recast as

$$\partial_\tau e_\theta = \frac{1}{a_1} \partial_{XX} e_\theta - w_b t^* e_\theta. \quad (4.6)$$

Proposition 4.3.1 Assume that $\hat{\theta}_0^{(j)}(X)$ is smooth, and consider the observer dynamics (4.2)-(4.3). For any $K > 0$, the L^2 -norm of the error

$$\|\theta(\cdot, \tau) - \hat{\theta}(\cdot, \tau)\|_{L^2}$$

converges exponentially to zero, with a decay rate not smaller than

$$w_b t^* + \min \left\{ \frac{K}{a_1}, \frac{\pi^2}{4a_1} \right\}.$$

Proof. Consider the Lyapunov functional candidate

$$V(\tau) = \frac{1}{2} \int_0^1 e_\theta^2 dX$$

corresponding to $\frac{1}{2} \|e_\theta\|_{L^2}^2$. We aim at differentiating $V(\tau)$ along the error system solutions and, as the latter are strong solutions thanks to the smoothness of initial and BCs, one is allowed to bring the derivative under the integral sign, so that

$$\begin{aligned} \dot{V}(\tau) &= \int_0^1 e_\theta \partial_\tau e_\theta dX \\ &= \frac{1}{a_1} \int_0^1 e_\theta \partial_{XX} e_\theta dX - w_b t^* \int_0^1 e_\theta^2 dX, \end{aligned}$$

where the equation of the error system (4.6) has been used. Let us focus on the first term on the right-hand side. Performing integration by parts and using the BCs,

one has

$$\begin{aligned} \int_0^1 e_\theta \partial_{XX} e_\theta dX &= e_\theta \partial_X e_\theta \Big|_0^1 - \int_0^1 (\partial_X e_\theta)^2 dX \\ &= e_\theta(1, \tau) \partial_X e_\theta(1, \tau) - e_\theta(0, \tau) \partial_X e_\theta(0, \tau) - \int_0^1 (\partial_X e_\theta)^2 dX \\ &= -K e_\theta(0, \tau)^2 - \int_0^1 (\partial_X e_\theta)^2 dX. \end{aligned}$$

Now, recalling Poincaré-Wirtinger inequality [39, Remark 2.2], we have that

$$-K e_\theta(0, \tau)^2 - \int_0^1 (\partial_X e_\theta)^2 dX \leq -\xi \int_0^1 e_\theta^2 dX,$$

with

$$\xi = \xi(K) = \begin{cases} K & K \in \left(0, \frac{\pi^2}{4}\right) \\ \frac{\pi^2}{4} & K \geq \frac{\pi^2}{4}. \end{cases}$$

Summarizing, we have shown that

$$\dot{V}(\tau) \leq -\left(\frac{\xi}{a_1} + w_b t^*\right) \int_0^1 e_\theta^2 dX = -2\left(\frac{\xi}{a_1} + w_b t^*\right) V(\tau),$$

thus yielding exponential convergence of $V(\tau)$ with a decay rate Λ not smaller than:

$$\Lambda \geq 2\left(\frac{\xi}{a_1} + w_b t^*\right) [\text{s}^{-1}]. \quad (4.7)$$

The statement then follows by the simple observation that the norm $\|e_\theta\|_{L^2}$ satisfies the identity

$$\|e_\theta\|_{L^2} = \sqrt{2}V(\tau)^{\frac{1}{2}}. \clubsuit$$

4.3.2 Unknown perfusion

The previous result proves observer convergence in nominal conditions. Now, we investigate the sensitivity of the error with respect to uncertainty in the coefficient w_b . Consider a claimed value \tilde{w}_b to be used in the observer state equation, which becomes

$$a_1 \partial_\tau \hat{\theta} = \partial_{XX} \hat{\theta} - \tilde{w}_b a_2 \hat{\theta} + a_3 \bar{P}(\tau) \exp(-a_4 X), \quad (4.8)$$

with BCs left unchanged. Considering again the Lyapunov functional $V(\tau)$ for the error $e_\theta(X, \tau)$ and following the same steps of the proof of Proposition 4.3.1, we end up with the inequality

$$\dot{V}(\tau) \leq -2\frac{\xi}{a_1} V(\tau) - w_b t^* \int_0^1 \theta e_\theta dX + \tilde{w}_b t^* \int_0^1 \hat{\theta} e_\theta dX. \quad (4.9)$$

Eq. (4.9) is then transformed in

$$\dot{V}(\tau) \leq -2\frac{\xi}{a_1} V(\tau) - w_b t^* \int_0^1 \theta e_\theta dX + \tilde{w}_b t^* \int_0^1 \hat{\theta} e_\theta dX.$$

Now, adding and subtracting the same quantity $\tilde{w}_b t^* \int_0^1 \theta e_\theta dX$ and using Young's inequality, the latter can be rewritten as

$$\begin{aligned}\dot{V}(\tau) &\leq -2 \frac{\xi}{a_1} V(\tau) - \tilde{w}_b t^* \int_0^1 e_\theta^2 dX + t^* (\tilde{w}_b - w_b) \int_0^1 \theta e_\theta dX \\ &= -2 \left(\frac{\xi}{a_1} + \tilde{w}_b t^* \right) V(\tau) + t^* (\tilde{w}_b - w_b) \int_0^1 \theta e_\theta dX \\ &\leq -2 \left(\frac{\xi}{a_1} + \tilde{w}_b t^* \right) V(\tau) + t^* \frac{|\tilde{w}_b - w_b|}{2} \left(\frac{1}{a} \int_0^1 \theta^2 dX + a \int_0^1 e_\theta^2 dX \right) \\ &= - \left(2 \frac{\xi}{a_1} + 2\tilde{w}_b t^* - at^* |\tilde{w}_b - w_b| \right) V(\tau) + t^* \frac{|\tilde{w}_b - w_b|}{2a} \int_0^1 \theta^2 dX,\end{aligned}$$

where $a > 0$ is an arbitrary positive scalar. Notice that, however, to keep the coefficient in the first term on the right-hand side negative, we must impose the upper bound $a < 2 \left(\frac{\xi}{a_1} + \tilde{w}_b t^* \right) / |\tilde{w}_b - w_b| t^*$.

Based on the previous computations, we can give the following corollary addressing observer sensitivity.

Corollary 4.3.1 *Let $w_b > 0$ be unknown, and consider the observer (4.8) with some guess $\tilde{w}_b > 0$. Assume that a constant $\theta_{\max} = 1$ exists such that the bound $\theta(X, \tau) \leq \theta_{\max}$ holds for any $X \in [0, 1]$ and any $\tau \geq 0$. Then the error $e_\theta(X, \tau)$ is ultimately bounded relative to the set*

$$\mathcal{E} := \{e_\theta \in L^2(0, 1) : \|e_\theta\|_{L^2}^2 \leq 4c_0\},$$

where $c_0 = \left(\frac{t^* |\tilde{w}_b - w_b|}{\frac{\xi}{a_1} + \tilde{w}_b t^*} \right)^2$. Furthermore, the convergence towards the attractive set \mathcal{E} is exponential, with a rate not smaller than $\left(\frac{\xi}{a_1} + \tilde{w}_b t^* \right) / 2$.

Proof. From earlier derivations, using the upper bound on $\theta(X, \tau)$ and further rearranging terms, we have

$$\begin{aligned}\dot{V}(\tau) &\leq -(2 \frac{\xi}{a_1} + 2\tilde{w}_b t^* - a|\tilde{w}_b - w_b|t^*) V(\tau) + \frac{|\tilde{w}_b - w_b|t^*}{2a} \int_0^1 \theta^2 dX \\ &\leq -\left(\frac{\xi}{a_1} + \tilde{w}_b t^* \right) V(\tau) - \left((\frac{\xi}{a_1} + \tilde{w}_b t^* - a|\tilde{w}_b - w_b|t^*) V(\tau) - \frac{|\tilde{w}_b - w_b|t^*}{2a} \right).\end{aligned}$$

The quantity in the brackets is non-negative whenever

$$V(\tau) \geq \frac{|\tilde{w}_b - w_b|t^*}{2a \left(\frac{\xi}{a_1} + \tilde{w}_b t^* - a|\tilde{w}_b - w_b|t^* \right)}$$

and, minimizing with respect to the positive parameter a , one gets the sharper condition

$$V(\tau) = \frac{1}{2} \|e_\theta(\cdot, \tau)\|_{L^2}^2 \geq 2c_0,$$

which in turn implies that

$$\dot{V}(\tau) \leq - \left(\frac{\xi}{a_1} + \tilde{w}_b t^* \right) V(\tau) \quad \forall e_\theta(\cdot, \tau) \in \mathcal{E}^c := L^2(0, 1) \setminus \mathcal{E}.$$

The latter inequality guarantees global attractivity of the bounded set \mathcal{E} , with the claimed rate of exponential convergence:

$$\Xi \geq \left(\frac{\xi}{a_1} + \tilde{w}_b t^* \right) / 2 \quad [\text{s}^{-1}]. \quad (4.10)$$



4.3.3 Discussion

This section presents a numerical analysis of the decay rates and error bounds associated with observer performance, considering different values of the uncertain reaction coefficient (perfusion w_b). The objective is to quantify these effects for a fixed set of system parameters—namely, a_1 and t^* —whose values are listed in Table 4.1, and for varying perfusion w_b representative of different tissue types, shown in Table 4.2. The output injection gain K is tuned to maximize ξ .

Table 4.1. Values employed for the calculation of decay rates.

| Parameter | Symbol | Value |
|---------------------------|--------|-------------------|
| PBHE constant [-] | a_1 | 18.992 |
| Output injection gain [-] | K | $\frac{\pi^2}{4}$ |
| Time span [s] | t^* | 1800 |

Table 4.2. Perfusion rates w_b for different tissues (From [37]).

| Tissue | w_b [s^{-1}] | Tissue | w_b [s^{-1}] |
|--------|---------------------------|--------|---------------------------|
| Bone | $6.3e-5$ | Tumor | $1.7e-3$ |
| Fat | $1.2e-3$ | Muscle | $3.3e-3$ |

For the case of known perfusion w_b , the decay rate Λ [s^{-1}], defined in Eq. (4.7), satisfies:

$$\Lambda \geq 2 \left(\frac{\xi}{a_1} + w_b t^* \right).$$

In contrast, when w_b is uncertain and the observer uses a guessed value \tilde{w}_b , the decay rate Ξ [s^{-1}], defined in Eq. (4.10), satisfies:

$$\Xi \geq \frac{1}{2} \left(\frac{\xi}{a_1} + \tilde{w}_b t^* \right),$$

and the corresponding upper bound on the estimation error e_θ is given by:

$$\|e_\theta\|_{L^2}^2 \leq 4c_0,$$

where the bounding constant c_0 is

$$c_0 = \left(\frac{t^* |\tilde{w}_b - w_b|}{\frac{\xi}{a_1} + \tilde{w}_b t^*} \right)^2.$$

Table 4.3 reports the minimum decay rates Ξ and Λ across all combinations of system and observer tissue types, using perfusion values from Table 4.2. Diagonal entries correspond to the case when the observer uses the correct perfusion value. Off-diagonal values indicate performance under mismatch.

Table 4.3. Minimum decay rates Ξ, Λ [s^{-1}] for different perfusion rates. Columns: observers, rows: systems.

| System \ Observers | Bone | Fat | Tumor | Muscle |
|--------------------|-------|-------|-------|--------|
| System | Bone | Fat | Tumor | Muscle |
| Bone | 0.486 | 1.151 | 1.551 | 3.037 |
| Fat | 0.121 | 4.604 | 1.551 | 3.037 |
| Tumor | 0.121 | 1.151 | 6.203 | 3.037 |
| Muscle | 0.121 | 1.151 | 1.551 | 12.15 |

Table 4.4 presents the corresponding error bounds via the constant c_0 , quantifying the sensitivity to mismatch in the assumed perfusion rate. Note that the error is zero on the diagonal (perfect knowledge), and increases as the difference between w_b and \tilde{w}_b grows.

Table 4.4. Bounding value error c_0 for different perfusion rates. Columns: observers, rows: systems.

| System \ Observers | Bone | Fat | Tumor | Muscle |
|--------------------|----------|----------|----------|----------|
| System | Bone | Fat | Tumor | Muscle |
| Bone | 0 | $4.4e-4$ | $6.2e-4$ | $1.1e-3$ |
| Fat | $1.4e-3$ | 0 | $4.8e-5$ | $4.5e-4$ |
| Tumor | $2.7e-3$ | $6.6e-5$ | 0 | $2.8e-4$ |
| Muscle | $1.1e-2$ | $1.5e-3$ | $6.7e-4$ | 0 |

It is evident from Table 4.2 that perfusion rates span several orders of magnitude—from as low as $6.3e-5$ [s^{-1}] in bone tissue to over $3.3e-3$ [s^{-1}] in muscle. This high variability translates directly into large differences in both decay rates (Table 4.3) and error bounds (Table 4.4). To ensure robust performance across all tissue types, an appropriate sampling strategy must be developed to cover the entire range of physiologically plausible perfusion values. This would allow for the design of observers that are resilient to the full spectrum of model uncertainty. Moreover, the small perfusion values associated with some tissues (e.g., bone) and the bounded nature of ξ inherently limit achievable decay rates in those cases. Consequently, careful initialization of the observer becomes crucial.

4.4 Multiple-model adaptive scheme

Based on Corollary 4.3.1, and given the boundedness of perfusion w_b , the objective is to implement a Multiple-model Adaptive Observer, consisting of an adaptive combination of observers, each one with a different guess for the perfusion value. Although the parameter w_b is generally unknown, it is reasonable to assume the knowledge of an admissible range $W = [w_b^{\min}, w_b^{\max}]$ with $w_b \in W$. Bearing this in mind, a set of fixed values $w_b^j \geq 0$, $j = 1, \dots, N$ for some $N \in \mathbb{N}$ is considered, with

$$\begin{aligned} w_b^1 &< w_b^2 < \dots < w_b^N \\ w_b^{\min} &\leq w_b^1, \quad w_b^N \leq w_b^{\max}. \end{aligned}$$

Accordingly, let us define a family of observers $\hat{\theta}^{(j)}(X, \tau)$ with the structure (4.8)-(4.2) and with the choice $\tilde{w}_b = w_b^j$, that is

$$\begin{aligned} a_1 \partial_\tau \hat{\theta}^{(j)} &= \partial_{XX} \hat{\theta}^{(j)} - a_2 w_b^j \hat{\theta}^{(j)} + \bar{P}(\tau) a_3 \exp(-a_4 X) \\ \partial_X \hat{\theta}^{(j)}(0, \tau) &= g_0 + K(\hat{y}_j(\tau) - y_2(\tau)) \\ \hat{\theta}^{(j)}(1, \tau) &= y_1(\tau), \end{aligned}$$

being $\hat{y}_j(\tau) = \hat{\theta}^{(j)}(0, \tau)$ the j -th observer's superficial estimate. It is worth noticing that, for each observer within this family, an ultimate bound for the corresponding estimation error is provided by Corollary 4.3.1. Following the approach proposed in [30], the idea is then to introduce an overall estimator

$$\hat{\theta}^\dagger(X, \tau) = p_1(\tau) \hat{\theta}^{(1)}(X, \tau) + p_2(\tau) \hat{\theta}^{(2)}(X, \tau) + \dots + p_N(\tau) \hat{\theta}^{(N)}(X, \tau), \quad (4.11)$$

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

$$\dot{p}_j(\tau) = -\lambda \left(1 - \frac{e^{-\mu_j(\tau)}}{\sum_{\ell=1}^N p_\ell(\tau) e^{-\mu_\ell(\tau)}} \right) p_j(\tau) \quad j = 1, \dots, N, \quad (4.12)$$

where $\mu_j(\tau) := |y_2(\tau) - \hat{y}_j(\tau)|$ is the absolute output error and $\lambda > 0$ is the adaptive gain.

Whenever one of these errors is ultimately smaller than the others, with

$$\lim_{\tau \rightarrow +\infty} \sup (\mu_j^*(\tau) - \mu_\ell(\tau)) < 0 \quad \forall \ell \neq j^*,$$

one may expect the convergence of the weights to the values

$$\lim_{\tau \rightarrow \infty} p_{j^*}(\tau) = 1, \quad \lim_{\tau \rightarrow \infty} p_\ell(\tau) = 0 \quad \forall \ell \neq j^*, \quad (4.13)$$

so that (4.11) reduces to $\hat{\theta}^\dagger(X, \tau) = p_{j^*}(\tau) \theta^{(j^*)}(X, \tau)$ (see [30, Theorem 1]). This means that $\hat{\theta}^{(j^*)}(X, \tau)$ results being the *best* observer among the considered family $\{\hat{\theta}^{(j)}(X, \tau)\}_{j=1,\dots,N}$, with respect to the inherent output error function.

We name Eq. (4.12) the DCWG. The overall observer (4.11) corresponds to a weighted average of the individual observers, whose weights adapt based on the size of the associated output errors. A scheme of the MM-Obs is given in Figure 4.2.

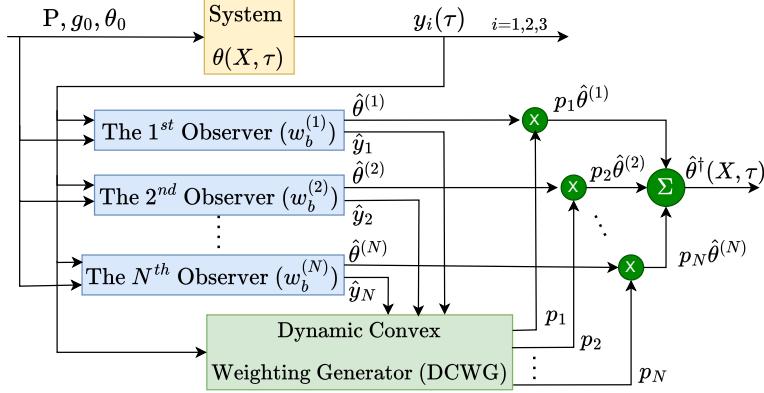


Figure 4.2. Scheme of the MM-Obs. The System evolves from a certain initial temperature distribution θ_0 due to the EM power given P and the heat removed by the water bolus g_0 . The boundary measurements of the system y_i feed the family of observers $\hat{\theta}^{(j)}$, whose superficial temperature estimate \hat{y}_j serves the DCWG to update the dynamic convex weights distribution. Finally, the MM-Obs $\hat{\theta}^\dagger$ is obtained as a weighted average of each observer's prediction.

Remark 4.4.1 Invoking again Corollary 4.3.1, we can notice that the finer the gridding of W is (or, equivalently, the larger the number of models N is), the smaller the ultimate set \mathcal{E} associated to the best observer $\hat{\theta}^{(j)}$ gets.

Remark 4.4.2 [Implementation issues] The weights $p_j(\tau)$ are naturally governed by the ODEs in (4.12). However, it can be observed that when using a built-in PDE solver such as `pdepe` in MATLAB, one may encounter some issues with getting in real time the output feedback from the PDEs' solutions needed to implement the ODEs. To overcome the problem, one may use a partial differential version of (4.12), obtained by introducing an artificial dependency on the spatial variable, and which delivers equivalent solutions. This can be done, for example, by defining $p_j(X, \tau)$, $j = 1, \dots, N$, as the solutions to the following system of nonlinear parabolic PDEs with homogeneous Neumann conditions

$$\begin{aligned}\partial_\tau p_j(X, \tau) &= \partial_{XX} p_j(X, \tau) - \lambda \left(1 - \frac{e^{-\mu_j(\tau)}}{\sum_{\ell=1}^N p_\ell(X, \tau) e^{-\mu_\ell(\tau)}} \right) \quad j = 1, \dots, N \\ \partial_X p_j(0, \tau) &= \partial_X p_j(1, \tau) = 0\end{aligned}$$

and initialization $p_j(X, 0)$ independent of X for any $j = 1, \dots, N$.

Remark 4.4.3 The following developments are meant to provide a family of functions satisfying the observer compatibility conditions in a physically interpretable representation for the initial temperature distribution. Given the compatibility conditions for the observer defined in Eq. 4.4:

$$\partial_X \hat{\theta}|_{X=0}(0) = g_0(0) + K(\hat{y}(0) - y_2(0)) \quad \wedge \quad \hat{\theta}_0^{(j)}(1) = y_1(0)$$

consider the following IC

$$\hat{\theta}_0^{(j)}(X) = c_1 X^2 + c_2 X + c_3 \quad X \in [0, 1] \quad (4.14)$$

The constants c_1 , c_2 and c_3 are defined following the compatibility with boundary conditions, and imposing the identity $\hat{\theta}_0^{(j)}(0) = \theta_0(0) = y_2(0)$:

$$\begin{aligned}\hat{\theta}_0^{(j)}(1) &= y_1(0) \\ \hat{\theta}_0^{(j)}(0) &= y_2(0) \\ \partial_X \hat{\theta}|_{X=0}(0) &= g_0(0)\end{aligned}$$

Substituting the expression in Eq. (4.14):

$$\begin{aligned}c_1 + c_2 + c_3 &= y_1(0) \\ c_3 &= y_2(0) \\ c_2 + c_3 &= g_0(0)\end{aligned}$$

The expression for the constants c_1 , c_2 and c_3 will be:

$$\begin{aligned}c_1 &= y_1(0) - g_0(0) \\ c_2 &= g_0(0) - y_2(0) \\ c_3 &= y_2(0).\end{aligned} \quad (4.15)$$

4.5 Conclusion

In this chapter, we provided the theoretical formulation for the MM-Obs of PBHE.

In the first part, an introduction to boundary control of PDEs was given, explaining its objectives and engineering applications along with the mathematical tools and logical steps for the theoretical formulation.

Then, these tools were applied to the uncertain parabolic reaction-diffusion PDE of the current use case. Boundary control of PBHE is achieved from the Robin-type control at the surface ($X = 0$, see Figure 4.1). We first designed an observer for the nominal case where all parameters are known, then explored the case of uncertain perfusion, providing a sensitivity analysis of the observer with respect to this parameter. In Section 4.3.3, a numerical analysis of the decay rates and error bounds of different observers is reported, demonstrating both the necessity for proper sampling across perfusion's wide variability range and the importance of adequate observer initialization.

Finally, the multiple-model setup is introduced following the approach proposed in [30]. The overall observer (4.11) corresponds to a weighted average of the individual observers, whose weights adapt based on the size of the associated output errors.

Chapter 5

PINNs-based implementation of the multiple-model observer

In this chapter, we propose to leverage the capabilities of Physics-Informed learning to solve the PBHE. In the context of HT superficial treatments, PINNs can be employed on two tasks, the HTP (offline phase) and feedback control of treatments (online phase).

For the offline phase, the currently-in-use solvers are employed only once, at the time of patient registration. All the treatments conducted in the following months are based on the patient’s map obtained on the registration day. To replace these solvers with a more flexible and fast DL-based solver, which is trained at the time of patient registration and then fine-tuned to accommodate for changes in the patient’s anatomy at each treatment, would be beneficial for the optimization of the HTP.

In this work, we focused on the online phase, employing Physics-Informed learning for the noninvasive, real-time temperature assessment at the target. This is achieved observing bio-heat transfer from boundary measurements, in the presence of uncertain properties of the domain, and then reconstructing the entire temperature distribution inside the patient, as theoretically formalized in Chapter 4. Our PINN-based approach leverages observers trained over a range of input variations, including spatial and temporal variables, as well as surface measurements. This training strategy allows the observers to generalize effectively across different scenarios, offering greater flexibility and adaptability compared to traditional numerical methods.

Classical numerical solvers, such as those implemented in MATLAB, can incur limitations, particularly in complex, high-dimensional problems, as it will be shown in Section 5.3.4. In contrast, PINNs integrate physical laws directly into the learning process, mitigating some of these constraints. While PINNs require significant computational resources during the offline training phase, once trained, they enable real-time solutions to the underlying PDEs. Moreover, their prediction is continuous in the spatiotemporal domain, and does not require a solving grid. Although solving a one-dimensional PBHE using MATLAB is computationally manageable, extending the problem to two and three dimensions becomes significantly more challenging—an area where PINNs can provide a substantial advantage.

The simulations presented in this chapter serve a dual purpose: first, to validate the accuracy of PINNs by comparing their performance with classical solvers such as

MATLAB (Section 5.3.3); and second, to demonstrate how PINNs can overcome the limitations of conventional numerical methods (Section 5.3.4).¹ Also, in Section 5.2 we present the inverse problem approach, in which the PINN is able to optimize an uncertain PDE parameter during training, provide some additional data of the solution. This can be useful when characterizing an unknown system (e.g., identification of perfusion bounds in experimental setup).

Our implementation of PINNs is carried out using the DeepXDE Domain-Specific Language (DSL)², and we will introduce and test the Neural Bio-Heat System (NBHS) as the PINN trained to solve the direct PBHE, and the Neural Bio-Heat Observer (NBHO) as the PINN that solves the observer's PDE problem incorporating boundary measurements as inputs to the network in real-time. PINNs have been trained on an Intel(R) Core(TM) i9-10900K CPU @ 3.70GHz with NVIDIA GeForce RTX 3090 GPU.

5.1 Direct problem

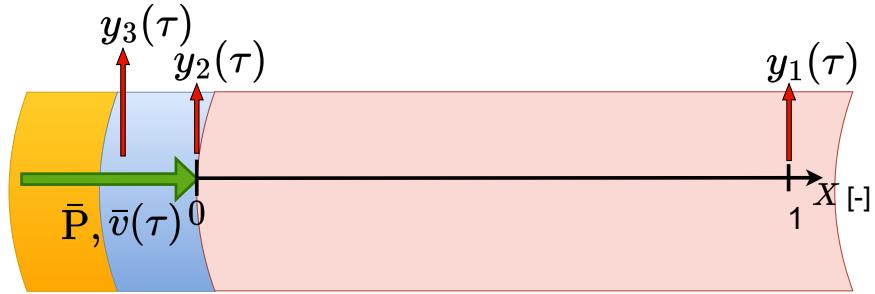


Figure 5.1. Scheme of the considered scaled setup. Consider an ideal needle that enters perpendicularly from the patient's surface, the left boundary being the surface and the right one being the in-depth bound.

Simulations are conducted to model the cooling of a phantom, which is a simplified representation of a human body. The 1D domain of length $L_0 = 0.07$ [m], perpendicular to the phantom's upper surface and centered on the latter, is considered. It is subject to a Robin-type boundary condition at the surface, representing convection heat transfer with a fluid at a scaled temperature y_3 , and a Dirichlet condition on the in-depth boundary, as reported in Figure 5.1. The partial differential equation governs the system:

$$a_1 \partial_\tau \theta = \partial_{XX} \theta - w_b a_2 \theta + \bar{P}(\tau) a_3 \exp(-a_4 X) \quad (5.1)$$

defined for $(X, \tau) \in [0, 1] \times [0, +\infty)$. The dimensionless temperature $\theta = \frac{T' - T}{\Delta T}$, with $T' = T - T_{\min}$ and $\Delta T = T_{\max} - T_{\min}$. The BCs are

$$\begin{aligned} \partial_X \theta|_{X=0} &= -a_5(y_3(\tau) - y_2(\tau)) \quad \tau \geq 0 \\ \theta(1, \tau) &= y_1(\tau) \quad \tau \geq 0. \end{aligned} \quad (5.2)$$

¹The results contributed to the works [12, 9].

²The code is available at: GitHub Repository.

The measurements $y_1(\tau)$ and $y_3(\tau)$ are assumed to remain constant and equal to their initial values, $y_1(0)$ and $y_3(0)$, while $y_2(\tau) = \theta(0, \tau)$. The coefficients a_i , $i = 1, \dots, 5$ are defined in Eq. (2.8) and their values are given in Table 5.2.

The system is initialized with

$$\theta_0(X) = b_1 X^3 + b_2 X^2 + b_3 X + b_4 \quad X \in [0, 1], \quad (5.3)$$

The coefficients b_i , $i = 1, \dots, 4$ satisfy the conditions of Eq. (2.11) interpolating $y_1(0)$, $y_2(0)$, $y_3(0)$ and θ_{gt} of Table 5.1.

Table 5.1. Scaled initial values of, respectively, in-depth boundary measurement, superficial boundary measurement, fluid temperature, internal measurement at $X_{\text{gt}} = 0.14$.

| $y_1(0)$ | $y_2(0)$ | $y_3(0)$ | θ_{gt} |
|----------|----------|----------|----------------------|
| 0.0 | 0.82857 | 0.0 | 0.95238 |

The coefficients a_i , $i = 1, \dots, 5$ depend on tissue's properties and thus are considered constant for a given setup, while constants b_i , $i = 1, \dots, 4$ depend on initial temperatures.

Table 5.2. Properties of the simulation.

| Parameter | Symbol | Value |
|--------------------------------------------|------------|-----------|
| PBHE constant [-] | a_1 | $1.9e1$ |
| PBHE constant [-] | a_2 | $3.4e4$ |
| PBHE constant [-] | a_3 | 0.0 |
| PBHE constant [-] | a_4 | 1.9 |
| PBHE constant [-] | a_5 | 1.2 |
| IC system constant [-] | b_1 | -1.3 |
| IC system constant [-] | b_2 | $-5.2e-1$ |
| IC system constant [-] | b_3 | $9.7e-1$ |
| IC system constant [-] | b_4 | $8.3e-1$ |
| Power [W] | P | 0.0 |
| Minimum temperature [$^{\circ}\text{C}$] | T_{\min} | 21.5 |
| Maximum temperature [$^{\circ}\text{C}$] | T_{\max} | 32.0 |
| Blood perfusion rate [s^{-1}] | w_b | $5.0e-4$ |

5.1.1 Neural Bio-Heat System (NBHS)

The NBHS is a PINN trained to solve the PBHE for the direct problem, where all equation parameters are known and the goal is to determine the temperature distribution. It takes as input the coordinates X, τ and returns the prediction $\theta(X, \tau)$. To ensure physical consistency, IC $\theta_0((X))$ and in-depth BC ($\theta(1, \tau) = 0$) are enforced as hard constraints through an output transformation ansatz:

$$\zeta : \tilde{y} = \tau \cdot (1 - X) \cdot y + \theta_0(X)$$

This transformation guarantees that the solution satisfies the in-depth BC and IC by construction, allowing the network to focus on learning the governing physics and the superficial BC. A scheme of the NBHS architecture is given in Figure 5.2, illustrating both the trainable neural network with its output transform ζ and the residual network that enforces the governing equation constraints.

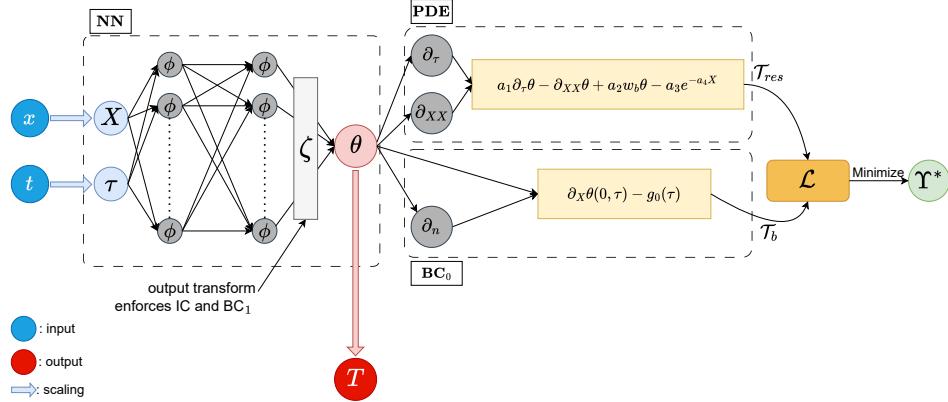


Figure 5.2. Scheme of NBHS. On the left, the trainable neural network with the output transform ζ . On the right, the residual network with the governing equation and soft constraints loss terms.

The NBHS is trained for 3000 iterations using the Adam optimizer and then L-BFGS until convergence to minimize the physics-informed loss function. A detailed progression of the training losses is presented in Figure 5.3, while HPs of NBHS are summarized in Table 5.3. The problem defined in Eq. (5.1), Eq. (5.2), and Eq. (5.3) is solved with MATLAB on a 100×100 grid with $(X, \tau) \in [0, 1] \times [0, 1]$. This dataset serves as a ground truth towards which compare NBHS prediction. The results are given in Figure 5.4, demonstrating excellent agreement between our physics-informed approach and classical numerical methods.

To evaluate the robustness and generalizability of this approach, NBHS was tested with different biological media (bone, fat, and muscle tissue) and perfusion fluids (water and blood). The results, reported in Figure 5.5, demonstrate that NBHS maintains consistent accuracy across these physiologically relevant variations, making it suitable for diverse clinical applications. Further studies on NBHS's hyperparameter HP optimization can be found in Appendix A.

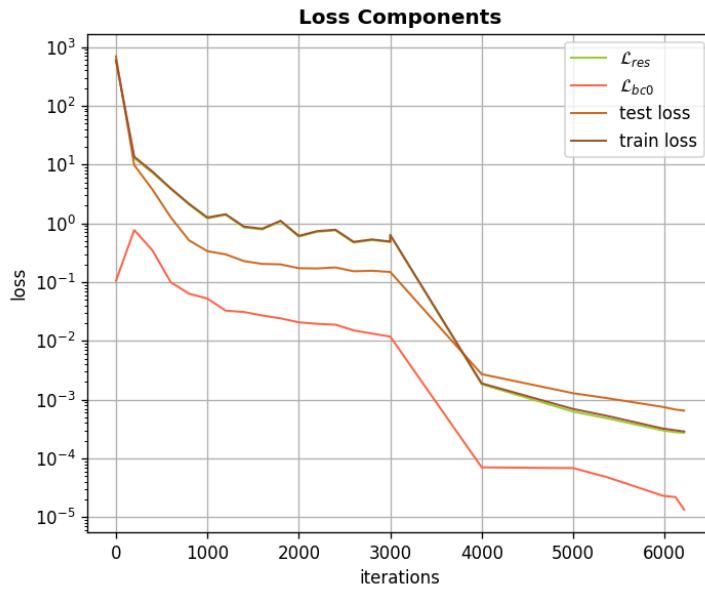
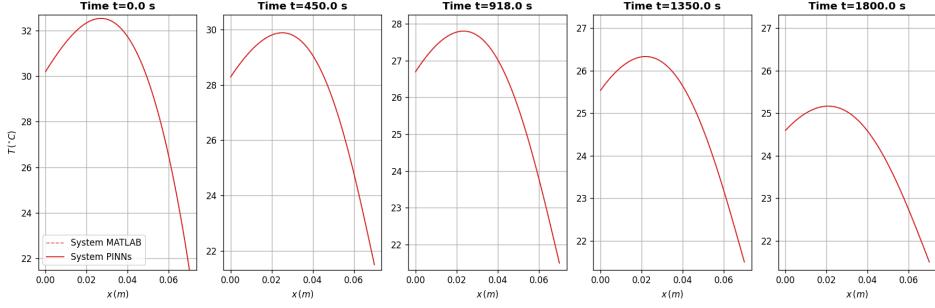


Figure 5.3. Training error for the Neural Bio-Heat System (NBHS).

Table 5.3. NBHS Configuration.

| Parameter | Symbol | Value |
|----------------------------------------|------------------|---------------|
| Activation function | Φ | tanh |
| Initial Weights Regularization | IWR | False |
| Initialization | Υ_0 | Glorot normal |
| Learning rate [-] | η | 0.001 |
| Depth [-] | $L - 1$ | 4 |
| Width [-] | N | 50 |
| Number of residual sampling points [-] | n_{res} | 500 |
| Number of boundary sampling points [-] | n_b | 200 |
| Number of test points [-] | n_{test} | 1000 |
| Resampling | RR | True |
| Weight of residual loss [-] | w _{res} | 1 |
| Weight of boundary loss [-] | w _{bco} | 1 |
| Number of inputs [-] | n_{ins} | 2 |
| Number of anchor points [-] | n_{anc} | 0 |



(a) NBHS vs MATLAB at different instants.

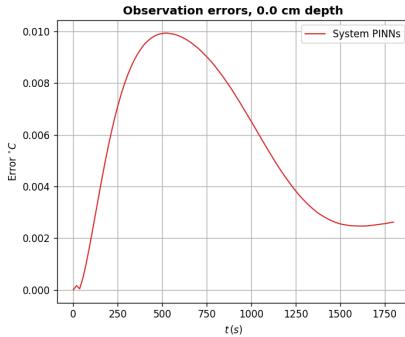
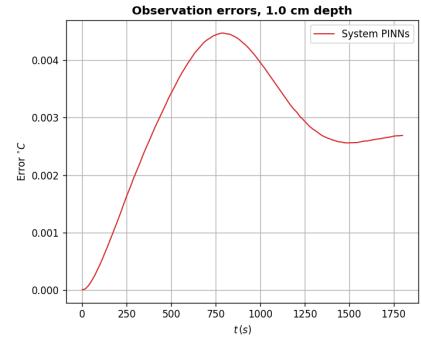
(b) $X = 0$.(c) $X = X_{gt}$.

Figure 5.4. Comparison of NBHS and MATLAB results. (Top) Overall trend at different instants. (Bottom) Detailed view at specific values of X .

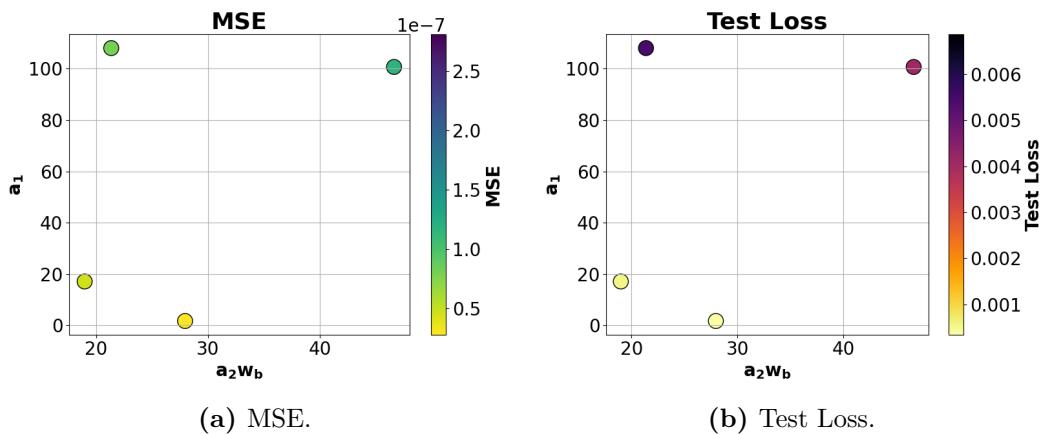


Figure 5.5. Summary of NBHS performance with different PBHE coefficients.

5.2 Inverse problem

As stated in Section 3.8.1, PINNs can solve the inverse problem. An inverse problem involves determining unknown parameters ψ from observed data. Unlike direct problems, where the goal is to solve known equations for given inputs, inverse problems aim to infer missing information by leveraging both data and the governing physical laws encoded in the PINN. This is achieved by minimizing a loss function that incorporates measurement discrepancies and residuals from the underlying differential equations.

For the present PBHE application, perfusion w_b is formulated as a trainable parameter within the NBHS framework. Starting with an initialization value, NBHS iteratively optimizes this parameter to best fit the observed data. The observed data is extracted from the MATLAB-generated (100×100) space-time grid of the PBHE solution generated for the direct problem in the previous section. Specifically, the NBHS is provided with measurements from a $(n_{\text{anc}} \times 100)$ grid, where n_{anc} represents the number of spatial measuring points. First, the inverse problem has been investigated in two configurations: using only boundary measurements (Figure 5.7a, $n_{\text{anc}}=2$ at $X=0$ and $X=1$), and adding an internal measurement at $X_{\text{gt}}=0.14$ (Figure 5.7b, $n_{\text{anc}}=3$).

In all cases, perfusion w_b was initialized with $w_b^0 = 6.3e-5$ [s^{-1}] and NBHS was trained for 10000 iterations using the Adam optimizer with a measurement loss weight of $w_i = 20$. To improve convergence, a warmstart (\mathcal{W}) approach [4] was implemented, using the results of previous simulations as initialization values for subsequent runs (Figure 5.7c, $n_{\text{anc}}=2$ and Figure 5.7d, $n_{\text{anc}}=3$). The methodology was further refined by training the $n_{\text{anc}}=3$ configuration with 3000 iterations of Adam followed by L-BFGS optimization (Figure 5.6). This combined optimizer approach yielded estimates closer to the actual perfusion value: the Table 5.4 reports that the best performance was achieved with 3 measurement points and LBFGS optimization ($\text{MSE} = 2.1e-7$) while using just 2 measurement points with warmstart (\mathcal{W}) resulted in the highest error ($\text{MSE} = 3.3e-3$).

These results demonstrate that NBHS can consistently retrieve perfusion values even with limited spatial measurements. However, this approach is not suitable for real-time w_b evaluation since the unknown parameters ψ are adjusted during the offline training phase.

Table 5.4. Performance metrics for inverse problem comparing MSE values across different configurations of measurement points (n_{anc}), initialization strategies (\mathcal{W}), and optimizers. The table reports the final recovered perfusion values w_b^f , while the unknown value is $w_b^{\text{sys}} = 5.0e-4$ [s^{-1}].

| n_{anc} | \mathcal{W} | L-BFGS | MSE | w_b^f [s^{-1}] |
|------------------|---------------|--------|----------|-----------------------------|
| 3 | false | true | $2.1e-7$ | $5.0e-4$ |
| 3 | true | false | $1.2e-4$ | $4.4e-4$ |
| 2 | false | false | $1.4e-4$ | $5.7e-4$ |
| 3 | false | false | $3.3e-4$ | $4.5e-4$ |
| 2 | true | false | $3.3e-3$ | $3.7e-4$ |

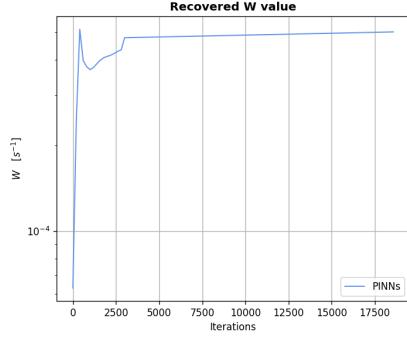


Figure 5.6. The evolution of parameter w_b during the combined optimizer training approach of NBHS in inverse problem, with three measurement points ($n_{\text{anc}}=3$).

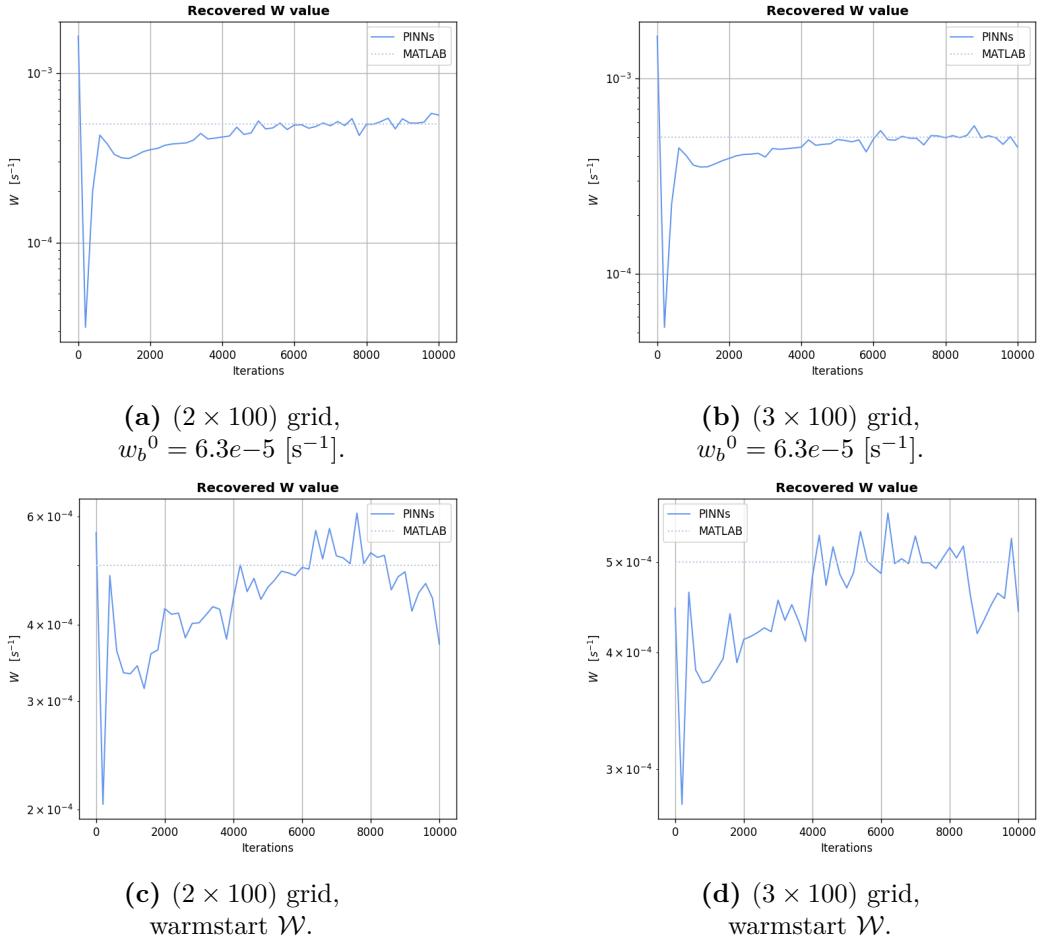


Figure 5.7. Evolution of parameter w_b during the training of NBHS in the inverse problem.
 (2 x 100) grid: only two boundary measurements are used.
 (3 x 100) grid: one internal measurement is added at $X_{\text{gt}} = 0.14$, corresponding to a depth of 1 [cm].
 Top row: initialized with $w_b^0 = 6.3e-5$ [s^{-1}].
 Bottom row: initialized using warmstart \mathcal{W} [4].

5.3 Adaptive estimation of PBHE

In the scheme of Figure 4.2, each observer is now implemented with PINN. With this method, after a period of training, the solution to the governing PDE is predicted in a few milliseconds, addressing the limitations associated with solving coupled systems of PDEs (observers) and the ODE of the DCWG in traditional numerical methods. Modifications are introduced to the PINN architecture, expanding the input space to incorporate system measurements, which serve as boundary conditions.

5.3.1 Neural Bio-Heat Observer (NBHO)

In PINN, the input can be in a physical space or a parametric space indifferently. The network is merely a surrogate of the solution. Hence, it is possible to extend the network's domain to n-dimensional geometry. The NBHO (Figure 5.8) is composed of four inputs:

$$\mathbf{x} = (X, y_1(\tau), y_2(\tau), \tau),$$

where X is the scaled space coordinate, $y_1(\tau)$ is the in-depth measurement, $y_2(\tau)$ is the superficial measurement, and τ is the time coordinate. The output $y_3(\tau)$ (temperature of the fluid above the domain) is considered constant and equal to 0 (free convection with air at T_{\min}). Initial condition and boundary condition at $X = 1$ are enforced as hard constraints, i.e., transforming network's output y :

$$\zeta : \tilde{y} = \tau \cdot (1 - X) \cdot y + \theta_0(X) + y_1(\tau) - y_1(0).$$

The NBHO undergoes the same training procedure as NBHS. The detail of the training losses for the NBHO 3 is given in Figure 5.9.

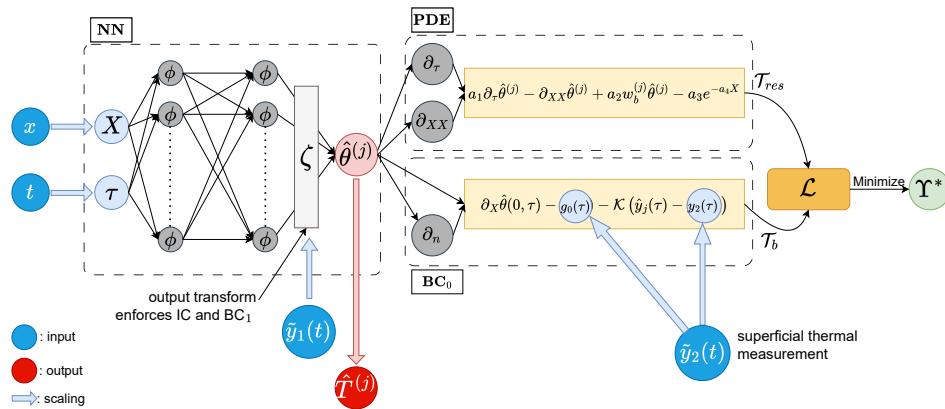


Figure 5.8. Scheme of NBHO. Additional inputs are in-depth temperature y_1 , which is needed for the output transform in the trainable network, and superficial temperature y_2 that constitutes the superficial boundary condition in the residual network.

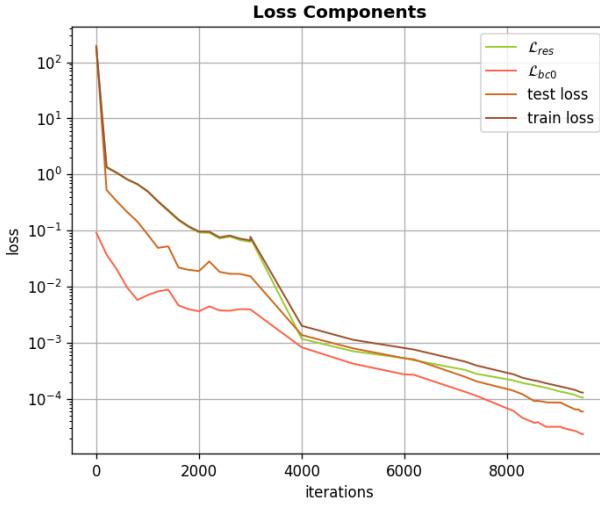


Figure 5.9. Training error for the Neural Bio-Heat Observer (NBHO), Observer 3.

5.3.2 Model

The theory formulated in Section 4.4 is reported here for convenience. Each observer $\hat{\theta}^{(j)}(X, \tau)$ is governed by the partial differential equation

$$a_1 \partial_\tau \hat{\theta}^{(j)} = \partial_X \hat{\theta}^{(j)} - \hat{w}_b^j a_2 \hat{\theta}^{(j)} + \bar{P}(\tau) a_3 \exp(-a_4 X),$$

defined for $(X, \tau) \in [0, 1] \times [0, +\infty)$. The BCs are

$$\begin{aligned} \partial_X \hat{\theta}^{(j)}|_{X=0} &= -a_5(y_3(\tau) - y_2(\tau)) + K(\hat{y}_j(\tau) - y_2(\tau)) \quad \tau \geq 0, \\ \hat{\theta}^{(j)}(1, \tau) &= y_1(\tau) \quad \tau \geq 0. \end{aligned}$$

With $\hat{y}_j(\tau) = \hat{\theta}^{(j)}(0, \tau)$. The observer is initialized with

$$\hat{\theta}_0^{(j)}(X) = c_1 X^2 + c_2 X + c_3 \quad X \in [0, 1].$$

The coefficients c_i , $i = 1, \dots, 3$ are defined in Eq. (4.15) and their values are given in Table 5.6. The overall estimator in Eq. (4.11):

$$\hat{\theta}^\dagger(X, \tau) = p_1(\tau)\theta^{(1)}(X, \tau) + p_2(\tau)\theta^{(2)}(X, \tau) + \dots + p_N(\tau)\theta^{(N)}(X, \tau)$$

is the dynamic convex combination of the observers $\hat{\theta}^{(j)}(X, \tau)$ with adaptive weights $p_j(\tau)$ being updated according to Eq. (4.12):

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

where $\mu_j(\tau) := |y_2(\tau) - \hat{y}_j(\tau)|$ is the absolute output error and $\lambda > 0$ is the adaptive gain.

The general observer (4.11) corresponds to a weighted average of individual observers, whose weights adapt based on the size of the associated output errors.

Choice of observers' perfusions

As shown in [35, Table I], when considering perfusion rate, biological tissues can be roughly divided into two groups: the ones with a high perfusion rate (ladder, kidney, muscle, tumor, and others) and bone, cartilage, and fat characterized by low perfusion rates. Moreover, as reported in [70, Table I], perfusion values can exhibit significant variation, particularly during HT treatment. For instance, perfusion in fat has been reported to fluctuate by a factor of 2, while in muscle, it can vary by up to an order of magnitude.

The proposed methodology uses a family of eight observers with perfusion values $w_b^{(j)}$ sampled on a logarithmic scale, from bone perfusion ($6.3e-5$ [s^{-1}]) to tumor perfusion ($1.7e-3$ [s^{-1}]), as detailed in Tab. 5.5. This sampling strategy is chosen to ensure a representative sampling of potential physiological perfusion scenarios.

Table 5.5. Values of \tilde{w}_b [s^{-1}] used in the simulations.

| | | | |
|-----------------------|----------|-----------------------|----------|
| $\tilde{w}_b^{(0)}$: | $6.3e-5$ | $\tilde{w}_b^{(1)}$: | $1.0e-4$ |
| $\tilde{w}_b^{(2)}$: | $1.6e-4$ | $\tilde{w}_b^{(3)}$: | $2.6e-4$ |
| $\tilde{w}_b^{(4)}$: | $4.1e-4$ | $\tilde{w}_b^{(5)}$: | $6.5e-4$ |
| $\tilde{w}_b^{(6)}$: | $1.0e-3$ | $\tilde{w}_b^{(7)}$: | $1.7e-3$ |

Table 5.6. Summary of $\hat{\theta}_0^{(j)}$ coefficients.

| c_1 | c_2 | c_3 | μ | λ |
|----------|---------|---------|---------|-----------|
| -1.79524 | 0.96667 | 0.82857 | $1.0e2$ | 1.0 |

5.3.3 Results

Temperature distributions computed with MATLAB implementation are reported in Figure 5.10, and the corresponding predictions with PINNs are depicted in Figure 5.11. A comparison between the error norm and the dynamic weights (DCWG) in two implementations is given in Figure 5.12.

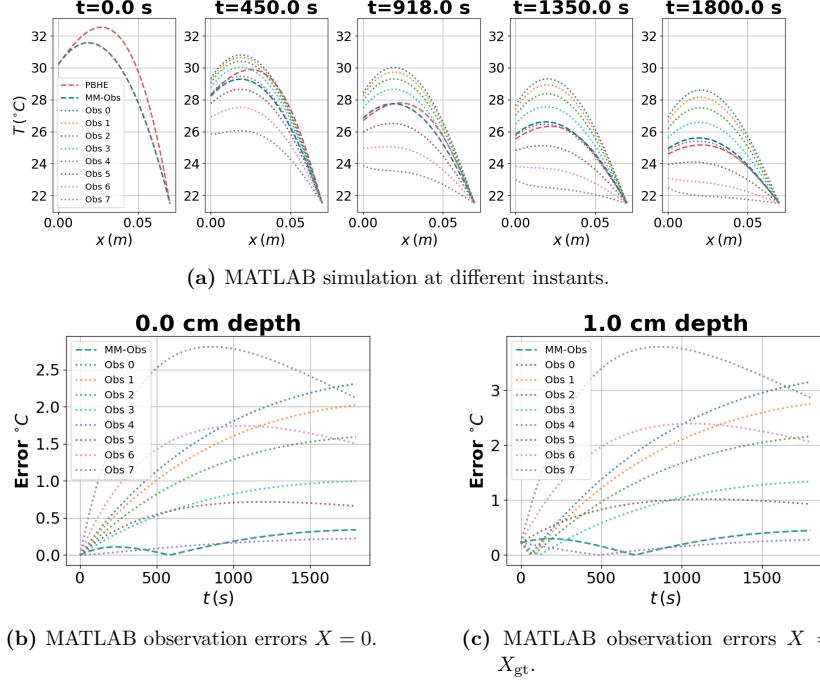


Figure 5.10. MATLAB simulation and observation errors at different instants.
 (Top) Temperature evolution over time. (Bottom) Observation errors at specific positions.

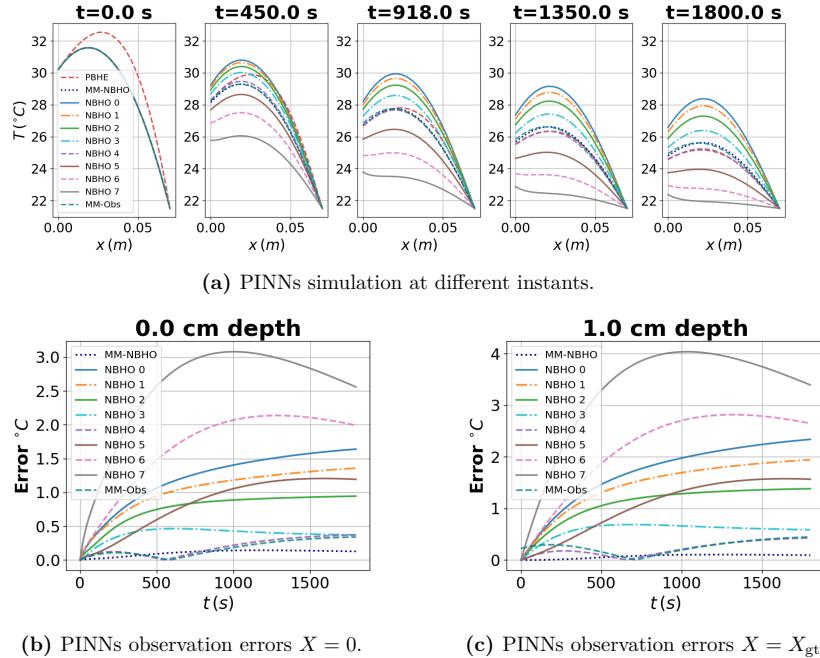


Figure 5.11. PINNs simulation and observation errors at different instants.
 (Top) Temperature evolution over time. (Bottom) Observation errors at specific positions.

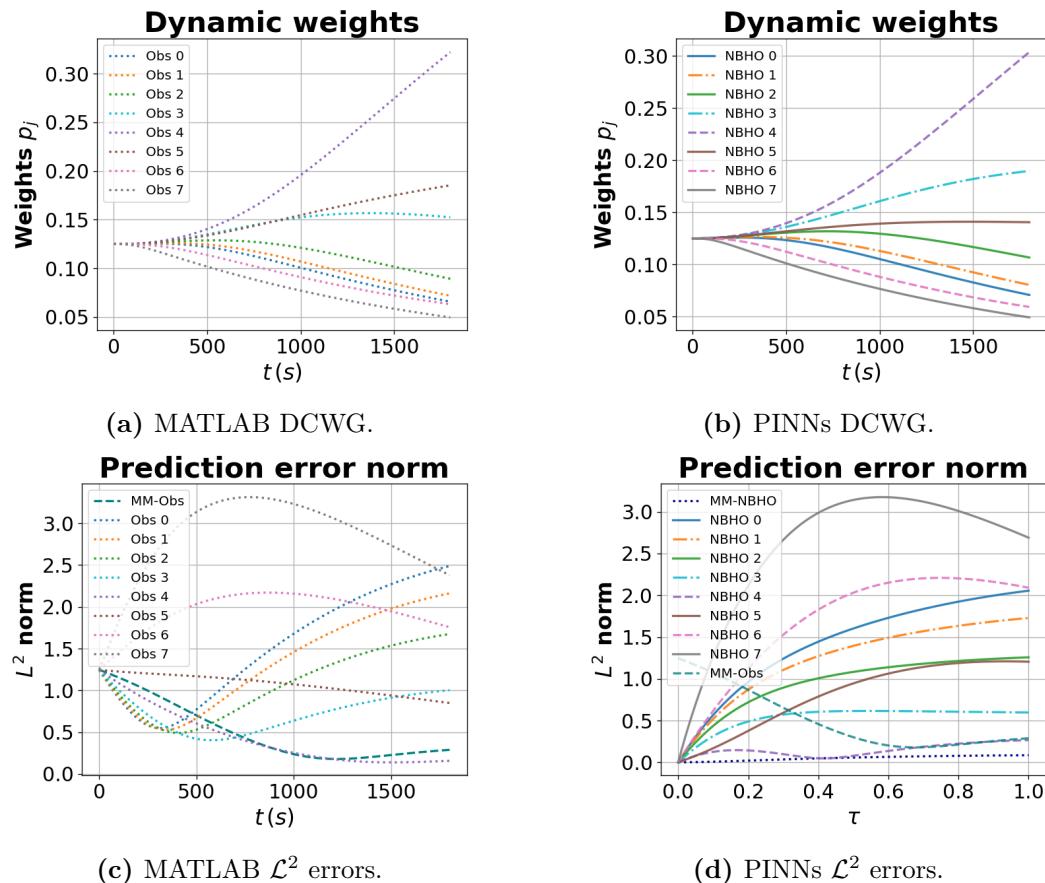


Figure 5.12. Comparison of \mathcal{L}^2 errors and DCWG output, MATLAB and PINNs.

5.3.4 Overcoming MATLAB limitations

As stated in Section 4.4, the overall observer corresponds to a weighted average of individual observers as in Eq. (4.11), whose weights are adjusted based on the size of the associated output errors. Whenever one of these errors is ultimately smaller than the others and the convergence of weights as in Eq. (4.13) is achieved, it is possible to identify the best observer $\hat{\theta}^*$ among the considered family. The perfusion of the best observer w_b^* will be the closer, among the observers' perfusion, to the perfusion of the system w_b .

When increasing the value of the parameter adaptive gain λ to accelerate the convergence of MM-Obs, MATLAB incurs numerical issues. These are probably related to the difficulty of integrating the system of PDE (PBHE of system and observers) and the ODE of the weights for the MM-Obs (see Rem. 4.4.2). At the same time, PINNs overcomes these thanks to the different approaches to solving PDE. In PINNs, the solution to the system of PDEs is straightforward. Figure 5.13 gives an overview of the result.

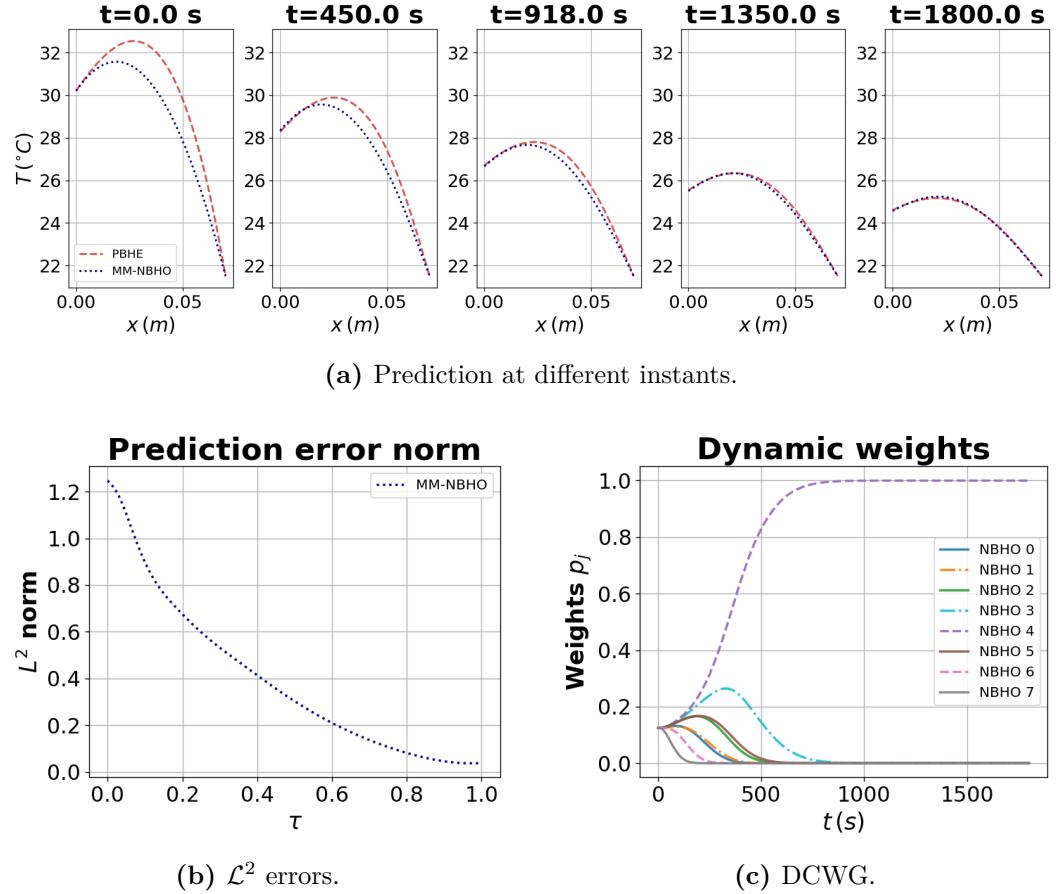


Figure 5.13. Increased $\lambda=100$ to converge to the best observer.

5.4 Conclusion

This chapter demonstrates the PINNs approach for the adaptive state estimation of PBHE within a numerical application.

The problem has been formulated in Chapter 4: we aim at evaluating the entire temperature distribution in a one-dimensional domain based only on boundary measurements, and in the presence of uncertain perfusion. To achieve this, an adaptive observer is designed for the boundary control of the PDE; then, a multiple-model adaptive setup is implemented as a family of observers, each one with a different arbitrary value for the perfusion, and the overall observer is a weighted average of the single observers, with weights adapted based on the observation error. The multiple model is represented by an ODE, while each observer follows the PBHE. In the numerical validation, limitations occur with classical numerical solvers as the integration of the system of PDEs and the ODE is not straightforward.

Implementing a NN instead of resolving the system of equations through a numerical solver entails differences in implementation and necessitates the selection and optimization of several parameters, but the proven ability of trained PINNs to expedite the resolution of PDEs and systems of ODE and PDEs has been demonstrated as a solution to the constraints typically encountered by classical numerical solvers when addressing infinite-dimensional problems. Moreover, this approach exhibits significant improvements in scalability and accuracy, as the same models can be applied to a range of different setups, and their predictions are continuous in the spatio-temporal domain.

Chapter 6

Application of Physics-informed multiple-model observer to a real case

In superficial HT, the effectiveness of the treatment can be assessed through the temperature distribution at the target volume which is named the Gross Tumor Volume (GTV).

Currently, the acquisition of such data necessitates the insertion of catheters into the patient's body or the use of natural cavities. Consequently, this approach yields multiple temperature readings that are discrete and distributed along a line, with their exact positions being difficult to ascertain. Furthermore, this method induces discomfort in patients and is frequently disregarded. The consequence is that superficial HT treatments are often conducted based on the treatment planning and on some measurements at the interface between the bolus and the patient skin, without any additional information about the temperature at the target.

The approach illustrated in the previous chapters, with an AI-oriented tool able to incorporate boundary measurements and produce in real-time a prediction of temperature distribution in the entire domain and in the presence of unknown parameters, could be beneficial for achieving of feedback of the treatment. This approach also solves limitations encountered in classical numerical solvers such as MATLAB. After a numerical validation of the approach, the experimental test of the MM-Obs implemented with PINNs was conducted at the laboratories of the Radiation Oncology department of Amsterdam UMC.

This chapter is structured as follows: first, we introduce the materials and methods utilized in this experiment, including heating instrumentation, a phantom, a water system to simulate perfusion, and a thermometry system. Second, we provide an overview of the model and main hypotheses of the experiment, as well as verification of key design criteria. Then, results are presented, followed by a critical analysis of the experimental setup and possible future developments.

6.1 Materials and methods

The objective is to model a HT treatment on a non-biological, heterogeneous medium represented by the phantom in Figure 6.1, with the mono-dimensional PBHE formulated in Section 2.3.1.

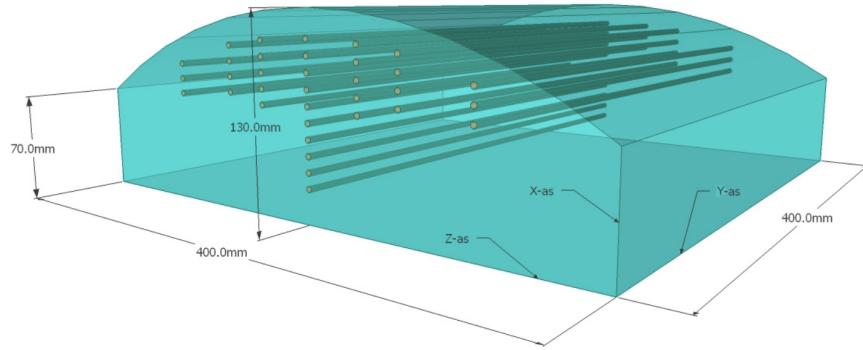


Figure 6.1. Slightly curved muscle-equivalent phantom (from [6]).

6.1.1 Preliminary considerations

The primary goal is to achieve a temperature profile that exhibits a peak within the domain ($X = X_{gt}$), as observed in actual treatments where the GTV lies 2-4 [cm] depth. To accomplish this, it is essential to balance the heating effect of the applicator with the cooling effect of the water bolus.

Preliminary tests were conducted using the ALBA ON400D, leveraging its integrated bolus temperature control system. This system is optimized for clinical treatments, where the typical operating temperature ranges from 40 to 45 [$^{\circ}\text{C}$], and the minimum allowable bolus temperature by the system is 30 [$^{\circ}\text{C}$]. Consequently, in order to induce a cooling effect, the phantom's temperature needed to exceed this threshold. The initial temperature of the phantom, however, was 21.5 [$^{\circ}\text{C}$], and not 37 [$^{\circ}\text{C}$] as the basal temperature of patients. As a result, substantial heating was required, and the cooling action of the bolus proved insufficient to counterbalance the applied heating.

This, together with the absence of the stabilization term represented by perfusion, prevented the system from reaching a steady-state condition. Also, a sharp rise in each of the TC temperature histories was noticed when they approached 29-30 [$^{\circ}\text{C}$]. This effect can be attributed to the properties of the thermocouples TC (Copper-Constantan) which have a reference temperature of around 29 [$^{\circ}\text{C}$]. As the measured temperature neared this value, sensor inaccuracies likely emerged. Moreover, since these TC are calibrated for clinical applications (from 35 to 45 [$^{\circ}\text{C}$]), their accuracy may be affected outside this interval.

To address these challenges, the following adjustments were implemented:

- **Translation of temperature range:** being $T_{\min} = 21.5 \text{ }^{\circ}\text{C}$, T_{\max} was set to $32 \text{ }^{\circ}\text{C}$, to ensure a meaningful temperature difference ΔT ;
- **Manual control of bolus temperature:** since the built-in system does not allow temperatures below $30 \text{ }^{\circ}\text{C}$, direct intervention was required to maintain the desired cooling conditions;
- **Introduction of a perfusion surrogate:** to facilitate the attainment of a steady state, a water flow system was introduced by circulating water through one or two catheters embedded within the phantom. The catheters were connected to a bucket at an adjustable height, allowing control over the perfusion effect.

The experimental setup is depicted in Figure 6.2.

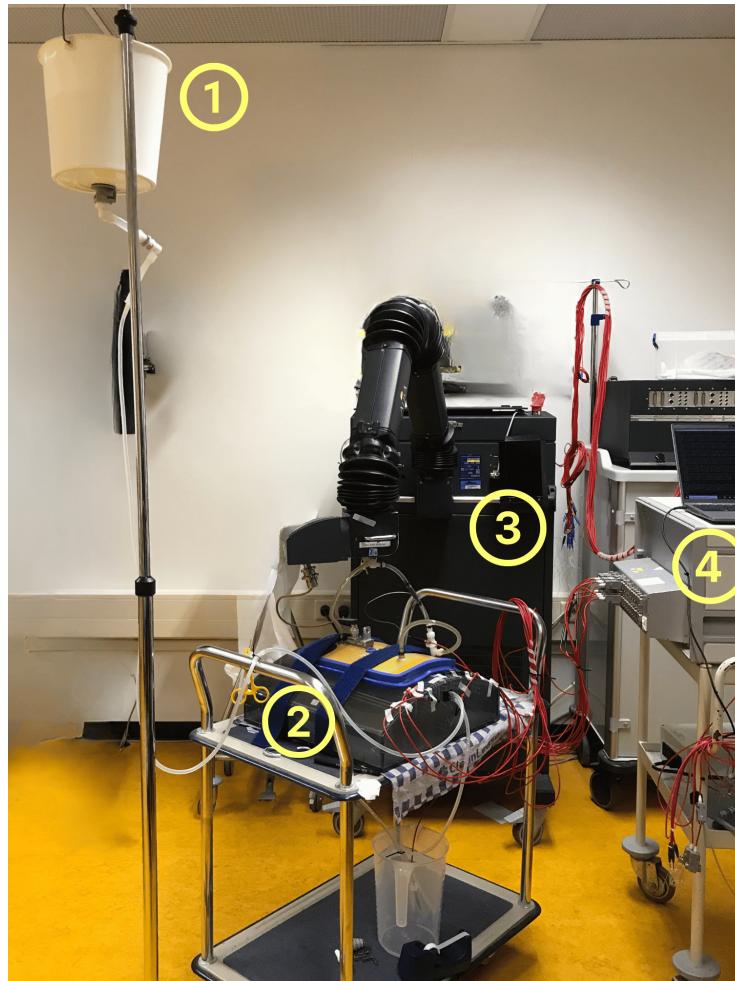


Figure 6.2. The Experimental Setup: bucket with adjustable height (1), slightly curved muscle-equivalent phantom (2), ALBA on4000D and 3H applicator (3), Hyp3 in-house build thermometry system (4).

6.1.2 Muscle-equivalent phantom and position of the probes

The muscle-equivalent phantom is a parallelepiped with a single curved upper surface, minimum height 7 [cm], maximum height 13 [cm], and side 50 [cm]. It is at room temperature, 21.5 [°C]. Several catheters with an outer diameter of 1.3 [mm] and a wall thickness of 0.15 [mm] [36] cross the phantom for the entire length. Each probe consists of one constantan wire (diameter \varnothing 70 [μm]) and seven copper wires with a diameter of 40 [μm] (Ella-CS, Czech Republic). The thermometry sensors are the soldered connection points of the copper wires and the constantan wire and the spacing between these sensors is 25 [mm]). The thermocouple probe is surrounded by a Teflon coating (\varnothing 0.9 [mm]). Several TC are inserted in the catheters of the phantom (red dots in Figure 6.3). TC have to be placed transversally with respect to the EM field. The duty cycle is 30 [s], that is, the antenna is on for 25 [s], then off for 5 [s] to measure the temperature.

The reference x-axis is oriented perpendicularly, entering from the center of the phantom's upper surface. Five TC are placed in different locations at 1 [cm] depth to monitor semi-deep temperature distribution on a surface orthogonal to the x-axis. To monitor the temperature distribution along the x-axis, six TC at different depths are placed. One thermocouple is placed in the water bolus bag. To simulate the cooling effect of the blood, water flow is made through one or two catheters of the phantom. These catheters have been connected to a bucket at an adjustable height. Temperature is monitored:

- at the interface between water bolus and phantom via a TMS made of eight TC with seven measuring points each, one every 25 [mm] (in purple in Figure 6.3);
- in the water bolus with a TC with three measuring points, one every 10 [mm];
- inside the phantom with TC into catheters.

The muscle-equivalent phantom is filled with wallpaper paste to take advantage of the fluid's viscosity, sufficient to suppress convective motions caused by temperature variations within the medium. The properties of wallpaper paste, since it is mainly water-based, are close to those of water.

Table 6.1. Muscle-equivalent phantom properties.

| Property | Symbol | Value |
|----------------------------------------------------------|-----------|-------|
| Density [kg m^{-3}] | ϱ | 1000 |
| Specific heat [$\text{J kg}^{-1} \text{K}^{-1}$] | c | 4180 |
| Thermal conductivity [$\text{W m}^{-1} \text{K}^{-1}$] | k | 0.6 |

6.1.3 Applicator characteristics, heating procedure

The chosen applicator is 3H type (in orange in Figure 6.3), with integrated water bolus (in blue in Figure 6.3). Measured Effective Field Sizes (parallel and perpendicular to the incident E-field), and PD of the 3H CFMA applicator at 434MHz measured on a rectangular fat/saline phantom with a 1.5 [cm] water-bolus (Ref. [43]) are given in Table 6.2.

Table 6.2. Applicator Characteristics (From [43]).

| Parameter | Value |
|--------------------|--------------------|
| Frequency [MHz] | 433 |
| Aperture size [cm] | 28.7×20.7 |
| PD [cm] | 1.36 ± 0.13 |
| EFS [cm] | 17.7×17.5 |

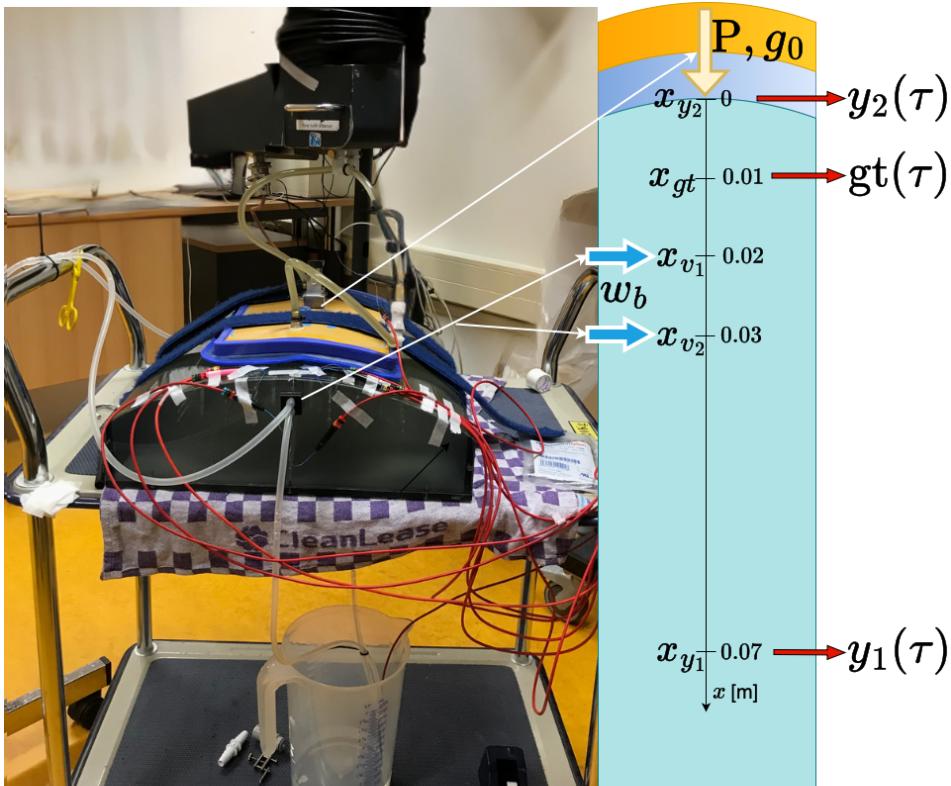


Figure 6.3. Detail of the phantom and scheme of the domain. P is the power given by the applicator, g_0 is the heat removed by the water bolus, w_b is the perfusion, y_1 , y_2 are the boundary measurements outputs, gt is the ground truth.

6.2 Main hypothesis and model

The reference x-axis enters perpendicularly from the center of the upper surface of the phantom and for an extension of $L_0 = 0.07$ [m]. Boundary information is:

1. the temperature of the deep end $\tilde{y}_1(t)$;
2. the temperature of the surface endpoint $\tilde{y}_2(t)$, which represents the measurement made by the thermocouple array placed between bolus and phantom;
3. temperature of the fluid above the phantom $\tilde{y}_3(t)$;

Note on $\tilde{y}_1(t)$: until now, it was considered constant and equal to the basal temperature because it is out of the heating range of the applicator. However, even at such depth, the temperature may rise due to conduction following long periods of heating and in the absence of a diffuse perfusion. For this reason, it must be considered as a variable. Additionally, a measurement inside the domain will be used, at a depth of $x_{gt} = 0.01$ [m], as a ground truth for the predictions of NBHO.

The convection coefficient h between the applicator's bolus and the phantom's surface can exhibit significant variability in its value. There are several methods to determine it:

- it can be assumed from literature; in particular, in [6], convective heat transfer coefficients are available for two types of antenna, slightly different from 3H: beta applicator ($h = 997 \pm 32$ [$\text{W m}^{-2} \text{K}^{-1}$]), and MA-100 Applicator ($h = 1345 \pm 241$ [$\text{W m}^{-2} \text{K}^{-1}$]));
- it can be measured from a steady state temperature distribution;
- it can be determined analytically.

In the cooling case, no power source is given, the phantom is exposed to open air, and the convection coefficient h can be retrieved from [41]. The air is assumed to be at a constant temperature $\tilde{y}_3 = 21.5$ [$^\circ\text{C}$].

Preliminary sizing of the vessels

The study in this section is conducted to investigate the conditions under which the laminar flow regime is established within the catheters.

Real fluids have viscosity and their flow through tubes causes friction. These losses of energy are called pressure drops. The Bernoulli equation for ideal fluids does not apply in this situation, but a modified version of the equation, that takes into account pressure drops, can be used. Pressure drops can be distributed or localized. Darcy-Weisbach equation describes distributed pressure drops:

$$y_c = \lambda \frac{L}{d} \cdot \frac{v^2}{2g} \quad [\text{mca}] \quad (6.1)$$

Where L [m] is the tube's length, d [m] is the diameter, v is the velocity [m s^{-1}], g is the gravity acceleration [m s^{-2}], and λ [-] is the Darcy friction factor. The shear coefficient is $\zeta = \lambda \frac{L}{d}$ [-]. In laminar flow ($Re \leq 2320$) applies

$$\lambda = \frac{64}{Re},$$

where $Re = \frac{\rho dv}{\mu}$. Localized pressure drop can be described with a similar formula:

$$y_{loc} = \zeta_{loc} \left(\frac{v^2}{2g} \right).$$

The idea is to establish a laminar motion regime by letting water flow from a bucket at height \mathbf{H} through the phantom (see Figure 6.4).

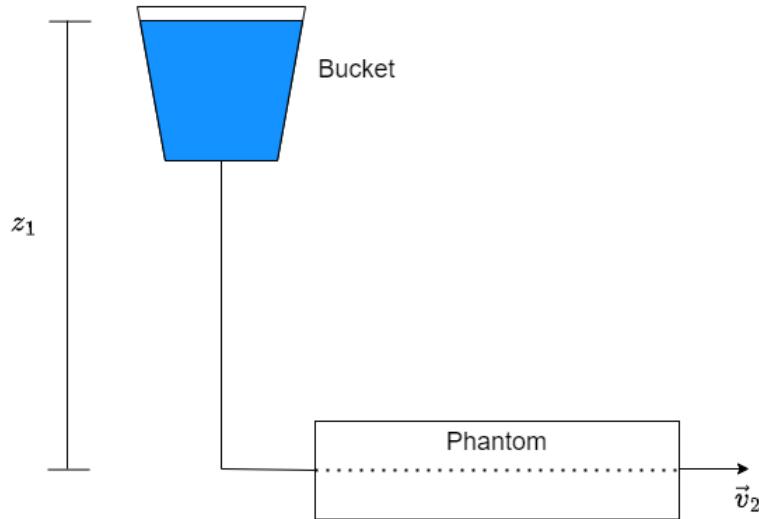


Figure 6.4. Scheme for pressure drop.

The condition of laminar flow translates into a condition on the flow velocity and the diameter of the tubes:

$$Re \leq 2300 \rightarrow vD \leq 2300 \frac{\mu}{\rho}. \quad (6.2)$$

This condition is graphically represented by a hyperbole on the $v - D$ axis (Figure 6.5a). The properties of water are $\mu = 8.9e-4$ [Pa s], $\rho = 1000$ [kg m^{-3}]. Suitable couples of (v_i, D_i) are the coordinates of the points under the hyperbole.

The inner diameter is 1.0 [mm], and the material is plastic. Water should not flow at speeds that could lead to turbulent flow. Given this constraint, the required height can now be determined. From the modified Bernoulli equation:

$$\left(\frac{p_1}{\rho g} + \frac{v_1^2}{2g} + z_1 \right) = \left(\frac{p_2}{\rho g} + \frac{v_2^2}{2g} + z_2 \right) + y_c + y_{loc}.$$

Considering $p_1 = p_2$, $v_1 = 0$, $v_2 = v$, $z_1 = \mathbf{H}$, $z_2 = 0$ and substituting the expressions for y_c and y_{loc} :

$$\mathbf{H} = \frac{v^2}{2g} (1 + \zeta + \zeta_{loc}).$$

No information about eventual localized pressure drops are given, hence as a first instance, it is possible to assume $\zeta_{loc} = 2 \cdot \zeta_c$. Considering the expressions for y_c and Re :

$$\mathbf{H} = \frac{v^2}{2g} \left(1 + 3 \cdot \frac{64\mu L}{\rho v D^2}\right). \quad (6.3)$$

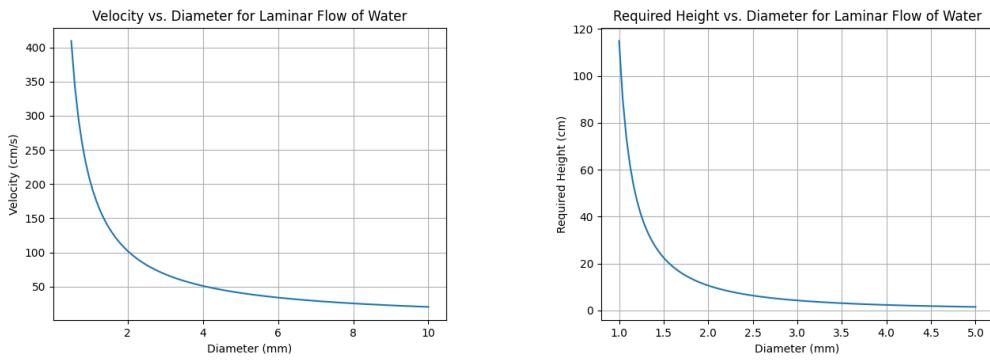
Then, the length of the tube $L \approx \mathbf{H} + l_{phantom}$ is defined, where $l_{phantom} = 0.5$ [m]. Rearranging Eq. (6.3):

$$\mathbf{H} = A[1 + B(h + l_{ph})]$$

$$\mathbf{H} = A + ABh + ABl_{ph}$$

$$\mathbf{H} = \frac{A + ABl_{ph}}{1 - AB}$$

where $A = \frac{v^2}{2g}$, $B = 2 \cdot \frac{64\mu}{\rho v D^2}$. In Figure 6.5b it is shown the required height for varying diameters, considering $v = 8.0$ [cm s⁻¹].



(a) Relationship velocity-diameter for laminar water flow Eq. (6.2). (b) Required height for varying diameters with $v = 8.0$ [cm s⁻¹].

Figure 6.5. Comparison of velocity-diameter relationship and required height for different diameters.

6.3 Overview of the measurements

The following sections report the prediction of MM-Obs obtained as a dynamic convex combination of NBHOs on two sets of measurements. The phantom is cooling down after a period of heating, with the upper surface exposed to open air. Water flows through one or two catheters inside the phantom thanks to the bucket at adjustable height $\mathbf{H} \in [\mathbf{H}_{min}, \mathbf{H}_{max}]$. The relative height between the phantom (0.655 [m] from the ground) and the lower point of the bucket has to be considered. The minimum distance from the floor to the bottom of the bucket is 0.89 [m], while the maximum is 1.74 [m], hence $\mathbf{H}_{min} = 0.235$ [m] and $\mathbf{H}_{max} = 1.085$ [m]. The height of the bucket is 0.21 [m]. It is possible to recover the velocity v_{cat} [cm s⁻¹], given the volumetric flow Λ [m³ s⁻¹] inside the $n_{cat} \in \{1, 2\}$ catheters of radius $r_{cat} = 1.0$ [mm] from Eq. (6.4):

$$v_{cat} = \frac{\Lambda}{\pi n_{cat} r_{cat}^2}. \quad (6.4)$$

These values are given in Table 6.3, while the parameters' values for the two measurements set are given in Table 6.4 and 6.5. The NBHO for the two sets of measurements share the same HPs reported in Table 6.6.

The networks will incorporate boundary measurements from the system (in-depth and superficial temperature y_1 and y_2), while the air temperature y_3 will be considered constant. The predictions will be validated towards a ground truth g_t measured inside the domain at 1 [cm] depth (Figure 6.6). These measurements are performed with TC that run through the phantom inside the catheters. Each TC has seven measuring points, 25[mm] spacing each other. From each TC, measurements from the measuring point of interest will be extracted. The point is the one aligned with the virtual needle entering perpendicularly from the phantom's upper surface and centered on it. In Figure 6.7, and Figure 6.8 is reported an overview of the measurements in the point of measurement of each TC, and in the closer points.

Table 6.3. Overview of the experiments.

| Meas Set | \mathbf{H} | n_{cat} | Λ [$\text{m}^3 \text{s}^{-1}$] | v_{cat} [m s^{-1}] |
|-----------|---------------------|------------------|------------------------------------------|----------------------------------------|
| Cooling 1 | \mathbf{H}_{\min} | 1 | $3.9e-7$ | $5.0e-1$ |
| Cooling 2 | \mathbf{H}_{\max} | 2 | $1.9e-6$ | 1.2 |

Table 6.4. Properties of the Cooling set 1.

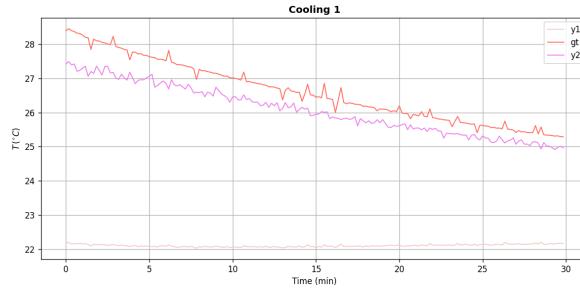
| Parameter | Symbol | Value |
|------------------------------------------------------------|------------------|----------|
| Length of the domain [m] | L_0 | 0.07 |
| Time span [s] | t^* | 1800.0 |
| Thermal conductivity [$\text{W m}^{-1} \text{K}^{-1}$] | k | 0.6 |
| Density [kg m^{-3}] | ϱ | 1000.0 |
| Specific heat [$\text{J kg}^{-1} \text{K}^{-1}$] | c | 4186.0 |
| Convection coefficient [$\text{W m}^{-2} \text{K}^{-1}$] | h | 10.0 |
| Minimum temperature [$^\circ\text{C}$] | T_{\min} | 21.5 |
| Maximum temperature [$^\circ\text{C}$] | T_{\max} | 32.0 |
| Initial in-depth temperature [$^\circ\text{C}$] | $\tilde{y}_1(0)$ | 22.17 |
| Initial superficial temperature [$^\circ\text{C}$] | $\tilde{y}_2(0)$ | 27.43 |
| Initial fluid temperature [$^\circ\text{C}$] | $\tilde{y}_3(0)$ | 21.5 |
| Fitting parameter [-] | β | 0.9 |
| Power [W] | P | 0.0 |
| Penetration Depth [m] | PD | 0.0136 |
| IC observer constant [-] | c_1 | -1.15984 |
| IC observer constant [-] | c_2 | 0.65889 |
| IC observer constant [-] | c_3 | 0.56476 |
| Output injection gain [-] | K | 2.4674 |

Table 6.5. Properties of the Cooling set 2.

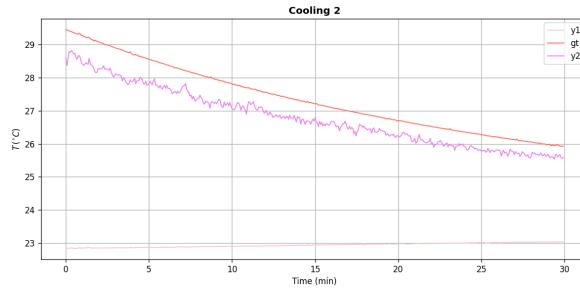
| Parameter | Symbol | Value |
|------------------------------------------------------------|------------------|----------|
| Length of the domain [m] | L_0 | 0.07 |
| Time span [s] | t^* | 1800.0 |
| Thermal conductivity [$\text{W m}^{-1} \text{K}^{-1}$] | k | 0.6 |
| Density [kg m^{-3}] | ϱ | 1000.0 |
| Specific heat [$\text{J kg}^{-1} \text{K}^{-1}$] | c | 4186.0 |
| Convection coefficient [$\text{W m}^{-2} \text{K}^{-1}$] | h | 10.0 |
| Minimum temperature [$^\circ\text{C}$] | T_{\min} | 21.5 |
| Maximum temperature [$^\circ\text{C}$] | T_{\max} | 32.0 |
| Initial in-depth temperature [$^\circ\text{C}$] | $\tilde{y}_1(0)$ | 22.84 |
| Initial superficial temperature [$^\circ\text{C}$] | $\tilde{y}_2(0)$ | 28.6 |
| Initial fluid temperature [$^\circ\text{C}$] | $\tilde{y}_3(0)$ | 21.5 |
| Fitting parameter [-] | β | 0.9 |
| Power [W] | P | 0.0 |
| Penetration Depth [m] | PD | 0.0136 |
| IC observer constant [-] | c_1 | -1.33746 |
| IC observer constant [-] | c_2 | 0.78889 |
| IC observer constant [-] | c_3 | 0.67619 |
| Output injection gain [-] | K | 2.4674 |

Table 6.6. NBHO Configuration for Measurements.

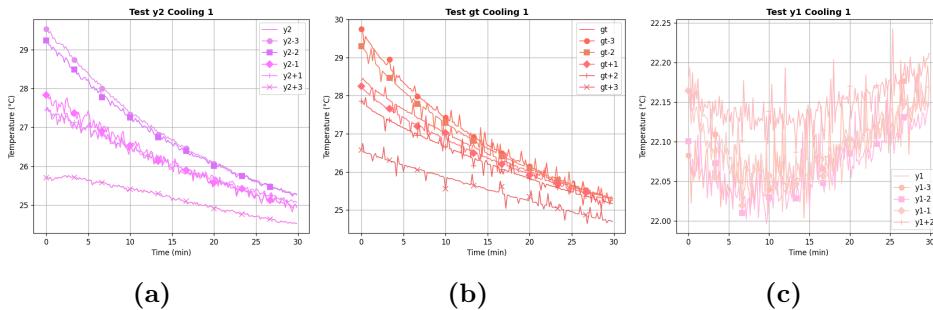
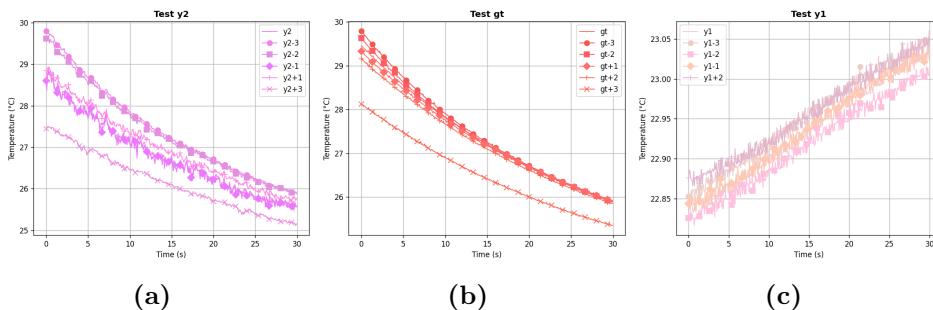
| Parameter | Symbol | Value |
|----------------------------------------|-------------------|----------------|
| Activation function | Φ | tanh |
| Initial Weights Regularization | IWR | False |
| Initialization | Υ_0 | Glorot uniform |
| Learning rate [-] | η | 0.001 |
| Depth [-] | $L - 1$ | 4 |
| Width [-] | N | 50 |
| Number of residual sampling points [-] | n_{res} | 1800 |
| Number of boundary sampling points [-] | n_b | 1000 |
| Number of test points [-] | n_{test} | 1000 |
| Resampling | RR | True |
| Weight of residual loss [-] | w_{res} | 1 |
| Weight of boundary loss [-] | w_{bc} | 1 |
| Number of inputs [-] | n_{ins} | 4 |
| Number of anchor points [-] | n_{anc} | 0 |



(a) Cooling 1.



(b) Cooling 2.

Figure 6.6. Cooling measurements for two different scenarios.**Figure 6.7.** Measurements at the points immediately before and after the reference measuring point, Cooling 1.**Figure 6.8.** Measurements at the points immediately before and after the reference measuring point, Cooling 2.

6.4 Results

Here we report the results of the PINNs-implemented MM-Obs tested on real measurements described in Table 6.3. Specifically, we employ the inverse problem (offline stage) for a preliminary evaluation of the unknown perfusion w_b . Then, we employ the adaptive setup for an online temperature assessment of temperature distribution. We demonstrate that, as the granularity of the observers n_{obs} increases, the prediction is more accurate. Also, we point out some limitations of the current experimental setup.

6.4.1 Inverse problem

This simulation was conducted to obtain a preliminary estimation of the expected perfusion value of the phantom in an offline setting. Obtained values are reported in Table 6.7, while their evaluation during training is reported in Figure 6.9. A difference in the perfusion value between the two measuring sets emerges, with Cooling 2 the one with higher perfusion.

Table 6.7. Inverse Problem.

| Meas Set | $w_b[\text{s}^{-1}]$ |
|-----------|----------------------|
| Cooling 1 | $2.0e-4$ |
| Cooling 2 | $2.2e-4$ |

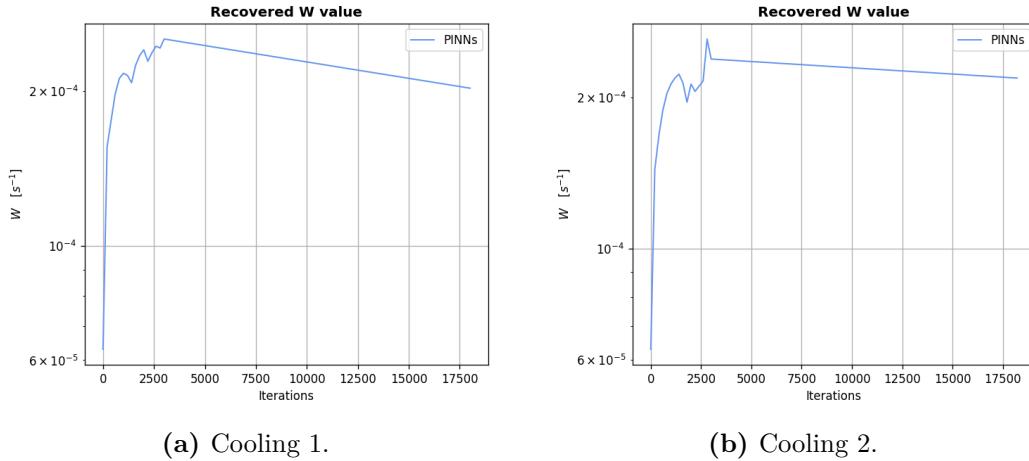


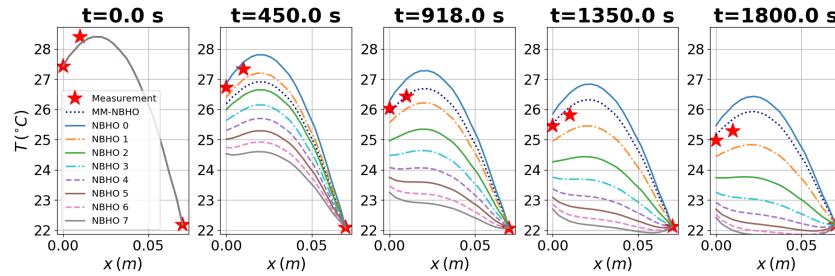
Figure 6.9. Retrieved values for perfusion with inverse problem.

6.4.2 Adaptive estimation

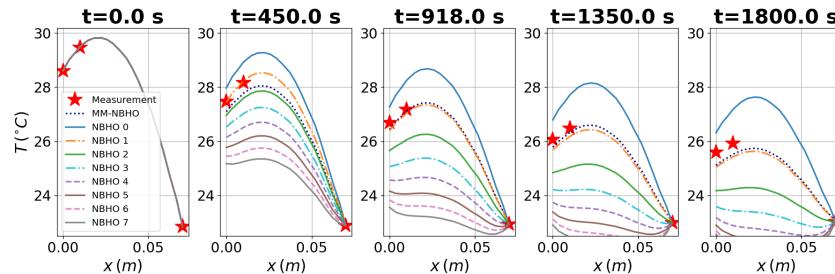
As the precise perfusion of the system is unknown, an arbitrary perfusion interval is selected. Based on the inverse problem's outcomes, it's probable that the system's perfusion will be minimal. Typical values for biological tissues are used to define the range, with bone as the lower limit and tumor as the upper limit. Results for a family of eight observers with perfusion values listed in Table 6.8 are shown in Figure 6.10. Predictions obtained using sixty-four NBHOs, sampled within the same perfusion range, demonstrate accurate convergence to the real measurements, as illustrated in Figure 6.11. A comparison with the system outputs is presented in Figure 6.12. As the number of observers n_{obs} —sampled within the same interval—increases, the observation error at the surface and at X_{gt} decreases, as shown in Figure 6.13. The L^2 error norm between $\hat{\theta}^\dagger$ and the system measurements, for increasing n_{obs} , is reported in Figure 6.14. Additional figures in Appendix B provide further insight into the network's performance and the behavior of the MM-Obs.

Table 6.8. Values of \tilde{w}_b [s^{-1}] used in the simulations.

| | | | |
|-----------------------|----------|-----------------------|----------|
| $\tilde{w}_b^{(0)}$: | $6.3e-5$ | $\tilde{w}_b^{(1)}$: | $1.0e-4$ |
| $\tilde{w}_b^{(2)}$: | $1.6e-4$ | $\tilde{w}_b^{(3)}$: | $2.6e-4$ |
| $\tilde{w}_b^{(4)}$: | $4.1e-4$ | $\tilde{w}_b^{(5)}$: | $6.5e-4$ |
| $\tilde{w}_b^{(6)}$: | $1.0e-3$ | $\tilde{w}_b^{(7)}$: | $1.7e-3$ |



(a) Cooling 1.



(b) Cooling 2.

Figure 6.10. Comparison of observers' predictions at different instants, $n_{\text{obs}}=8$.

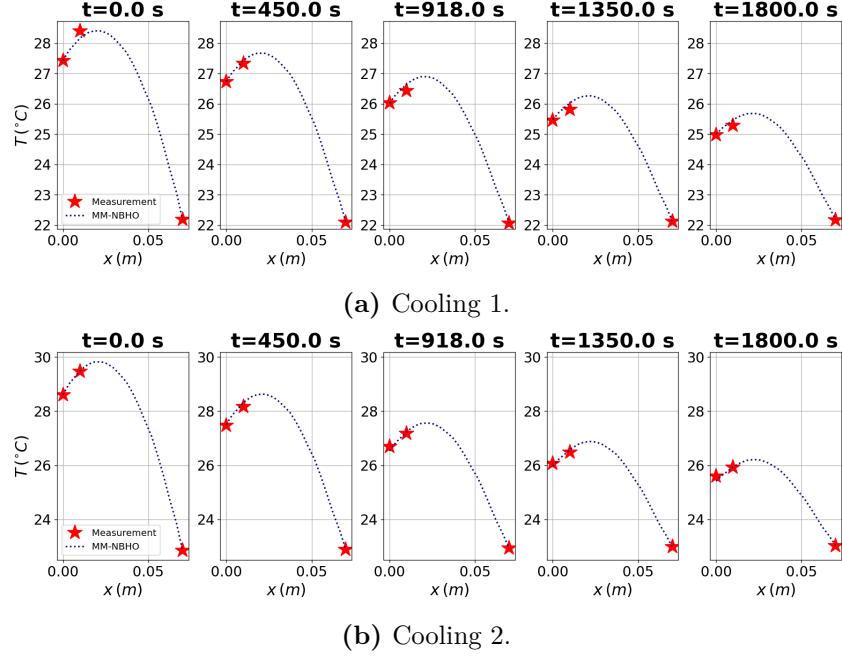


Figure 6.11. Predictions at different instants, $n_{\text{obs}} = 64$.

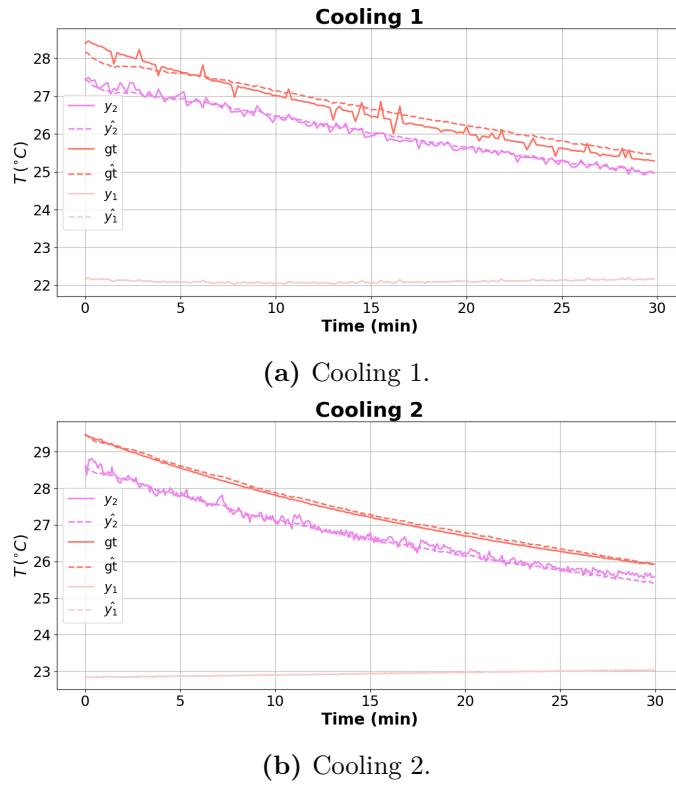


Figure 6.12. Overview of outputs and predictions, $n_{\text{obs}} = 64$.

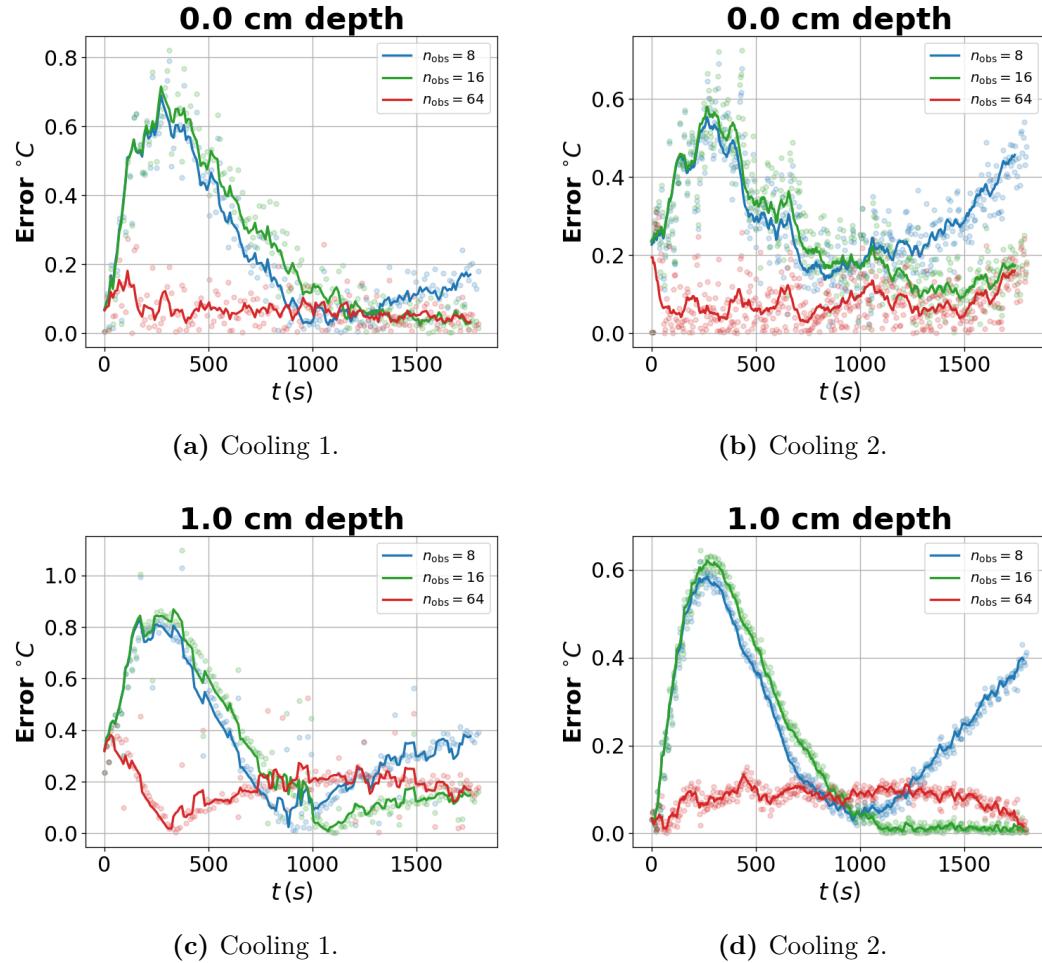


Figure 6.13. Prediction errors for different numbers of observers.

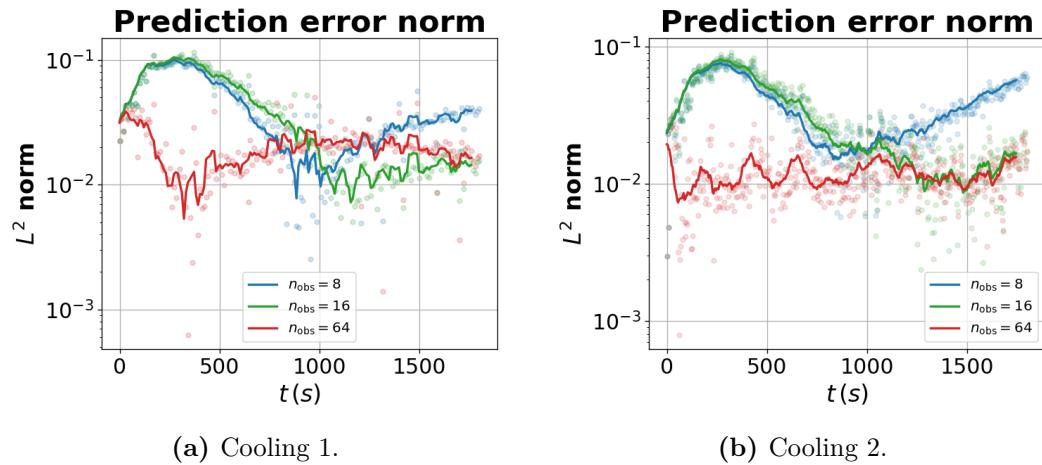


Figure 6.14. \mathcal{L}^2 errors for different number of observers.

6.4.3 Considerations on in-depth ground truth measurements

In the region beneath the water catheters, the predicted temperature values exhibit a significant deviation from the actual measurements at $x_{gt_1} = 0.04$ [cm] (Figure 6.15 and Figure 6.16).

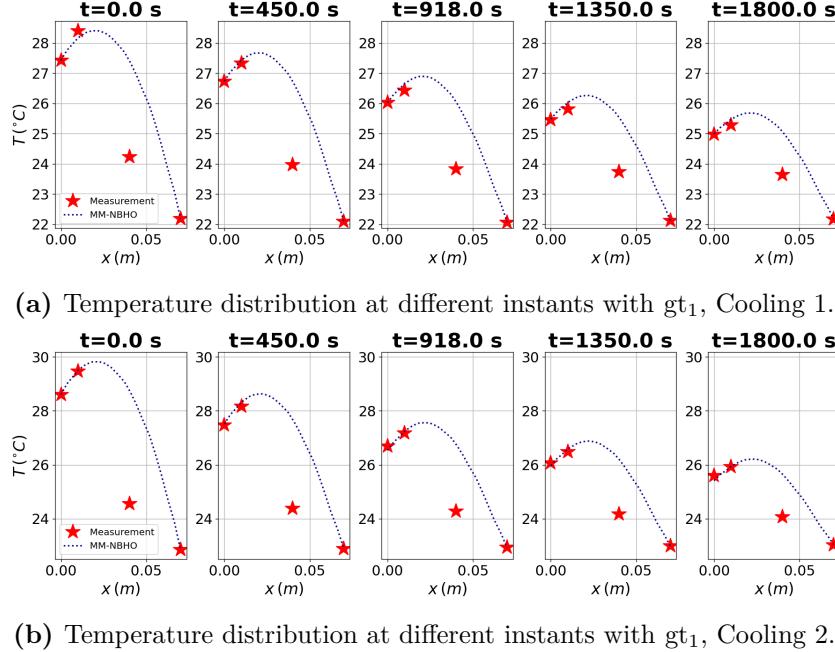


Figure 6.15. Temperature distributions and predictions with different cooling measurements and considering gt_1 .

This discrepancy can be attributed to the cold track effect induced by the water vessel, in combination with the low penetration depth of the applicator. Given the latter, only a relatively small amount of EM energy is deposited in the deeper regions. As a result, heating in that region primarily occurs through heat conduction from the upper region. The water vessel, acting as a heat sink, represents a resistance to the heat conduction in this region.

To take into account the effect of the discrete vessel, it is necessary to consider a more complex model. Specifically, thermal properties and perfusion can be assumed as space-variant. Two approaches can be identified, the first consisting of splitting the domain in two parts characterized by constant properties, while the second is to consider the properties as spatially variable.

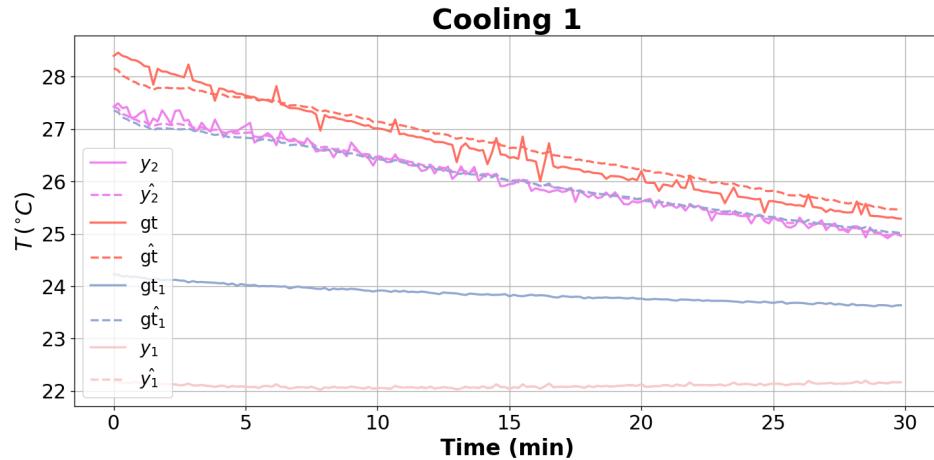
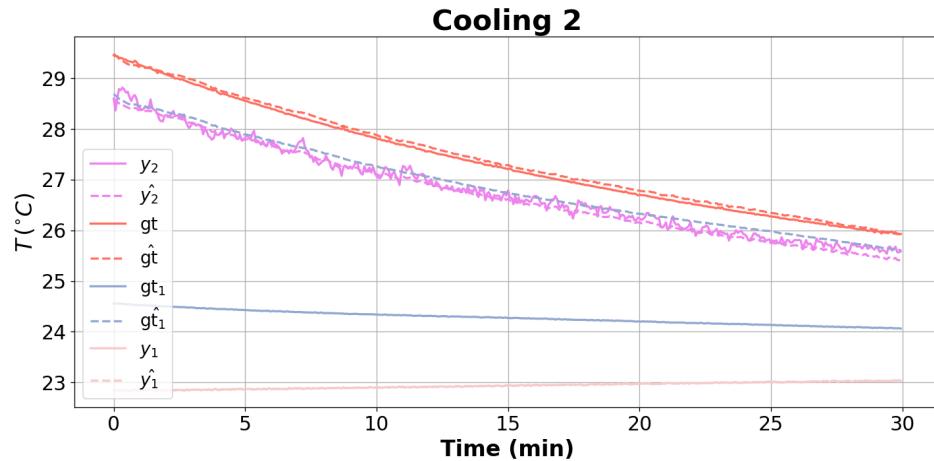
(a) Measurements and predictions with gt_1 , Cooling 1.(b) Measurements and predictions with gt_1 , Cooling 2.

Figure 6.16. Comparison with system outputs with different cooling measurements and considering gt_1 . While the MM-Obs can predict effectively the measured ground truth gt , it fails in the prediction of the most internal gt_1 .

Splitting the domain

The first proposed approach is to split the domain in two parts $\Omega_I : (X, \tau) \in [0, X_{gt_1}] \times [0, +\infty)$ and $\Omega_{II} : (X, \tau) \in [X_{gt_1}, 1] \times [0, +\infty)$, with $X_{gt_1} = \frac{x_{gt_1}}{L_0}$

On the second region, a MM-Obs $\hat{\theta}_{II}^\dagger$ can be introduced, with every single observer $\hat{\theta}_{II}$ defined as:

$$a_1^{II} \partial_\tau \hat{\theta}_{II} = \partial_{XX} \hat{\theta}_{II} - \hat{w}_b^j a_2^{II} \hat{\theta}_{II} + a_3^{II} \exp(-a_4^{II} X) \quad (6.5)$$

for $(X, \tau) \in [X_{gt_1}, 1] \times [0, +\infty)$. The BCs are

$$\begin{aligned} \partial_X \hat{\theta}_{II}|_{X=X_{gt_1}} &= q(\tau) + K^{II}(\hat{\theta}_{II}(0, \tau) - p(\tau)) \quad \tau \geq 0 \\ \hat{\theta}_{II}(1, \tau) &= y_1(\tau) \quad \tau \geq 0 \end{aligned} \quad (6.6)$$

The outputs q, p being related to physical considerations of the system ζ and the temperature and flux prediction¹ at the boundary:

$$q, p = f(\hat{\theta}^\dagger(X_{gt_1}), \partial_X \hat{\theta}^\dagger|_{X=X_{gt_1}}, \zeta)$$

Recalling Par. 2.3.2 an information ζ of interest could be the expected vessel temperature $\langle T(0) \rangle$.

Considering varying properties

Instead of dividing the domain into two parts, one can follow a similar procedure as the one described in Chapter 4 considering the functions $\varrho(\cdot)$, $c(\cdot)$ and $k(\cdot)$ as continuous, with bounds given by

$$\begin{aligned} 0 < \varrho_{min} \leq \varrho(x) \leq \varrho_{max} &\quad \forall x \in [0, L_0] \\ 0 < c_{min} \leq c(x) \leq c_{max} &\quad \forall x \in [0, L_0] \\ 0 < k_{min} \leq k(x) \leq k_{max} &\quad \forall x \in [0, L_0], \end{aligned}$$

And the Lyapunov functional candidate

$$V(e_T(\cdot, t)) = \frac{1}{2} \int_0^{L_0} \varrho(x) c(x) e_T^2(x, t) dx,$$

which, thanks to the coercivity of the term $\varrho(x)c(x)$, satisfies the quadratic bounds

$$\frac{\varrho_{min} c_{min}}{2} \|e_T(\cdot, t)\|_{\mathcal{L}^2}^2 \leq V(e_T(\cdot, t)) \leq \frac{\varrho_{max} c_{max}}{2} \|e_T(\cdot, t)\|_{\mathcal{L}^2}^2.$$

¹In PINNs, thanks to AD, the derivative of the outputs with respect to the inputs are easily accessible

6.5 Conclusion

The proposed simulations demonstrate the possibility of obtaining the temperature distribution inside a domain from the solution of PBHE in real-time based on boundary measurements and in the presence of unknown properties of the system. Consistency with theoretical expectations has been confirmed, as increasing the number of observers leads to a reduction in prediction error 4.4.1.

A key challenge identified in the experimental design is the difficulty of replicating perfusion in a non-biological system. To address this, a method to simulate perfusion is introduced by forcing water flow through the phantom's catheters. However, modeling the cooling effect of discrete vessels as a surrogate for physiological perfusion—where heat dissipation is influenced by the complex, diffused, and chaotic nature of capillary vascularization—may introduce inaccuracies in the predictions.

A key challenge identified in the experimental design is the difficulty of replicating perfusion in a non-biological system. To address this, a method to simulate perfusion by forcing water flow through the phantom's catheters is introduced. However, modeling the cooling effect of discrete vessels as a surrogate for physiological perfusion—where heat dissipation is influenced by the complex, diffused, and chaotic nature of capillary vascularization—may introduce inaccuracies in the predictions. The perfusion values retrieved offline, with the inverse problem procedure, are relatively low, being the perfusion of the first set (one vessel, minimum height) slightly lower than the perfusion of the second set (two vessels, maximum height). In the online stage, the adaptive estimation of PBHE with the proposed PINNs-based MM-Obs approach shows accordance to the results of the inverse problem since the MM-Obs lies between NBHO 0 ($\tilde{w}_b^{(0)} = 6.3e-5$ [s⁻¹]) and NBHO 2 ($\tilde{w}_b^{(2)} = 1.6e-4$ [s⁻¹]), see Figure 6.10).

Another important finding regards the in-depth temperature rise, which in our first assumptions was negligible, because outside of the heating range of the antenna. In our experimental setup, temperature y_1 is slowly rising (see Figure 6.12), even if the antenna is shut down, as a consequence of heat conduction from the uppermost, warmer region. In a biological system, with proper cooling due to perfusion, this effect should be negligible, and in-depth measurement could be avoided thus ensuring minimal invasiveness for patients.

Potential extensions include applying the method to a multilayer composition, extensions to multiple spatial dimensions, as well as considering a variable perfusion term $w_b(X, \tau)$. This can be achieved by adopting more advanced perfusion models, such as in Ref. [32, 29, 27, 16].

Chapter 7

A novel application: the wave equation

This chapter demonstrates the application of PINNs for solving the wave equation, with the objective of predicting the deformation of biological tissues following interaction with surgical devices in mixed reality.

The development of virtual environments that accurately simulate the physical properties and behaviors of human tissues represents a significant frontier in medical technology, aimed at improving surgical outcomes, patient safety, and clinical training. While soft tissue simulation remains challenging and is the subject of active research, no definitive tool currently allows real-time user interaction with virtual environments featuring dynamically simulated deformable materials.

The proposed PINNs-based approach for simulating tissue deformation during dynamic interaction represents a significant departure from traditional simulation techniques, which typically rely on conventional numerical solvers such as FDTD and are poorly suited for real-time applications. This implementation revealed key aspects of PINNs models not observed in previous bio-heat transfer applications, particularly the inherent bias toward low-frequency solutions. This limitation was addressed through Fourier feature embedding of the input [68], enabling the networks to learn high-frequency solutions within low-dimensional input domains.

Following an overview on the current state of the art given in Section 7.1, the wave equation is introduced and the PDE problem formalized in Section 7.2. Optimization studies for PINNs in this context are then presented in 7.3, concluding with numerical validation and future work considerations in Sections 7.4-7.5.

7.1 Introduction

The medical and surgical domains are experiencing a significant period of innovation, largely driven by the introduction of advanced technologies. Among these, simulation models for surgical training and planning have emerged as critical components of the modern healthcare workflow. This innovation not only improves the efficiency of medical procedures, but it also contributes to safer patient outcomes and more personalized care. A particularly, challenging but vital, aspect of this innovation is the dynamic simulation of soft tissues, which plays a crucial role in creating realistic

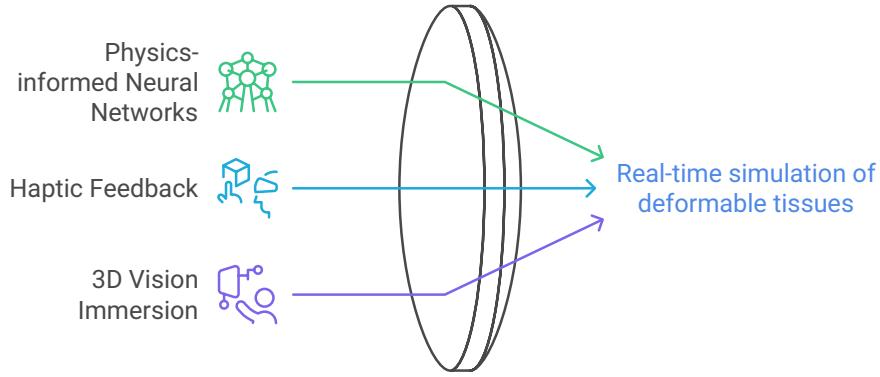


Figure 7.1. Real-time simulation of deformable tissues using PINNs: integrating haptic feedback and 3D vision for high-fidelity, immersive interaction.

simulation environments and tools for physicians.

Simulating soft materials presents a unique set of challenges. The complexity of soft tissue behavior—its ability to deform, stretch, and compress in response to surgical manipulations—requires sophisticated computational models to replicate accurately. The main problems tackled are computation time and accuracy level, parameters that cannot be improved at the same time.

Despite significant advancements, current soft tissue simulation methods face inherent trade-offs between computational speed and physical accuracy. High-fidelity models like FDTD offer precise simulations but are computationally intensive, making real-time applications challenging. Conversely, simplified models such as mass-spring systems enable faster computations but lack the ability to accurately replicate complex tissue behaviors, including nonlinear elasticity and anisotropy. This dichotomy underscores the difficulty in achieving both real-time performance and high accuracy simultaneously, a balance that remains a central challenge in the field.

We propose to tackle this problems with a novel approach based on PINNs, as part of a Italian National Recovery and Resilience Plan (PNRR) project which focuses on the creation of physical and digital phantoms for surgical training and planning. The aim is to construct a digital twin of the physical phantom, enabling real-time interaction through advanced haptic feedback and visual immersion devices (Figure 7.1).

The integration of haptic perception and 3D vision immersion into this system is designed to render the physical experience of interaction with a high degree of realism. Haptic devices provide tactile feedback, simulating the sensation of touching and manipulating tissues, while 3D vision immersion helps replicate the visual context of surgery, enhancing the overall fidelity of the simulation.

Although preliminary, the results presented are highly promising. This approach has the potential to revolutionize surgical training and planning, offering a safe environment for physicians to hone their skills without risk to patients. Furthermore, enabling the analysis and planning of personalized interventions supports the broader trend toward customized healthcare solutions.

7.2 The wave equation for vibrating strings

7.2.1 The ideal string

The simplest model for string vibration is derived from the classical wave equation, which describes the motion of an idealized string under tension. For an infinitesimal element of the string, applying Newton's second law and assuming small displacements, the governing equation becomes:

$$\partial_{tt}^2 y = a^2 \partial_{xx}^2 y, \quad (7.1)$$

defined for $(x, t) \in [0, L] \times [0, +\infty)$, where $y(x, t)$ represents the transverse displacement of the string of length L [m] at position x and time t , and $a = \sqrt{\frac{F}{m}}$ [m s^{-1}] is the wave speed, with F [N] being the string tension and \bar{m} [kg m^{-1}] the linear mass density. The mathematical problem is completed by specifying appropriate IC and BCs:

$$\begin{cases} y(x, 0) = w_1(x) \\ \partial_t y(x, 0) = w_2(x) \\ y(0, t) = 0 \\ y(L_0, t) = 0 \end{cases}$$

where $w_1(x)$ and $w_2(x)$ represent the initial displacement and velocity distributions, respectively, and the ends are fixed (Dirichlet BCs).

The hyperbolic PDE (7.1) admits analytical solutions through separation of variables, and the general solution can be expressed as a Fourier series:

$$y(x, t) = \sum_{n=1}^{\infty} \left[A_n \cos \left(\frac{n\pi a t}{L} \right) + B_n \sin \left(\frac{n\pi a t}{L} \right) \right] \sin \left(\frac{n\pi x}{L} \right)$$

The coefficients A_n and B_n are determined by the initial conditions through orthogonality relations:

$$A_n = \frac{2}{L} \int_0^L w_1(x) \sin \left(\frac{n\pi x}{L} \right) dx$$

$$B_n = \frac{2}{n\pi a} \int_0^L w_2(x) \sin \left(\frac{n\pi x}{L} \right) dx$$

7.2.2 Introducing stiffness effects

Real strings exhibit additional physical properties that deviate from the idealized model. Most notably, the bending stiffness of the string introduces fourth-order spatial derivatives, leading to the modified wave equation for a vibrating string [20]:

$$\partial_{tt}^2 y = a^2 \partial_{xx}^2 y - \frac{EI}{\mu} \partial_{xxxx}^4 y$$

This can be rewritten in the form:

$$\partial_{tt}^2 y = a^2 \partial_{xx}^2 y - E \cdot S \cdot K^2 \partial_{xxxx}^4 y$$

defined for $(x, t) \in [0, L] \times [0, +\infty)$, where:

- E [N m^{-2}] is Young's modulus characterizing the material stiffness;
- $S = \pi r^2$ [m^2] is the cross-sectional area of the string with radius r [m];
- $K = \frac{r}{2}$ [m] is the gyration factor.

The fourth-order term $E \cdot S \cdot K^2 \partial_{xxxx}^4 y$ represents the bending stiffness contribution, which becomes particularly significant for thick strings where the radius is not negligible compared to the length, for high-frequency modes where spatial gradients are large, and for materials with high elastic modulus.

7.2.3 Scaling

The inclusion of the stiffness fundamentally changes the mathematical character of the equation, introducing several computational challenges. Closed-form solutions are generally not available for the full stiffness-modified equation, and the fourth-order spatial derivative requires higher-order numerical schemes and finer spatial discretizations.

These challenges motivate the use of advanced numerical methods, such as PINNs, which can handle the nonlinear coupling between different spatial derivatives while maintaining computational efficiency for real-time applications.

As reported in [72], which refers to [45] and [26], feedforward NNs need all the inputs to be possibly in $[0, 1]$, to improve the training. This reduces the magnitude of the weights, thus improving the optimization algorithm in finding the solution that minimizes the loss function. In this case, also the output of PINN should be in this range.

To achieve this, it is possible to produce an equivalent PDE (together with equivalent IC and BCs) that is nondimensionalized and in the range $[0, 1]$.

To scale the coordinates space x and time t to $\mathcal{O}(1)$, an upper bound is needed. For space this will be the length of the domain L , while for time a reference interval is chosen as $\tau_0 = 20$ [s].

The new coordinates will be:

$$X = \frac{x}{L}, \quad \tau = \frac{t}{\tau_0}.$$

After the transformation, the equation can be recast as follows:

$$\partial_{\tau\tau}^2 Y = C^2 \partial_{XX}^2 Y - \mathcal{E} \partial_{XXXX}^4 Y,$$

with $C = a \frac{\sqrt{\tau_0}}{L}$ and $\mathcal{E} = E \cdot S \cdot K^2 \frac{\tau_0}{L^4}$.

As detailed in Section 7.4, the choice of the parameter C , or equivalently, of the parameters a and τ_0 , is strategically important, as it is known that PINNs present a bias towards low-frequency solutions. For this reason, some changes must be applied in the implementation of the conventional FNN, as we will see in the next Section.

Algorithm 3 Workflow for PINN-Based Problem Solving.

-
- 1: Define the model and non-dimensionalise the variables for network stability;
 - 2: Validate PINNs prediction against solution computed with classical solvers;
 - 3: **if** necessary **then**
 - 4: Employ techniques to improve accuracy;
 - 5: **end if**
 - 6: Increase the network dimensionality: the same network can solve different problems. With a classical solver, it would require different runs;
 - 7: Perform experimental tests of the networks.
-

7.3 PINNs for simulation of deformable materials

The solution of PDEs with DL attracts interdisciplinary attention. In the many different domains of application, a common methodological workflow can be extrapolated, and it is reported in Algorithm 3.

The implementation on the wave equation revealed key aspects of PINNs not observed in bio-heat transfer applications of previous chapters. Several techniques to improve training were investigated, and we report here the early stopping to prevent overfitting and the Fourier features embedding to capture high-frequency solutions within low-dimensional input domains.

Definition 17 (Early stopping) *Early stopping is a regularization technique used to prevent overfitting in neural networks by monitoring the performance of the model on a validation set during training. If the validation loss stops improving for a predefined number of epochs, training is halted to avoid excessive fitting to the training data. This method ensures that the model retains generalization capabilities while reducing computational cost and mitigating overfitting, making it a simple yet effective strategy for improving performance.*

In [68] the authors demonstrated an implicit bias in PINNs to learn high-frequency solutions. To mitigate this bias, they proposed a Fourier feature embedding of the input, demonstrating that with such an embedding the networks can learn high-frequency solutions with low-dimensional input domains.

This encoding is expressed as follows:

$$\gamma(v) = \begin{pmatrix} a_1 \cos(2\pi \mathbf{b}_1^T v) \\ a_1 \sin(2\pi \mathbf{b}_1^T v) \\ \vdots \\ a_m \cos(2\pi \mathbf{b}_m^T v) \\ a_m \sin(2\pi \mathbf{b}_m^T v) \end{pmatrix}$$

where $a_j = 1 \forall j$ and \mathbf{b}_j are a random vector sampled from a normal distribution $\mathcal{N}(\mu, \sigma^2)$ with $\mu = 0$ and σ appropriately chosen based on the expected frequency range of the solution. All these parameters, even if randomly selected, are not trainable parameters, and thus they remain fixed during training. This encoding is applied before passing the input to the MLP (as represented in Figure 7.2), so that

the MLP receives the encoded input. Note that no output decoding is performed, so MLP learns the intended task in the original output domain.

Definition 18 *A Spatio-temporal Multi-scale Fourier Feature Network (STMsFFN) is an MLP with Fourier Feature embedding of the input.*

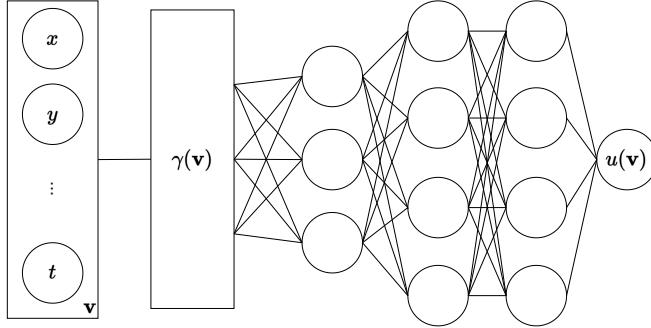


Figure 7.2. Input encoding for STMsFFN.

This encoding showed a particular improvement in the solution learned by the PINN, which will be detailed in the subsequent sections of the experiments.

7.4 Experiments

Given the PDE:

$$\partial_{\tau\tau}^2 Y = C^2 \partial_{XX}^2 Y - \mathcal{E} \partial_{XXXX}^4 Y$$

with boundary conditions

$$\begin{cases} Y(0, \tau) = 0 & \tau \geq 0 \\ Y(L_0, \tau) = 0 & \tau \geq 0 \end{cases}$$

Initial computational experiments were performed in Python using DeepXDE as the DSL to analyze a scenario that features a specific initial displacement $w_1(X)$ with zero velocity (“plucked string”).

7.4.1 Plucked string

The string is plucked at one-fifth of its length (Figure 7.3). The expression for $w_1(X)$ will be:

$$w_1(X) = \begin{cases} 5X & X \in [0, 0.2] \\ 1.25(1 - X) & X \in (0.2, 1] \end{cases} \quad (7.2)$$

The initial velocity is set to zero:

$$w_2(X) = 0 \quad X \in [0, 1].$$

The spatiotemporal coordinates (X, τ) are jointly embedded using two separate Fourier feature mappings initialized with $\sigma = 1, 10$, respectively. Training of the model STMsFFN (3 layers, 100 hidden units, tanh activations) was carried out for 1 million iterations of the optimizer Adam with early stopping (Figure 7.4). The results for $C=10$ are given in Figure 7.5.

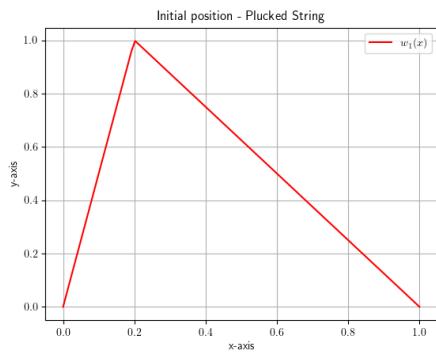


Figure 7.3. Initial position for plucked string, as defined in Eq. (7.2).

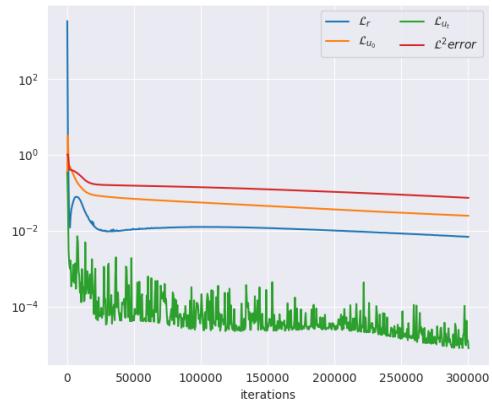


Figure 7.4. Training Losses Plucked string, C=10.

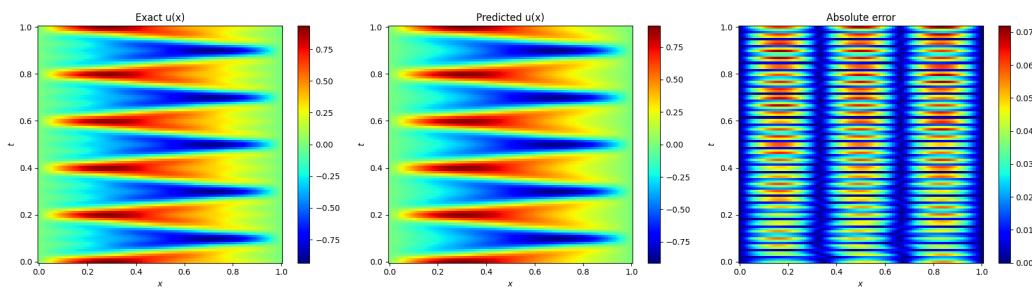


Figure 7.5. Plucked string, C=10.

Increasing the velocity

Augmenting the velocity to $C = 25$ adversely affects the performance of the network. Increasing the value of σ subsequently improves the precision of the prediction. This observation indicates a need for further investigation into the relationship between σ and C (see Figure 7.6).

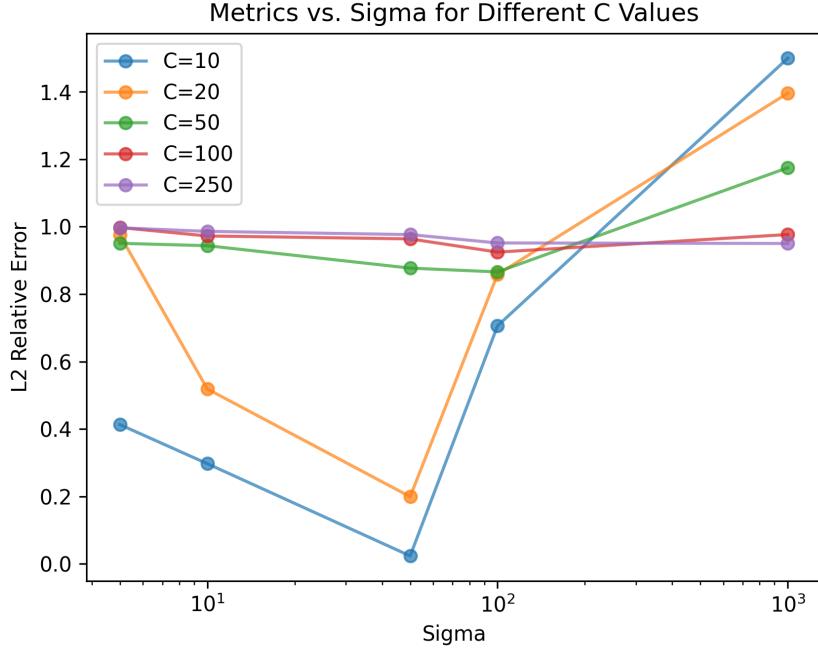


Figure 7.6. Study on the correlation between Fourier feature mapping for time σ and wave velocity C .

7.5 Discussion

Simulations showed the difficulty of PINNs in predicting the solution of forward problems exhibiting high-frequency or multiscale behavior [73]. In particular, different terms in a PINNs loss may dominate one another, as highlighted by Neural Tangent Kernel (NTK) theory [73], leading to models that cannot simultaneously fit BCs or IC and minimize the PDE residual. The second weakness is spectral bias, i.e., networks are not able to learn high-frequency functions. We have focused our attention on

- Fourier feature networks that use a simple Fourier feature mapping to enhance the ability of MLP to learn high-frequency functions[68].
- NTK theory as a theoretical tool for detecting and analyzing spectral bias.
- An algorithm for adaptive weights for the loss function to prevent inhomogeneous fitting. As stated in Ref. [47], however, this solution should only be investigated when the implementation of hard constraints is not feasible;

7.6 Further developments

7.6.1 Struck string and membrane

The second scenario to be studied considers zero initial displacements and a certain velocity distribution $w_2(X)$ (“struck string”).

$$\begin{cases} w_1(X) = 0 & X \in [0, 1] \\ w_2(X) = 50e^{-\frac{(X-0.5)^2}{0.05}} & X \in [0, 1] \end{cases}$$

The extension to the two-dimensional case, with a certain initial velocity distribution representing an impulse on the membrane at the initial time and the possibility of augmenting the input dimensionality to account for the module and position of the impulse have been studied. This enables the possibility to apply PINN to control problems (PINC).

7.6.2 Mathematical formulation of the PINC problem

Given a time step T , and given a function approximator $\bar{u}(x, t, f, w_1^{(k)}, w_2^{(k)}, \theta) \simeq u(x, t)$ represented by the well-trained NN, the solution for the PDE of a vibrating string is computed for a given time $t \in [kT, (k+1)T]$:

$$u(x, t) = \bar{u}(x, t, \bar{f}, w_1^{(k)}, w_2^{(k)}, \theta)$$

with:

$$\begin{aligned} \bar{f} &= f(x, kT) \\ w_1^{(k)} &= \bar{u}(x, kT, \bar{f}, w_1^{(k-1)}, w_2^{(k-1)}, \theta) \\ w_2^{(k)} &= \frac{\partial \bar{u}(x, kT, \bar{f}, w_1^{(k-1)}, w_2^{(k-1)}, \theta)}{\partial t} \\ w_1^{(0)} &= w_1 \\ w_2^{(0)} &= w_2 \end{aligned}$$

The PINC approach has been implemented following the schema in Figure 7.7. Given that NN has a fixed and limited number of inputs, states $w_1^{(k)}(x)$ and $w_2^{(k)}(x)$ must be sampled in the domain \mathcal{D} . This sampling is critical for the dimension of the overall input domain, since if the state $w_1^{(k)}(x)$ and $w_2^{(k)}(x)$ is sampled with n points each, the NN input (and so the input domain for which NN is trained) increases its dimension by $2n$. Moreover, when calculating the loss functions IC during training, since $w_1^{(k)}(x)$ and $w_2^{(k)}(x)$ are represented only by some points, an interpolation has to be done, in order to allow calculating such loss functions for a generic $x \in \mathcal{D}$ that may be given as input during training. In our implementation, the interpolation is linear. This approach has several problems in learning the correct solution, given the complexity of the problem and the high number of state variables (i.e., sample points for position and speed of the particles of the string/membrane) that are currently under investigation.

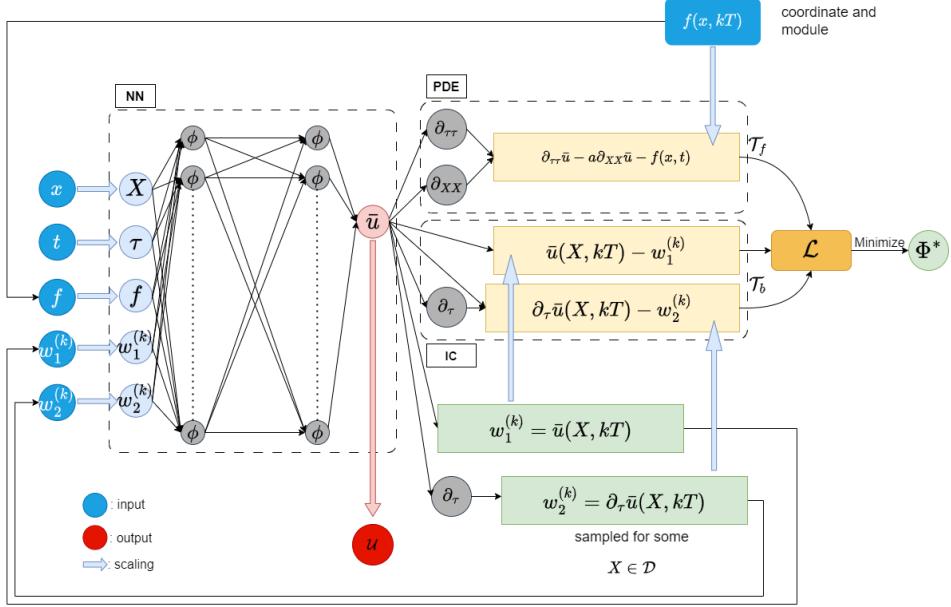


Figure 7.7. Scheme for PINC with augmented input.

7.7 Conclusions

PINNs provide a paradigm shift in simulating soft tissue mechanics by reconciling the trade-off between computational efficiency and physical accuracy. Unlike traditional numerical solvers, PINNs do not require extensive training datasets—only the governing physical laws and boundary conditions are needed. This enables real-time resolution of idealized PDE-based models, such as the wave equation, without compromising mathematical rigor.

More importantly, PINNs offer a powerful framework for solving inverse problems. By integrating sparse experimental measurements, for example from piezoelectric pressure sensors, they can infer spatially varying physical parameters such as tissue elasticity, damping, or density. This is particularly relevant for personalizing simulations to match patient-specific characteristics. Just as in the case of the bio-heat equation, observer design and parameter estimation techniques can be embedded within the PINN architecture to dynamically adapt the model to real-world tissue responses.

Consequently, PINNs are not merely a computational shortcut—they enable high-fidelity real-time simulations that can be grounded in idealized theory or empirical data, making them ideal for interactive surgical planning, diagnostics, and therapeutic guidance.

Chapter 8

Conclusions

This thesis explored the observability of temperature inside a domain when boundary measurements are given. The use case is oncological HT, a treatment used in combination with RT and/or CH to enhance their effectiveness, especially for relapses and hard-to-treat tumors. Currently, these treatments lack an online quality assessment that could improve their effectiveness. This is because the only temperature information at the target is currently obtainable through invasive measurements, that return pointly and non-precisely localized measurements. The situation is made even more complex by the presence of unknowns in the properties of the patient, in particular the blood perfusion rate, which can change when the tissue undergoes heating. It is not only the solution of the PDE that models bio-heat transfer (PBHE), but the state estimation of a system governed by PDE with unknown and potentially time-varying properties.

Moreover, this approach paves the way for the fast simulation of deformable materials through the solution of the wave equation with application to surgical training in mixed reality.

8.1 Contributions

In Chapter 4, we proposed a setup based on multiple-model adaptive estimation, mimicking the approach described in [30] for finite-dimensional linear systems. The multiple-model adaptive observer is constituted by a family of observers that can access the temperature at the boundaries of the system and evolves following a copy of PBHE. Each of the observers has a different arbitrary value for the perfusion. The overall estimation is a weighted average of the prediction of each observer. Observer weights evolve following an ODE. The observation error is ultimately bounded and converges to the observer whose perfusion is the closest to the actual one.

In Chapter 5 we report limitations shown by traditional numerical solvers in approaching this type of problem. The bottleneck appeared to be the solution of a system of PDEs in combination with a ODE. We proposed an AI-oriented solution based on PINNs that can seamlessly solve the system of PDEs without requiring a pre-computed or measured training dataset. Instead, PINNs leverage the underlying physical law PBHE for training. Moreover, they enable continuous sampling of the domain without the need for a solving grid as FDTD methods. This method offers

high adaptability by incorporating parameters and boundary conditions as inputs to the NN, allowing it to generalize across a wide range of cases.

In Chapter 6, we tested our AI-based model on real measurements conducted at Amsterdam UMC mimicking a superficial HT treatment. Consistency with theoretical expectations has been confirmed, as increasing the number of observers leads to a reduction in prediction error. A key challenge identified in the experimental design is the difficulty of replicating perfusion in a non-biological system. To address this, we introduced a method to simulate perfusion by forcing water flow through the phantom’s catheters. However, modelling the cooling effect of discrete vessels as a surrogate for physiological perfusion—where heat dissipation is influenced by the complex, diffused, and chaotic nature of capillary vascularization—may introduce inaccuracies in the predictions.

In Chapter 7, we approached the solution of the wave equation with application to the prediction of deformable tissue with PINNs. This represents a novelty not only with respect to conventional solvers for the wave equation but also to techniques already in use in DL. The networks showed some limitations connected to the high-frequency behaviour of the system. Different strategies to tackle these hurdles emerged: from random Fourier features embedding of the inputs, to the adaptive loss balancing, and the weights factorisation.

8.2 Future works

Hyperthermia. In the context of solving Pennes’ Bio-Heat Equation for HT application, we are studying the thermal characterization of the applicator based on the correlation between temperature rise and SAR. Extensions of the one-dimensional method include considering a multilayer composition, as well as a variable perfusion term $w_b(X, \tau)$. This can be achieved by adopting more advanced perfusion models, such as in Ref. [32, 29, 27, 16]. Extension to the two-dimensional case has been already validated for the direct problem.

Deformation of tissues. Regarding the solution of wave equation, the main focus is on the extension to the two-dimensional case, with a certain initial velocity distribution representing an impulse on the membrane at the initial time and the possibility of augmenting the input dimensionality to account for the module and position of the impulse. This enables the possibility to apply PINN to control problems (PINC). We are currently investigating more complex architectures, specifically Graph Networks to simulate physics on complex geometries, in combination with RNN to better capture the evolution in time. Moreover, the integration of this system into a virtual reality environment, incorporating haptic feedback for the training of medical procedures, is currently underway. Real measurements can be obtained from the use of piezoelectric sensorised fabrics.

Appendix A

HPO of PINNs

A.1 Choice of sampling points

Here it is reported an optimization on the choice of sampling points n_{res} , n_b for the NBHS and the NBHO. For the NBHS, grid search (See 3.9.2) was performed, assigning three possible values to $n_{\text{res}}, n_b \in \{200, 500, 1000\} \times \{200, 500, 1000\}$. Additionally, the effect of resampling (RR) of n_{res}, n_b every 100 iterations is evaluated on every possible combination of n_{res}^i, n_b^i , yielding a total of 18 runs. Results are reported in Figure A.1 and in Figure A.2 with a different random seed initialization.

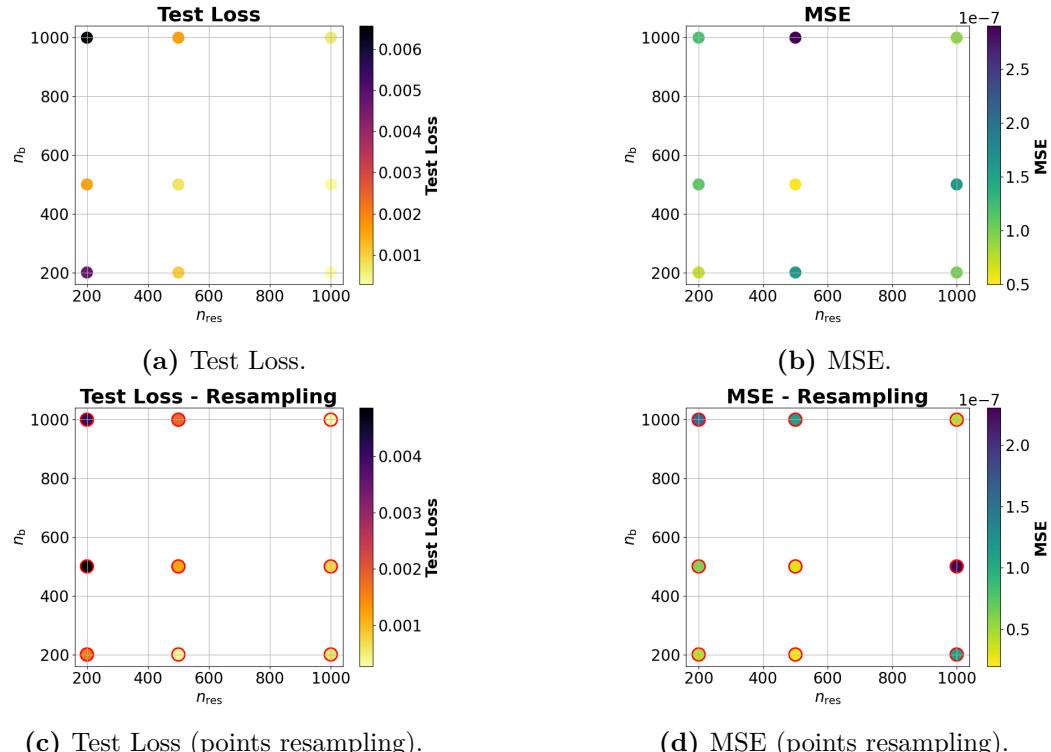


Figure A.1. Tuning the Sampling Points n_{res} and n_b for the NBHS. Scatter plots for resampled (red circle) and non-resampled training points, showing Test Loss and MSE as color maps.

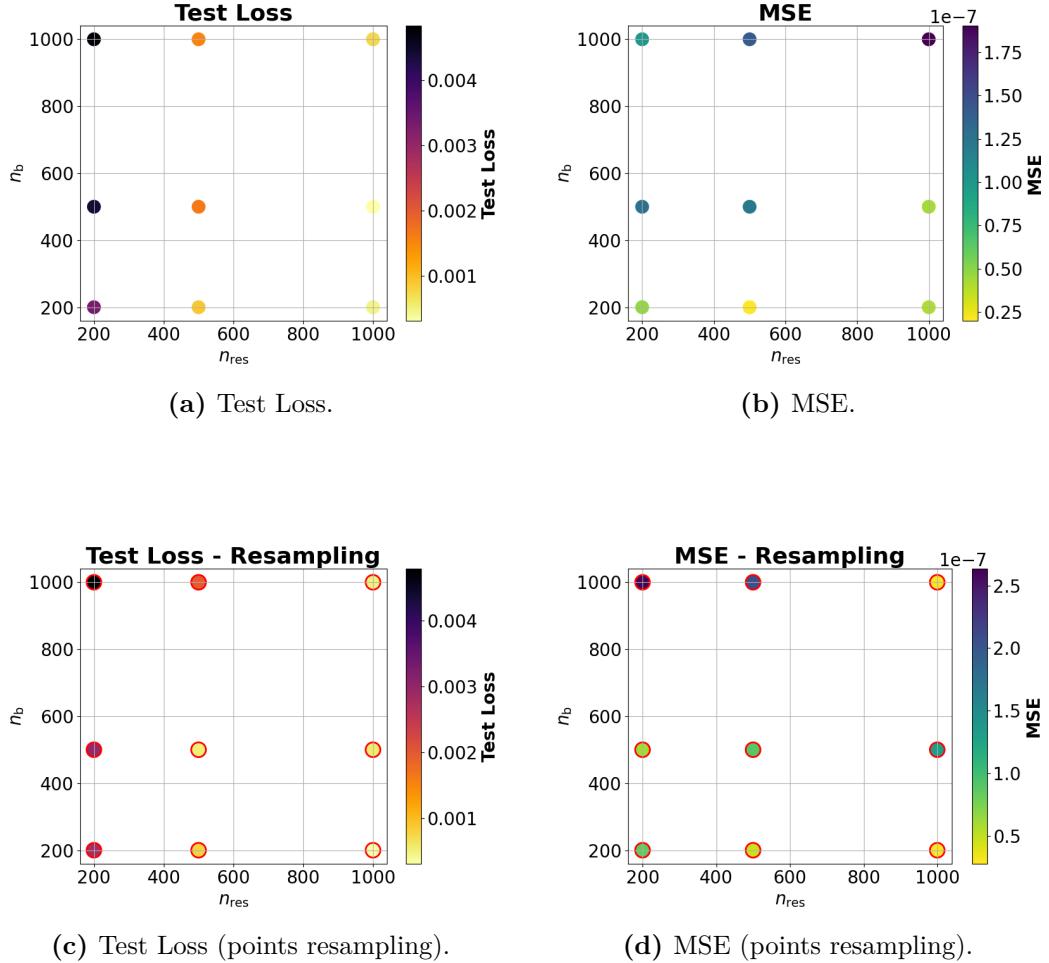


Figure A.2. The procedure of the choice of sampling points for NBHS is repeated with a different random seed initialization.

Also for the NBHO, the approach consisted of a grid search over five possible values for $n_{\text{res}}, n_b \in \{200, 500, 1000, 2000, 5000\} \times \{200, 500, 1000, 2000, 5000\}$. Additionally, the effect of resampling RR of n_{res}, n_b every 100 iterations is evaluated on every possible combination of n_{res}^i, n_b^i , yielding a total of 50 runs. Results are reported in Figure A.3.

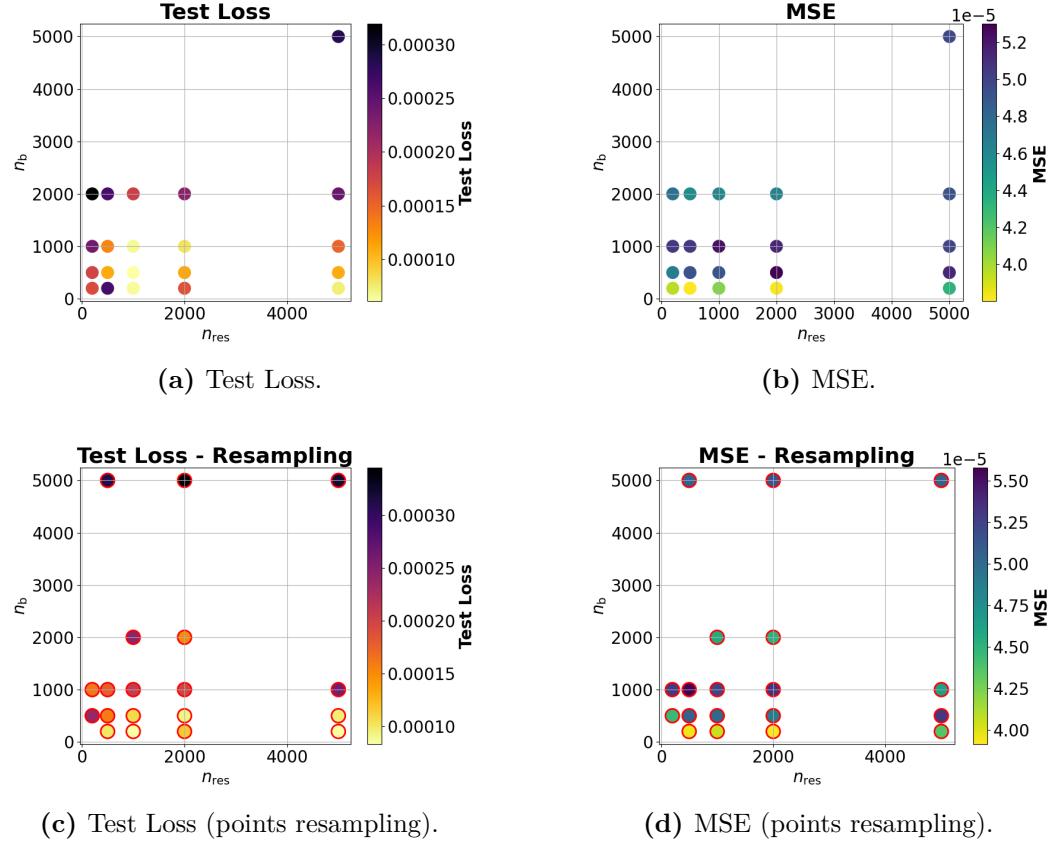


Figure A.3. Tuning the Sampling Points n_{res} and n_b , NBHO. Scatter plots for resampled (red circle) and non-resampled training points, showing Test Loss and MSE as color maps.

A.2 Optimization of PINN architecture

Figure A.4 reports a study on the influence of activation function, width, and depth of NBHS and NBHO on train loss.

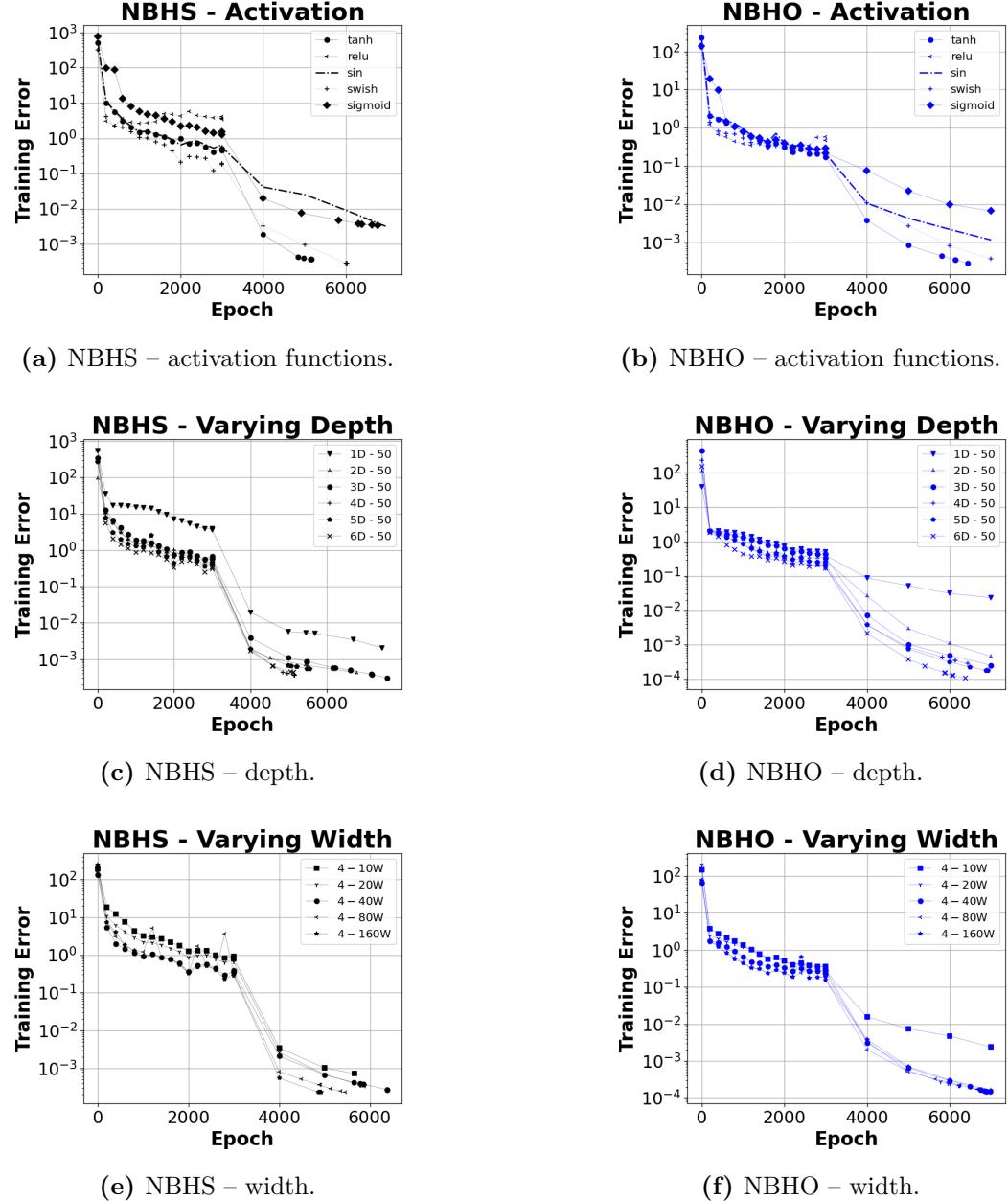


Figure A.4. Training error comparisons under different network configurations for PINNs: (top) activation functions, (middle) depth (D1–D6: 1–6 hidden layers, 50 neurons each), (bottom) width (W10–W160: number of neurons per layer with 4 hidden layers).

A.3 Choice of output injection gain for the NBHO

The following analysis examines the impact of varying output injection gain K values on the NBHO performance, specifically assessing network accuracy (MSE) and test loss metrics. With ground truth solutions established using MATLAB simulations on the problem formulated in Chapter 5, our findings demonstrate the robust predictive capabilities of the NBHO framework across a range of K values, indicating low sensitivity to this hyperparameter within the tested domain.

Table A.1. Tuning the K for the NBHO.

| K | Runtime [s] | Test Loss | MSE |
|-----------------------|-------------|------------------|------------|
| 0.42 | 52.0 | $5.8e-5$ | $1.4e-4$ |
| 1.05 | 48.9 | $8.4e-5$ | $1.2e-4$ |
| 2.45 | 57.9 | $8.7e-5$ | $2.8e-5$ |
| 0.7 | 55.7 | $8.8e-5$ | $1.0e-4$ |
| 1.75 | 53.7 | $1.0e-4$ | $6.0e-5$ |
| 2.1 | 48.0 | $1.2e-4$ | $6.3e-5$ |
| 1.54 | 47.7 | $1.4e-4$ | $1.2e-4$ |
| 0.91 | 46.3 | $1.5e-4$ | $1.0e-4$ |
| 0.28 | 43.7 | $1.6e-4$ | $1.8e-4$ |
| 2.24 | 53.0 | $1.9e-4$ | $3.4e-5$ |
| 0.21 | 58.5 | $2.1e-4$ | $1.7e-4$ |
| 0.07 | 52.7 | $2.2e-4$ | $2.2e-4$ |
| 1.26 | 60.7 | $2.2e-4$ | $7.6e-5$ |
| 0.14 | 50.7 | $2.4e-4$ | $1.8e-4$ |
| 0.35 | 48.0 | $2.4e-4$ | $1.7e-4$ |
| 0.49 | 35.6 | $2.6e-4$ | $1.7e-4$ |
| 0.56 | 43.7 | $2.2e-3$ | $1.1e-4$ |

Appendix B

Companion figures

This appendix presents a comprehensive analysis of the MM-Obs implementation using PINNs through a family of NBHOs applied to the Cooling 1 and Cooling 2 measurement sets detailed in Chapter 6. We evaluate the performance for an increasing number of observers $n_{\text{obs}} \in \{8, 16, 64\}$.

The analysis includes comparative visualizations of predicted temperature distributions at various time points (Figure B.1 and Figure B.4), quantification of temperature observation errors both at the surface and at the ground truth position X_{gt} (Figure B.2 and Figure B.5), and temporal evolution plots comparing system measurements with corresponding network predictions (Figure B.3 and Figure B.6).

B.1 Cooling 1

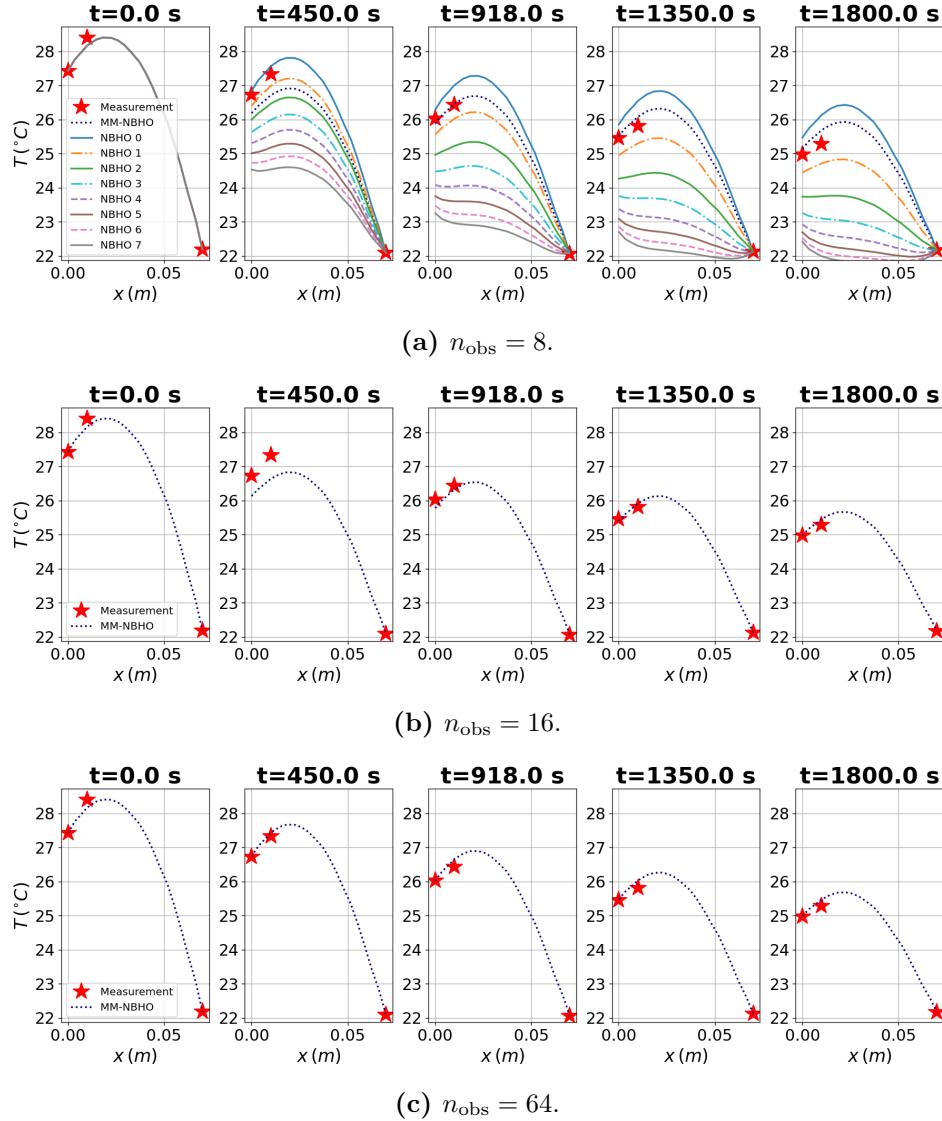


Figure B.1. Observers' and Multi-Observer's predictions at different instants, Cooling 1, varying number of observers.

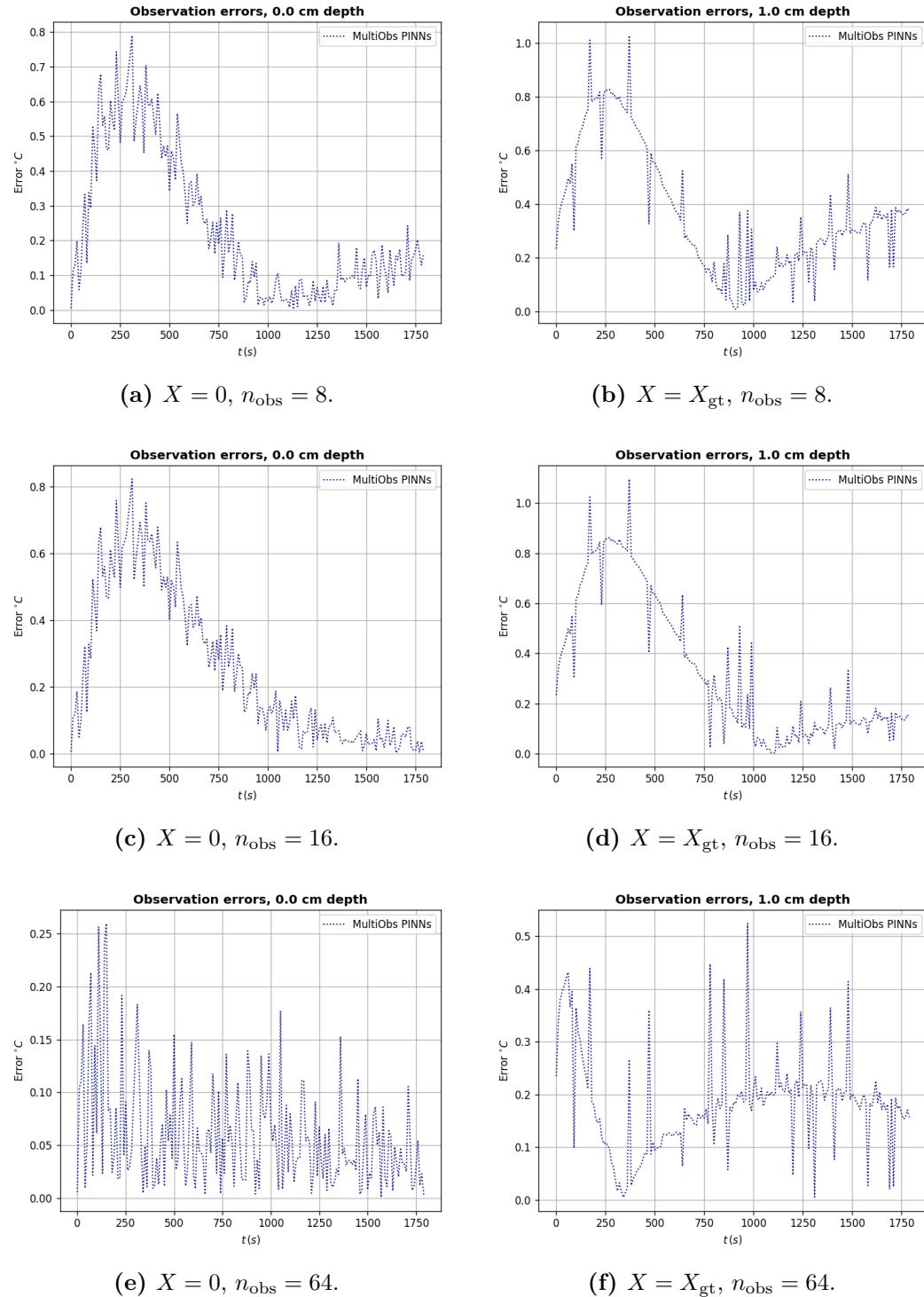


Figure B.2. PINNs observation errors at different X positions and varying number of observers during Cooling 1.

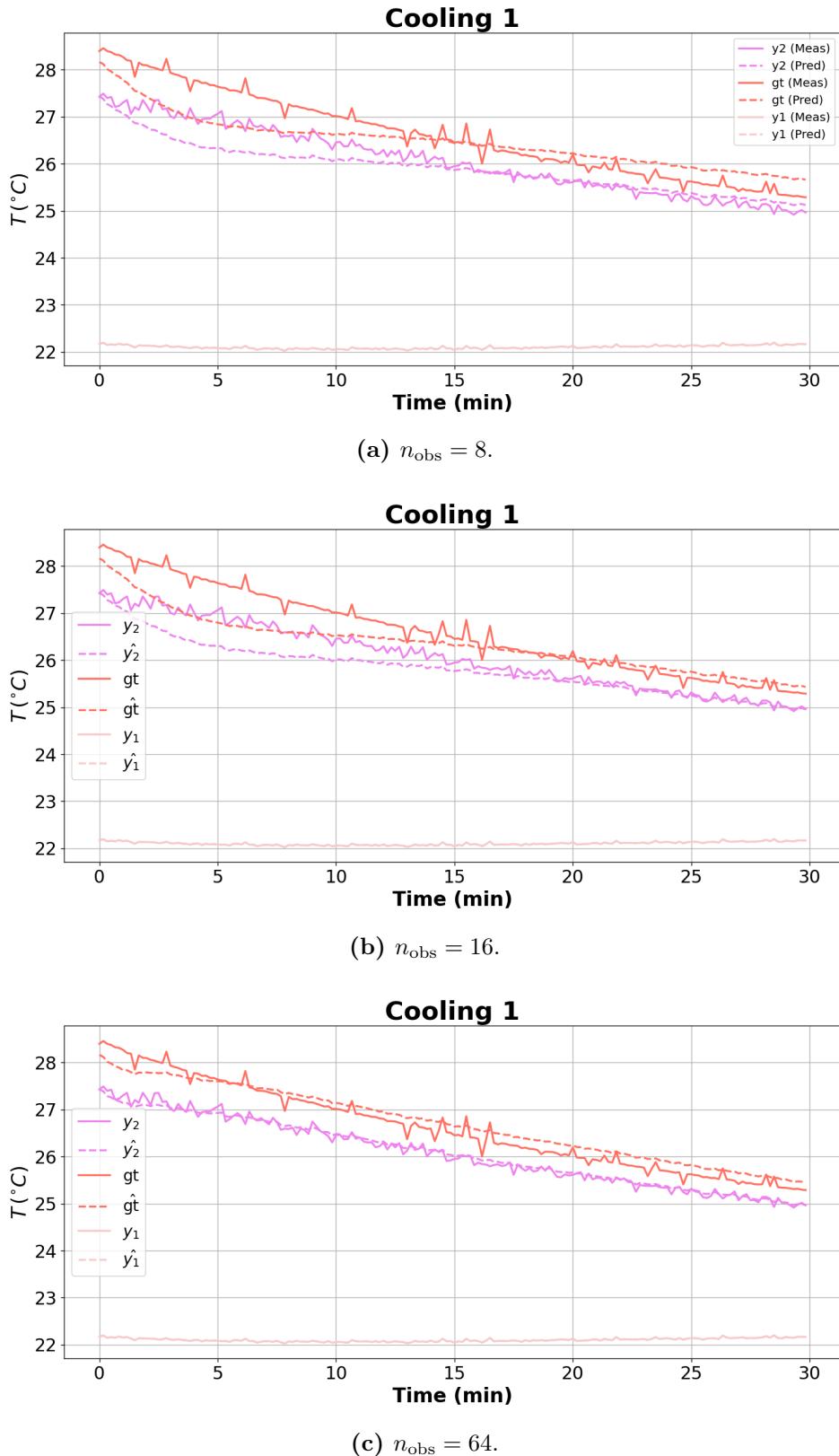


Figure B.3. Time evolution of system measurements and corresponding MM-Obs estimates during Cooling 1 for different numbers of observers.

B.2 Cooling 2

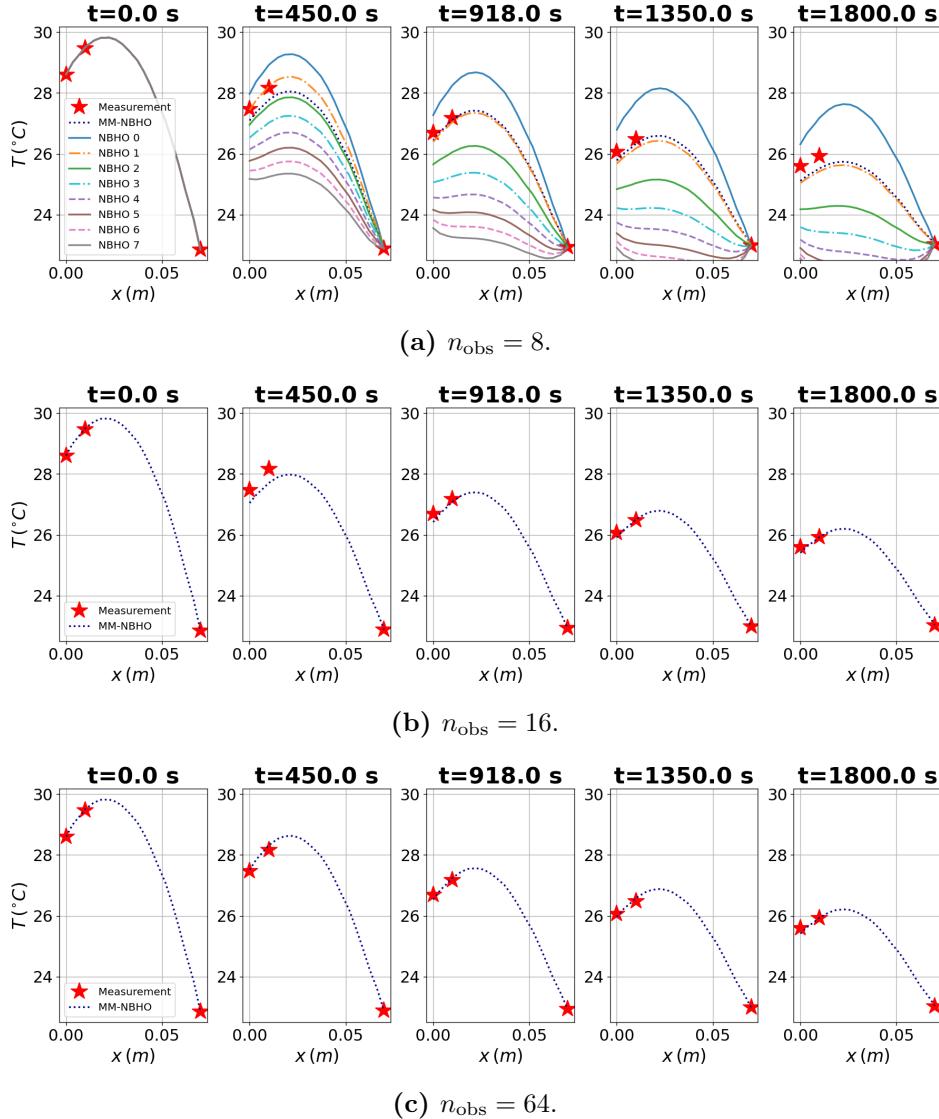


Figure B.4. Observers' and Multi-Observer's predictions at different instants, Cooling 2, varying number of observers.

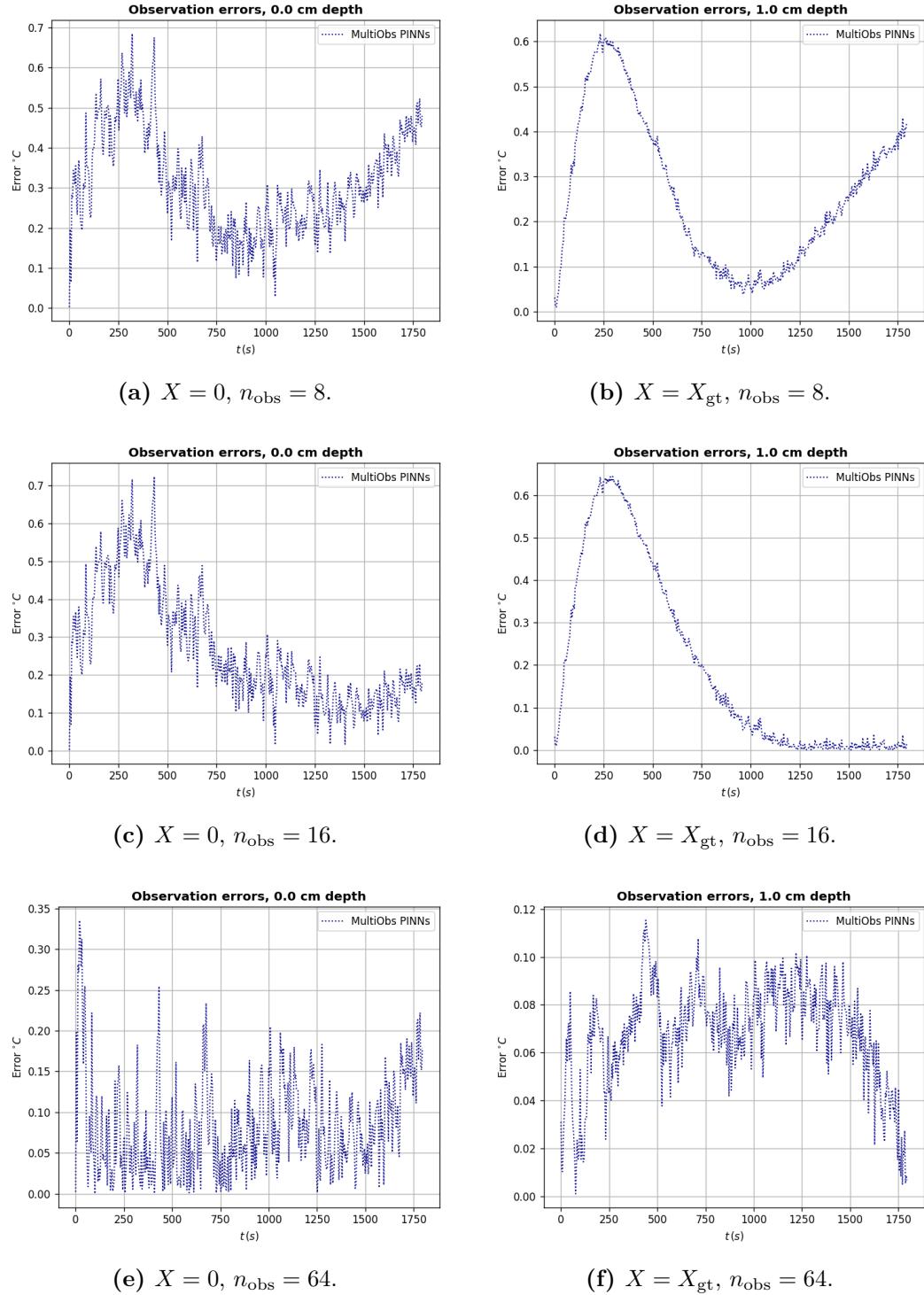


Figure B.5. PINNs observation errors at different X positions and varying number of observers during Cooling 2.

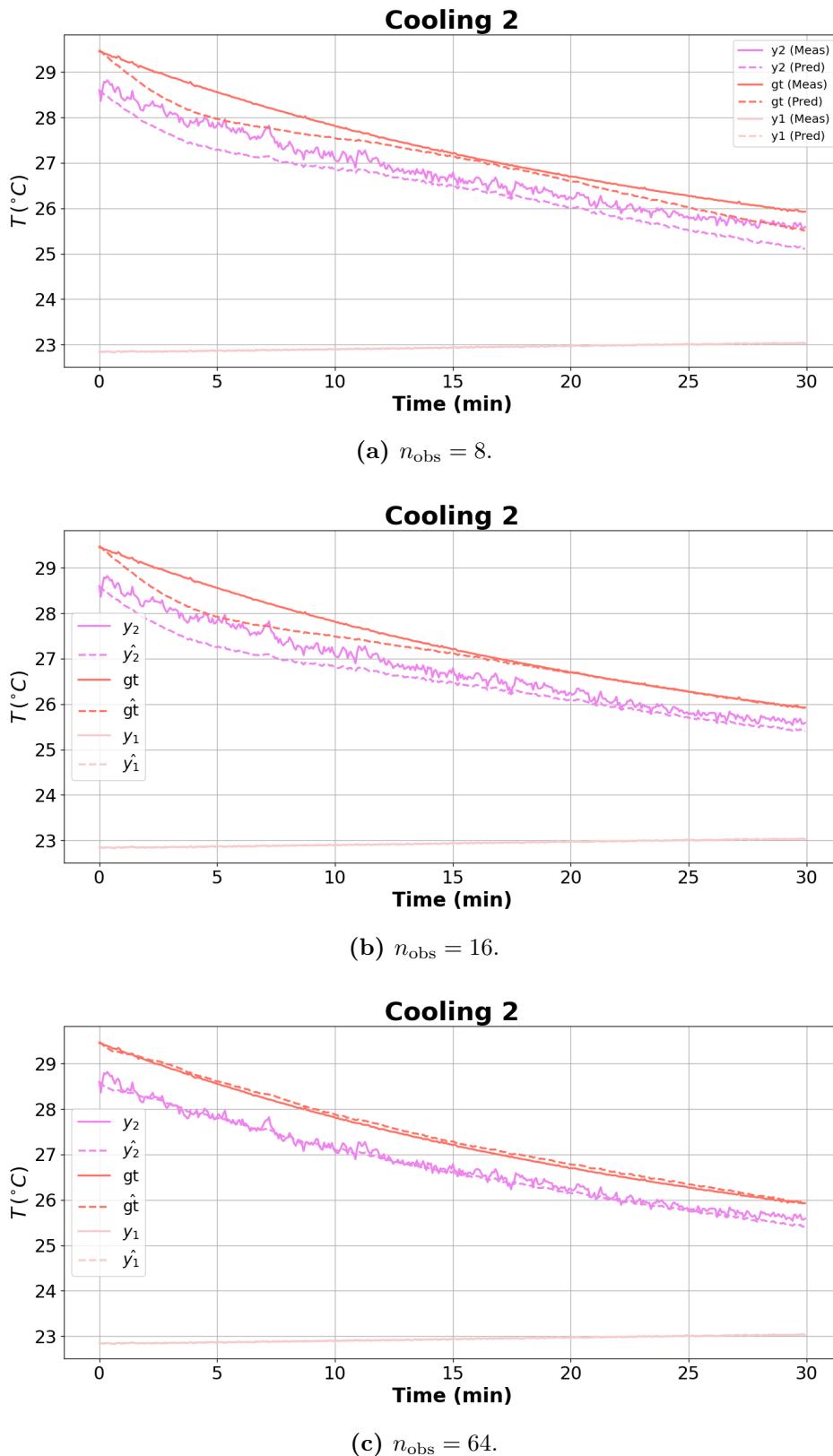


Figure B.6. Time evolution of system measurements and corresponding MM-Obs estimates during Cooling 2 for different numbers of observers.

Appendix C

Abstract

Questa tesi affronta una sfida cruciale nell'ambito dell'ipertermia oncologica superficiale: la valutazione non invasiva ed in tempo reale della temperatura al target. Il trattamento viene utilizzato in combinazione con la Radioterapia o la Chemoterapia, specialmente per recidive e i tumori più resistenti, con l'obiettivo di aumentarne l'efficacia senza somministrare ulteriore dose al paziente. L'efficacia è strettamente connessa con il raggiungimento delle temperature prestabilite in maniera ripetibile tra i trattamenti. Allo stato attuale, l'unico metodo per monitorare il trattamento è rappresentato dalla misurazione invasiva tramite l'uso di termocoppie, un metodo che spesso viene evitato, poiché comporta notevole disagio per i pazienti e fornisce solo misurazioni discrete e difficili da localizzare. La situazione è ulteriormente complicata dalla presenza di incertezze nelle proprietà specifiche del paziente, in particolare nel tasso di perfusione sanguigna, che non è direttamente misurabile e varia durante il trattamento, influenzando notevolmente la distribuzione di temperatura.

L'obiettivo consiste nell'ottenere predizioni della distribuzione della temperatura in tutto il dominio soggetto al trattamento utilizzando solo misurazioni al contorno, i dati clinici del setup e l'anatomia del paziente. La metodologia sviluppata consiste nella stima adattiva dell'equazione utilizzando un osservatore multi-modello. Il processo di validazione ha messo in evidenza alcune limitazioni dei tradizionali metodi di simulazione numerica, che hanno motivato il ricorso a tecniche di Deep Learning. In particolare, il Physics-Informed Learning si è dimostrato essere un approccio particolarmente adatto a scenari in cui le leggi fisiche sono note, mentre i dati per il training sono limitati ad alcuni punti o del tutto assenti. L'approccio Physics-Informed si distingue per tre vantaggi fondamentali: elimina la necessità di discretizzare il dominio spazio-temporale, permettendo un campionamento continuo; in secondo luogo, dopo un periodo di training, la soluzione alle equazioni è fornita in tempo reale; infine, lo stesso modello può incorporare variazioni dei parametri dell'equazione come input, garantendo così un'elevata adattabilità a diversi scenari clinici e anatomici.

L'ultima parte di questo studio si concentra sulla risoluzione dell'equazione d'onda per la simulazione di materiali morbidi, con applicazioni nella formazione chirurgica in ambienti di realtà mista. Questa ricerca interdisciplinare integra matematica applicata, informatica e ingegneria per affrontare complesse sfide cliniche.

Acronyms

| | |
|---------------|---------------------------------------------------------------------------------|
| SAR | Specific Absorption Rate. 1, 9, 11, 12, 96 |
| AD | Automatic Differentiation. 26, 27, 30, 31, 82 |
| Adam | Adaptive Moment Estimation. 27, 28, 52, 55, 90 |
| AI | Artificial Intelligence. 18–20, 29, 65, 95, 96 |
| Amsterdam UMC | Amsterdam University Medical Center. ii, 65, 96 |
| BC | Boundary Condition. 8, 10, 11, 13, 30, 31, 36, 38–41, 50–52, 58, 82, 87, 88, 92 |
| BP | Backpropagation. 26, 27 |
| CH | Chemotherapy. 1, 5, 17, 95 |
| CNN | Convolutional Neural Network. 29 |
| DCWG | Dynamic Convex Weighting Generator. 35, 45, 46, 57, 59, 61, 62 |
| DL | Deep Learning. 19, 26–28, 31, 32, 34, 49, 89, 96 |
| DSL | Domain-Specific Language. 50, 90 |
| EFS | Effective Field Size. 10, 69 |
| EHD | Effective Heating Depth. 10 |
| EM | Electromagnetic. 1, 6, 7, 9, 39, 46, 68, 80 |
| ESHO | European Society for Hyperthermic Oncology. 9 |
| FDTD | Finite-Difference Time-Domain. 2, 85, 86, 95 |
| FNN | Feedforward Neural Network. 24, 26, 88 |
| GTV | Gross Tumor Volume. 65, 66 |
| HP | Hyper-parameter. 25, 28, 31–33, 52, 73 |
| HPO | Hyper-parameters optimization. 25, 32, 33 |

| | |
|--------|---------------------------------------------------------------------------------------------------------------------------|
| HT | Hyperthermia. 1, 2, 5–8, 12, 16–18, 35, 49, 59, 65, 66, 95, 96 |
| HTP | Hyperthermia Treatment Planning. 5, 16, 49 |
| IC | Initial Condition. 8, 11, 13, 31, 38, 47, 51, 52, 87, 88, 92, 93 |
| IWR | Initial Weights Regularization. 33, 53, 74 |
| KAN | Kolmogorov-Arnold Network. 29 |
| L-BFGS | Limited-memory Broyden–Fletcher–Goldfarb–Shanno. 27, 28, 52, 55 |
| LS | Least Squares. 23 |
| LSTM | Long Short-Term Memory Network. 29 |
| ML | Machine Learning. 2, 19–21 |
| MLP | Multilayer Perceptron. 24, 26, 29, 30, 89, 90, 92 |
| MM-Obs | Multiple-model Adaptive Observer. 2, 38, 45–47, 62, 65, 72, 76, 77, 81–83, 103, 106, 109 |
| MR | Magnetic Resonance. 29 |
| NBHO | Neural Bio-Heat Observer. 50, 57, 70, 72–74, 77, 83, 97, 99–101, 103 |
| NBHS | Neural Bio-Heat System. 50–57, 97, 98, 100 |
| NN | Neural Network. 2, 19, 24–30, 34, 63, 88, 93, 96 |
| NTK | Neural Tangent Kernel. 92 |
| ODE | Ordinary Differential Equation. 35, 36, 46, 57, 62, 63, 95 |
| PBHE | Pennes’ Bio-Heat Equation. 9–12, 14, 16, 17, 35, 38, 39, 47, 49–51, 54, 55, 62, 63, 66, 83, 95 |
| PCE | Polynomial Chaos Expansion. 16 |
| PD | Penetration Depth [m]. 10, 69, 73, 74 |
| PDE | Partial Differential Equation. 9, 11, 19, 29–31, 35–40, 46, 47, 49, 50, 57, 62, 63, 85, 87–90, 92–95 |
| PINN | Physics-Informed Neural Network. 12, 19, 27–33, 49–51, 55, 57, 59, 62, 63, 65, 76, 82, 83, 85, 86, 88–90, 92–96, 100, 103 |

| | |
|---------|---------------------------------------------------------|
| PNRR | Italian National Recovery and Resilience Plan. 86 |
| RAR | Residual-based Adaptive Refinement. 34 |
| ReLU | Rectified Linear Units. 27, 29, 33 |
| RNN | Recurrent Neural Network. 29, 96 |
| RT | Radiotherapy. 1, 5, 6, 17, 95 |
| SGD | Stochastic Gradient Descent. 25 |
| STMsFFN | Spatio-temporal Multi-scale Fourier Feature Network. 90 |
| TC | Thermocouples. 1, 11, 66, 68, 73 |
| TMS | Thermal Monitoring Sheet. 1, 68 |

Glossary

| | |
|------------------------|-------------------------------------------------------------------|
| K | Output injection gain [-]. 39–41, 43, 45, 46, 58, 73, 74, 82, 101 |
| $L - 1$ | Depth [-]. 24, 25, 32, 33, 53, 74 |
| L_0 | Length of the domain [m]. 9, 11–13, 50, 70, 73, 74, 82, 87, 90 |
| N | Width [-]. 24, 25, 32, 33, 53, 74 |
| T_B | Steady-state temperature of the biological tissue [°C]. 11 |
| T_a | Arterial blood temperature [°C]. 9, 11, 12, 14 |
| T_f | Temperature of the fluid [°C]. 10, 11 |
| T_w | Temperature of the wall [°C]. 10, 11 |
| T_{R2} | Temperature at R2 [°C]. 15, 16 |
| T_{\max} | Maximum temperature [°C]. 8, 12, 50, 51, 67, 73, 74 |
| T_{\min} | Minimum temperature [°C]. 12, 16, 50, 51, 57, 67, 73, 74 |
| T | Temperature [°C]. 9–12, 50 |
| X_{gt} | Ground truth position [-]. 13, 51, 54–56, 66, 77, 103 |
| Γ | HP search space. 33 |
| Λ | Decay rate for known perfusion [s^{-1}]. 41, 43, 44 |
| Ω | Computational domain. 8–11, 30, 31, 82 |
| Φ | Activation function. 24, 25, 27, 32, 33, 53, 74 |
| Υ_0 | Initialization. 25, 26, 28, 33, 53, 74 |
| Υ | Weights and biases. 23, 24, 26, 30, 31, 33 |
| Ξ | Decay rate for unknown perfusion [s^{-1}]. 43, 44 |
| β | Fitting parameter [-]. 9, 12, 73, 74 |
| δ_B | Characteristic length of bioheat transfer [m]. 11, 13 |
| η | Learning rate [-]. 22, 25, 26, 32, 33, 53, 74 |
| γ | HP set. 33 |
| $\hat{\theta}_0^{(j)}$ | Initial condition of the observer. 40, 46, 47, 58, 59 |

| | |
|------------------------|--------------------------------------------------------------------|
| $\hat{\theta}^{(j)}$ | Dimensionless j-th observer. 45, 46, 58 |
| $\hat{\theta}^\dagger$ | Dimensionless multiple-model adaptive observer. 45, 46, 58, 77, 82 |
| $\hat{\theta}$ | Dimensionless observer. 39–41, 46, 47, 62, 82 |
| λ | Adaptive gain [-]. 45, 46, 58, 59, 62 |
| W | Weight matrix. 24 |
| b | Bias vector. 24 |
| \mathcal{L}^2 | L2 error. 37, 38, 61, 62, 79 |
| \mathcal{L} | Loss function. 23, 25, 26, 28, 30, 31, 33 |
| \mathcal{N} | Differential operator. 8, 30 |
| \mathcal{Q} | Heat source [W m^{-3}]. 9, 12 |
| \mathcal{T} | Set of sampling points. 23, 25, 30, 31 |
| \mathcal{W} | Warmstart. 55, 56 |
| E | Young's modulus [Nm^{-2}]. 87, 88 |
| RR | Resampling. 53, 74, 97, 99 |
| F | Tension [N]. 87 |
| μ | Absolute output error [-]. 45, 46, 58, 59 |
| \bar{m} | Mass per unit length [kg m^{-1}]. 87 |
| ψ | Parameters. 30, 31, 55 |
| σ | Electric conductivity [Sm^{-1}]. 9 |
| w_{bc} | Weight of boundary loss [-]. 30–32, 53, 74 |
| w_{res} | Weight of residual loss [-]. 30–32, 53, 74 |
| θ_0 | Scaled initial condition. 13, 38, 39, 47, 51, 57 |
| θ | Dimensionless temperature [-]. 12, 13, 38–40, 42, 50, 51 |
| \tilde{d} | Initial condition function. 8 |
| \tilde{g} | Boundary condition function. 8, 30, 31 |
| $\tilde{y}_1(0)$ | Initial in-depth temperature [$^\circ\text{C}$]. 73, 74 |
| \tilde{y}_1 | In-depth temperature [$^\circ\text{C}$]. 11, 12 |
| $\tilde{y}_2(0)$ | Initial superficial temperature [$^\circ\text{C}$]. 73, 74 |
| \tilde{y}_2 | Superficial temperature [$^\circ\text{C}$]. 11, 12 |
| $\tilde{y}_3(0)$ | Initial fluid temperature [$^\circ\text{C}$]. 73, 74 |
| \tilde{y}_3 | Fluid temperature [$^\circ\text{C}$]. 11, 12 |
| ϱ_b | Density of blood [kg m^{-3}]. 9, 11, 12, 15 |
| ϱ | Density [kg m^{-3}]. 9, 11–13, 68, 73, 74, 82 |
| ξ | Poincaré-Wirtinger inequality constant [-]. 41–44 |
| ζ | Output transform. 51, 52, 57 |
| a_1 | PBHE constant [-]. 12, 13, 38–43, 45, 50, 51, 58, 82 |
| a_2 | PBHE constant [-]. 12, 13, 38–41, 45, 50, 51, 58, 82 |
| a_3 | PBHE constant [-]. 12, 13, 38, 39, 41, 45, 50, 51, 58, 82 |

| | |
|-------------------|---------------------------------------------------------------------------------------------------------------|
| a_4 | PBHE constant [-]. 12, 13, 38, 39, 41, 45, 50, 51, 58, 82 |
| a_5 | PBHE constant [-]. 12, 13, 38, 39, 50, 51, 58 |
| a | Velocity of the wave [m s^{-1}]. 87, 88 |
| b_1 | IC system constant [-]. 13, 38, 51 |
| b_2 | IC system constant [-]. 13, 38, 51 |
| b_3 | IC system constant [-]. 13, 38, 51 |
| b_4 | IC system constant [-]. 13, 38, 51 |
| c_0 | Bounding error [-]. 42–44 |
| c_1 | IC observer constant [-]. 47, 58, 59, 73, 74 |
| c_2 | IC observer constant [-]. 47, 58, 59, 73, 74 |
| c_3 | IC observer constant [-]. 47, 58, 59, 73, 74 |
| c_b | Specific heat of blood [$\text{J kg}^{-1} \text{K}^{-1}$]. 9, 11, 12, 15 |
| c | Specific heat [$\text{J kg}^{-1} \text{K}^{-1}$]. 9, 11–13, 16, 68, 73, 74, 82 |
| g_0 | Scaled superficial boundary condition function. 39, 40, 45–47, 69 |
| h | Convection coefficient [$\text{W m}^{-2} \text{K}^{-1}$]. 10–13, 15, 70, 73, 74 |
| k_b | Thermal conductivity of blood [$\text{W m}^{-1} \text{K}^{-1}$]. 15 |
| k_{eff} | Effective thermal conductivity [$\text{W m}^{-1} \text{K}^{-1}$]. 14, 15 |
| k | Thermal conductivity [$\text{W m}^{-1} \text{K}^{-1}$]. 9–13, 15, 16, 68, 73, 74, 82 |
| n_b | Number of boundary sampling points [-]. 53, 74, 97, 99 |
| n_{res} | Number of residual sampling points [-]. 53, 74, 97, 99 |
| n_{test} | Number of test points [-]. 53, 74 |
| n_{anc} | Number of anchor points [-]. 53, 55, 56, 74 |
| n_{ins} | Number of inputs [-]. 53, 74 |
| n_{obs} | Number of observers [-]. 76–78, 103–109 |
| q_w | Heat flux [W m^{-2}]. 10 |
| q_{met} | Metabolic heat generation rate [W m^{-3}]. 9, 11 |
| t^* | Time span [s]. 12, 40–43, 73, 74 |
| v | Velocity [cm s^{-1}]. 14, 15, 70 |
| w_b | Blood perfusion rate [s^{-1}]. 9, 11–13, 35, 37–45, 50, 51, 55, 56, 58, 59, 62, 69, 76, 82, 83, 96 |
| w_b^{\max} | Maximum blood perfusion rate [s^{-1}]. 45 |
| w_b^{\min} | Minimum blood perfusion rate [s^{-1}]. 45 |
| y_1 | Scaled in-depth temperature [-]. 11, 13, 39, 40, 45–47, 50, 51, 57, 58, 69, 73, 82, 83 |

- y_2 Scaled superficial temperature [-]. 11, 13, 39, 40, 45–47, 50, 51, 57, 58, 69, 73
 y_3 Scaled fluid temperature [-]. 11, 13, 39, 50, 51, 57, 58, 73
P Power [W]. 9, 13, 38, 39, 41, 45, 46, 50, 51, 58, 69, 73, 74

Bibliography

- [1] ABADI, M. ; BARHAM, P. ; CHEN, J. ; CHEN, Z. ; DAVIS, A. ; DEAN, J. ; DEVIN, M. ; GHEMAWAT, S. ; IRVING, G. ; ISARD, M. ; KUDLUR, M. ; LEVENBERG, J. ; MONGA, R. ; MOORE, S. ; MURRAY, D. G. ; STEINER, B. ; TUCKER, P. A. ; VASUDEVAN, V. ; WARDEN, P. ; WICKE, M. ; YU, Y. ; ZHANG, X. : TensorFlow: A system for large-scale machine learning. In: *CoRR* abs/1605.08695 (2016). <http://arxiv.org/abs/1605.08695>
- [2] ARUNACHALAM, K. ; MACCARINI, P. F. ; CRACIUNESCU, O. I. ; SCHLORFF, J. L. ; STAUFFER, P. R.: Thermal characteristics of thermobrachytherapy surface applicators for treating chest wall recurrence. In: *Phys Med Biol* 55 (2010), Apr, Nr. 7, S. 1949–1969. <http://dx.doi.org/10.1088/0031-9155/55/7/011>. – DOI 10.1088/0031-9155/55/7/011. – ISSN 1361–6560 (Electronic); 0031–9155 (Print); 0031–9155 (Linking)
- [3] ASCENCIO, P. ; ASTOLFI, A. ; PARISINI, T. : An adaptive observer for a class of parabolic PDEs based on a convex optimization approach for backstepping PDE design. In: *2016 American Control Conference (ACC)*, 2016, S. 3429–3434
- [4] ASH, J. T. ; ADAMS, R. P.: On Warm-Starting Neural Network Training. (2019). <http://dx.doi.org/10.48550/ARXIV.1910.08475>. – DOI 10.48550/ARXIV.1910.08475
- [5] ASSEN, M. van ; MUSCOGIURI, G. ; CARUSO, D. ; LEE, S. J. ; LAGHI, A. ; DE CECCO, C. N.: Artificial intelligence in cardiac radiology. In: *La radiologia medica* 125 (2020), Nr. 11, 1186–1199. <http://dx.doi.org/10.1007/s11547-020-01277-w>. – DOI 10.1007/s11547-020-01277-w. ISBN 1826–6983
- [6] BAKKER, A. ; ZWEIJ, R. ; KOK, H. P. ; KOLFF, M. W. ; BONGARD, H. J. G. D. d. ; SCHMIDT, M. ; TIENHOVEN, G. van ; CREZEE, H. : Clinical Feasibility of a High-Resolution Thermal Monitoring Sheet for Superficial Hyperthermia in Breast Cancer Patients. In: *Cancers* 12 (2020), Nr. 12. <http://dx.doi.org/10.3390/cancers12123644>. – DOI 10.3390/cancers12123644. – ISSN 2072–6694
- [7] BERDOV, B. A. ; MENTESHASHVILI, G. Z.: Thermoradiotherapy of patients with locally advanced carcinoma of the rectum. In: *Int J Hyperthermia* 6 (1990), Sep-Oct, Nr. 5, S. 881–890. <http://dx.doi.org/10.3109/02656739009140970>. – DOI 10.3109/02656739009140970. – ISSN 0265–6736 (Print); 0265–6736 (Linking)

- [8] CAI, S. ; WANG, Z. ; WANG, S. ; PERDIKARIS, P. ; KARNIADAKIS, G. E.: Physics-Informed Neural Networks for Heat Transfer Problems. In: *Journal of Heat Transfer* 143 (2021), 04, Nr. 6, 060801. <http://dx.doi.org/10.1115/1.4050542>. – DOI 10.1115/1.4050542. – ISSN 0022–1481
- [9] CAPPELLINI, G. ; TRAPPOLINI, G. ; STAFFETTI, E. ; CRISTOFARO, A. ; VENDITTELLI, M. : Adaptive Estimation of the Pennes' Bio-Heat Equation - II: A NN-Based Implementation for Real-Time Applications. In: *2023 62nd IEEE Conference on Decision and Control (CDC)*, 2023, S. 5364–5369
- [10] CREZEE, J. ; LAGENDIJK, J. J.: Temperature uniformity during hyperthermia: the impact of large vessels. In: *Physics in medicine and biology* 37 (1992), Jun, Nr. 6, S. 1321–1337. <http://dx.doi.org/10.1088/0031-9155/37/6/009>. – DOI 10.1088/0031–9155/37/6/009. – ISSN 0031–9155
- [11] CREZEE, J. ; LAGENDIJK, J. J. W.: Experimental verification of bioheat transfer theories: measurement of temperature profiles around large artificial vessels in perfused tissue. In: *Physics in Medicine & Biology* 35 (1990), jul, Nr. 7, S. 905
- [12] CRISTOFARO, A. ; CAPPELLINI, G. ; STAFFETTI, E. ; TRAPPOLINI, G. ; VENDITTELLI, M. : Adaptive Estimation of the Pennes' Bio-Heat Equation - I: Observer Design. In: *2023 62nd IEEE Conference on Decision and Control (CDC)*, 2023, S. 1931–1936
- [13] CYBENKO, G. : Approximation by superpositions of a sigmoidal function. In: *Mathematics of Control, Signals and Systems* 2 (1989), Nr. 4, 303–314. <http://dx.doi.org/10.1007/BF02551274>. – DOI 10.1007/BF02551274. ISBN 1435–568X
- [14] DATTA, N. ; ORDÓÑEZ, S. G. ; GAIPL, U. ; PAULIDES, M. ; CREZEE, H. ; GELLERMANN, J. ; MARDER, D. ; PURIC, E. ; BODIS, S. : Local hyperthermia combined with radiotherapy and-/or chemotherapy: Recent advances and promises for the future. In: *Cancer Treatment Reviews* 41 (2015), Nr. 9, S. 742–753. <http://dx.doi.org/https://doi.org/10.1016/j.ctrv.2015.05.009>. – DOI <https://doi.org/10.1016/j.ctrv.2015.05.009>. – ISSN 0305–7372
- [15] DEWEY, W. C. ; HOPWOOD, L. E. ; SAPARETO, S. A. ; GERWECK, L. E.: Cellular Responses to Combinations of Hyperthermia and Radiation. In: *Radiology* 123 (1977), Nr. 2, S. 463–474
- [16] DRIZDAL, T. ; TOGNI, P. ; VISEK, L. ; VRBA, J. : Comparison of Constant and Temperature Dependent Blood Perfusion in Temperature Prediction for Superficial Hyperthermia. In: *Radioengineering* 19 (2010), 06
- [17] DUCHI, J. ; HAZAN, E. ; SINGER, Y. : Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. In: *Journal of Machine Learning Research* 12 (2011), Nr. 61, S. 2121–2159
- [18] ESCAPIL-INCHAUSPÉ, P. ; RUZ, G. A.: *Hyper-parameter tuning of physics-informed neural networks: Application to Helmholtz problems*. <http://dx.doi.org/10.48550/ARXIV.2205.06704>. Version: 2022

- [19] FERRO, M. ; PAVONI, P. ; VENDITELLI, M. : Robot-assisted superficial hyperthermia treatments: the ROBHOT system. In: *2024 IEEE 20th International Conference on Automation Science and Engineering (CASE)*, 2024, S. 1594–1599
- [20] FLETCHER, N. H. ; ROSSING, T. D.: *The Physics of Musical Instruments*. Springer Link, 1998
- [21] FOSTER, K. R. ; SCHWAN, H. P.: Dielectric properties of tissues and biological materials: a critical review. In: *Crit Rev Biomed Eng* 17 (1989), Nr. 1, S. 25–104. – ISSN 0278–940X (Print); 0278–940X (Linking)
- [22] GAL, Y. ; GHAHRAMANI, Z. : *Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning*. 2016
- [23] GAVAZZI, S. : *Technical developments for MR-based electrical property mapping*. Utrecht, The Netherlands, Utrecht University, Diss., 2020
- [24] GELVICH, E. ; MAZOKHIN, V. : Contact flexible microstrip applicators (CFMA) in a range from microwaves up to short waves. In: *IEEE Transactions on Biomedical Engineering* 49 (2002), Nr. 9, S. 1015–1023. <http://dx.doi.org/10.1109/TBME.2002.802053>. – DOI 10.1109/TBME.2002.802053
- [25] GLOROT, X. ; BENGIO, Y. : Understanding the difficulty of training deep feedforward neural networks. In: TEH, Y. W. (Hrsg.) ; TITTERINGTON, M. (Hrsg.): *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics* Bd. 9. PMLR (Proceedings of Machine Learning Research), 249–256
- [26] GLOROT, X. ; BENGIO, Y. : Understanding the Difficulty of Training Deep Feed-forward Neural Networks. In: *Proceedings of the 13th International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2010
- [27] GRABSKI, J. K. ; LESNIC, D. ; JOHANSSON, B. T.: Identification of a time-dependent bio-heat blood perfusion coefficient. In: *International Communications in Heat and Mass Transfer* 75 (2016), 218–222. <https://doi.org/10.1016/j.icheatmasstransfer.2015.12.028>. – DOI <https://doi.org/10.1016/j.icheatmasstransfer.2015.12.028>. – ISSN 0735–1933
- [28] GROEN, J. A. ; CREZEE, J. ; LAARHOVEN, H. W. M. ; BIJLSMA, M. F. ; KOK, H. P.: Quantification of tissue property and perfusion uncertainties in hyperthermia treatment planning: Multianalysis using polynomial chaos expansion. In: *Comput Methods Programs Biomed* 240 (2023), Oct, S. 107675. <http://dx.doi.org/10.1016/j.cmpb.2023.107675>. – DOI 10.1016/j.cmpb.2023.107675. – ISSN 1872–7565 (Electronic); 0169–2607 (Linking)
- [29] GUIOT, C. ; MADON, E. ; ALLEGRO, D. ; PIANTÀ, P. G. ; BAIOTTO, B. ; GABRIELE, P. : Perfusion and thermal field during hyperthermia. Experimental measurements and modelling in recurrent breast cancer. In: *Physics in Medicine & Biology* 43 (1998), oct, Nr. 10, 2831. <http://dx.doi.org/10.1088/0031-9155/43/10/012>. – DOI 10.1088/0031–9155/43/10/012

- [30] HASSANI, V. ; PEDRO AGUIAR, A. ; PASCOAL, A. M. ; ATHANS, M. : Further results on plant parameter identification using continuous-time multiple-model adaptive estimators. In: *Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*, 2009, S. 7261–7266
- [31] HE, K. ; ZHANG, X. ; REN, S. ; SUN, J. : Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. In: *CoRR* abs/1502.01852 (2015). <http://arxiv.org/abs/1502.01852>
- [32] ISMAILOV, M. I. ; BAZÁN, F. S. ; BEDIN, L. : Time-dependent perfusion coefficient estimation in a bioheat transfer problem. In: *Computer Physics Communications* 230 (2018), 50–58. <http://dx.doi.org/https://doi.org/10.1016/j.cpc.2018.04.019>. – DOI <https://doi.org/10.1016/j.cpc.2018.04.019>. – ISSN 0010–4655
- [33] ISSELS, R. D. ; LINDNER, L. H. ; VERWEIJ, J. ; WUST, P. ; REICHARDT, P. ; SCHEM, B.-C. ; ABDEL-RAHMAN, S. ; DAUGAARD, S. ; SALAT, C. ; WENDTNER, C.-M. ; VUJASKOVIC, Z. ; WESSALOWSKI, R. ; JAUCH, K.-W. ; DÜRR, H. R. ; PLONER, F. ; BAUR-MELNYK, A. ; MANSMANN, U. ; HIDDEMANN, W. ; BLAY, J.-Y. ; HOHENBERGER, P. : Neo-adjuvant chemotherapy alone or with regional hyperthermia for localised high-risk soft-tissue sarcoma: a randomised phase 3 multicentre study. In: *Lancet Oncol* 11 (2010), Jun, Nr. 6, S. 561–570. [http://dx.doi.org/10.1016/S1470-2045\(10\)70071-1](http://dx.doi.org/10.1016/S1470-2045(10)70071-1). – DOI 10.1016/S1470-2045(10)70071-1. – ISSN 1474–5488 (Electronic); 1470–2045 (Print); 1470–2045 (Linking)
- [34] KINGMA, D. P. ; BA, J. : *Adam: A Method for Stochastic Optimization*. <https://arxiv.org/abs/1412.6980>, 2017
- [35] KOK, H. P. ; BERG, C. A. T. d. ; BEL, A. ; CREZEE, J. : Fast thermal simulations and temperature optimization for hyperthermia treatment planning, including realistic 3D vessel networks. In: *Medical Physics* 40 (2013), Nr. 10, 103303. <http://dx.doi.org/https://doi.org/10.1118/1.4821544>. – DOI <https://doi.org/10.1118/1.4821544>
- [36] KOK, H. P. ; BERG, C. A. T. V. ; HAAREN, P. M. A. V. ; CREZEE, J. : Artefacts in intracavitary temperature measurements during regional hyperthermia. In: *Physics in Medicine & Biology* 52 (2007), Nr. 17, S. 5157–5172. <http://dx.doi.org/10.1088/0031-9155/52/17/001>. – DOI 10.1088/0031-9155/52/17/001
- [37] KOK, H. P. ; CREZEE, J. : Validation and practical use of Plan2Heat hyperthermia treatment planning for capacitive heating. In: *International Journal of Hyperthermia* 39 (2022), Nr. 1, 952–966. <http://dx.doi.org/10.1080/02656736.2022.2093996>. – DOI 10.1080/02656736.2022.2093996. – PMID: 35853733
- [38] KOK, H. P. ; CRESSMAN, E. N. K. ; CEELEN, W. ; BRACE, C. L. ; IVKOV, R. ; GRÜLL, H. ; HAAR, G. ter ; WUST, P. ; CREZEE, J. : Heating technology

- for malignant tumors: A review. In: *International Journal of Hyperthermia* 37 (2020), Nr. 1, S. 711–741
- [39] KRSTIC, M. ; SMYSHLYAEV, A. : *Boundary control of PDEs: A course on backstepping designs.* SIAM, 2008
- [40] LAGARIS, I. ; LIKAS, A. ; FOTIADIS, D. : Artificial neural networks for solving ordinary and partial differential equations. In: *IEEE Transactions on Neural Networks* 9 (1998), Nr. 5, S. 987–1000
- [41] Kapitel 24. In: LAGENDIJK, J. J. W.: *Heat Transfer in Tissues.* Springer Netherlands. – ISBN 978-94-009-3597-6, 517–552
- [42] LAGENDIJK, J. J. W. ; VAN RHOON, G. C. ; HORNSLETH, S. N. ; WUST, P. ; DE LEEUW, A. C. C. ; SCHNEIDER, C. J. ; VAN DDK, J. D. P. ; VAN DER ZEE, J. ; VAN HEEK-ROMANOWSKI, R. ; RAHMAN, S. A. ; GROMOLL, C.: Esho Quality Assurance Guidelines for Regional Hyperthermia. In: *International Journal of Hyperthermia* 14 (1998), Nr. 2, 125–133. <http://dx.doi.org/10.3109/02656739809018219>. – DOI 10.3109/02656739809018219. – PMID: 9589319
- [43] LAMAITRE, G. ; VAN DIJK, J. D. ; GELVICH, E. A. ; WIERSMA, J. ; SCHNEIDER, C. J.: SAR characteristics of three types of Contact Flexible Microstrip Applicators for superficial hyperthermia. In: *Int J Hyperthermia* 12 (1996), Mar-Apr, Nr. 2, S. 255–269. <http://dx.doi.org/10.3109/02656739609022513>. – DOI 10.3109/02656739609022513. – ISSN 0265-6736 (Print); 0265-6736 (Linking)
- [44] LANG, J. ; ERDMANN, B. ; SEEBASS, M. : Impact of nonlinear heat transfer on temperature control in regional hyperthermia. In: *IEEE Transactions on Biomedical Engineering* 46 (1999), Nr. 9, S. 1129–1138. <http://dx.doi.org/10.1109/10.784145>. – DOI 10.1109/10.784145
- [45] LECUN, Y. ; BOTTOU, L. ; BENGIO, Y. ; HAFFNER, P. : Gradient-based learning applied to document recognition. In: *Proceedings of the IEEE* 86 (1998), Nr. 11, S. 2278–2324. <http://dx.doi.org/10.1109/5.726791>. – DOI 10.1109/5.726791
- [46] LIU, D. C. ; NOCEDAL, J. : On the limited memory BFGS method for large scale optimization. In: *Mathematical Programming* 45 (1989), 503-528. <https://api.semanticscholar.org/CorpusID:5681609>
- [47] LIU, Z. ; YANG, Y. ; CAI, Q.-D. : *Solving Differential Equation with Constrained Multilayer Feedforward Network.* 2019
- [48] LIU, Z. ; WANG, Y. ; VAIDYA, S. ; RUEHLE, F. ; HALVERSON, J. ; SOLJAČIĆ, M. ; HOU, T. Y. ; TEGMARK, M. : *KAN: Kolmogorov-Arnold Networks.* 2024
- [49] LONGO, T. A. ; GOPALAKRISHNA, A. ; TSIVIAN, M. ; VAN NOORD, M. ; RASCH, C. R. ; INMAN, B. A. ; GEIJSEN, E. D.: A systematic review of regional hyperthermia therapy in bladder cancer. In: *Int J Hyperthermia* 32 (2016), Jun,

- Nr. 4, S. 381–389. <http://dx.doi.org/10.3109/02656736.2016.1157903>. – DOI 10.3109/02656736.2016.1157903. – ISSN 1464–5157 (Electronic); 0265–6736 (Print); 0265–6736 (Linking)
- [50] LU, L. ; MENG, X. ; MAO, Z. ; KARNIADAKIS, G. E.: DeepXDE: A deep learning library for solving differential equations. In: *SIAM Review* 63 (2021), Nr. 1, S. 208–228. <http://dx.doi.org/10.1137/19M1274067>. – DOI 10.1137/19M1274067
- [51] MARKIDIS, S. : *The Old and the New: Can Physics-Informed Deep-Learning Replace Traditional Linear Solvers?* <https://arxiv.org/abs/2103.09655>. Version: 2021
- [52] MED-LOGIX SRL: *Brochure ALBA ON4000D*. 2020
- [53] MISHRA, S. ; MOLINARO, R. : Estimates on the generalization error of physics-informed neural networks for approximating PDEs. In: *IMA Journal of Numerical Analysis* 43 (2022), 01, Nr. 1, 1-43. <http://dx.doi.org/10.1093/imanum/drab093>. – DOI 10.1093/imanum/drab093. – ISSN 0272–4979
- [54] MISHRA, S. ; MOLINARO, R. : *Estimates on the generalization error of Physics Informed Neural Networks (PINNs) for approximating a class of inverse problems for PDEs*. <https://arxiv.org/abs/2007.01138>. Version: 2023
- [55] NIELSEN, O. S. ; HORSMAN, M. ; OVERGAARD, J. : A future for hyperthermia in cancer treatment? In: *European Journal of Cancer* 37 (2001), 2025/02/19, Nr. 13, 1587–1589. [http://dx.doi.org/10.1016/S0959-8049\(01\)00193-9](http://dx.doi.org/10.1016/S0959-8049(01)00193-9). – DOI 10.1016/S0959-8049(01)00193-9. ISBN 0959–8049
- [56] OKAJIMA, J. ; MARUYAMA, S. ; TAKEDA, H. ; KOMIYA, A. : Dimensionless solutions and general characteristics of bioheat transfer during thermal therapy. In: *Journal of Thermal Biology* 34 (2009), Nr. 8, S. 377–384. <http://dx.doi.org/https://doi.org/10.1016/j.jtherbio.2009.08.001>. – DOI <https://doi.org/10.1016/j.jtherbio.2009.08.001>. – ISSN 0306–4565
- [57] OVERGAARD, J. : History and Heritage—An Introduction. In: *Hyperthermic Oncology 1984* Bd. 2. Taylor & Francis, 1985, S. 3–8
- [58] OVERGAARD, J. : The current and potential role of hyperthermia in radiotherapy. In: *Int J Radiat Oncol Biol Phys* 16 (1989), Mar, Nr. 3, S. 535–549. [http://dx.doi.org/10.1016/0360-3016\(89\)90470-7](http://dx.doi.org/10.1016/0360-3016(89)90470-7). – DOI 10.1016/0360–3016(89)90470–7. – ISSN 0360–3016 (Print); 0360–3016 (Linking)
- [59] OVERGAARD, J. ; GONZALEZ GONZALEZ, D. ; HULSHOF, M. C. ; ARCANGELI, G. ; DAHL, O. ; MELLA, O. ; BENTZEN, S. M.: Randomised trial of hyperthermia as adjuvant to radiotherapy for recurrent or metastatic malignant melanoma. European Society for Hyperthermic Oncology. In: *Lancet* 345 (1995), Mar, Nr. 8949, S. 540–543. [http://dx.doi.org/10.1016/s0140-6736\(95\)90463-8](http://dx.doi.org/10.1016/s0140-6736(95)90463-8). – DOI 10.1016/s0140–6736(95)90463–8. – ISSN 0140–6736 (Print); 0140–6736 (Linking)

- [60] PASZKE, A. ; GROSS, S. ; CHINTALA, S. ; CHANAN, G. ; YANG, E. ; DEVITO, Z. ; LIN, Z. ; DESMAISON, A. ; ANTIGA, L. ; LERER, A. : Automatic Differentiation in PyTorch. In: *NIPS 2017 Workshop on Autodiff*, 2017
- [61] PENNES, H. H.: Analysis of Tissue and Arterial Blood Temperatures in the Resting Human Forearm. In: *Journal of Applied Physiology* 1 (1948), Nr. 2, S. 93–122
- [62] POLYAK, B. T. ; JUDITSKY, A. B.: Acceleration of Stochastic Approximation by Averaging. In: *SIAM Journal on Control and Optimization* 30 (1992), Nr. 4, 838-855. <http://dx.doi.org/10.1137/0330046>. – DOI 10.1137/0330046
- [63] RAISSI, M. ; PERDIKARIS, P. ; KARNIADAKIS, G. : Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. In: *Journal of Computational Physics* 378 (2019), 686-707. <http://dx.doi.org/https://doi.org/10.1016/j.jcp.2018.10.045>. – DOI <https://doi.org/10.1016/j.jcp.2018.10.045>. – ISSN 0021-9991
- [64] RUMELHART, D. E. ; HINTON, G. E. ; WILLIAMS, R. J.: Learning representations by back-propagating errors. In: *Nature* 323 (1986), Nr. 6088, 533–536. <http://dx.doi.org/10.1038/323533a0>. – DOI 10.1038/323533a0. ISBN 1476-4687
- [65] SCARDAPANE, S. : *Alice's Adventures in a Differentiable Wonderland – Volume I, A Tour of the Land*. <https://arxiv.org/abs/2404.17625>. Version: 2024
- [66] SMYSHLYAEV, A. ; KRSTIC, M. : Backstepping observers for a class of parabolic PDEs. In: *Systems & Control Letters* 54 (2005), Nr. 7, S. 613–625
- [67] STAUFFER, P. R. ; MACCARINI, P. ; ARUNACHALAM, K. ; CRACIUNESCU, O. ; DIEDERICH, C. ; JUANG, T. ; ROSSETTO, F. ; SCHLORFF, J. ; MILLIGAN, A. ; HSU, J. ; SNEED, P. ; VUJASKOVIC, Z. : Conformal microwave array (CMA) applicators for hyperthermia of diffuse chest wall recurrence. In: *Int J Hyperthermia* 26 (2010), Nr. 7, S. 686–698. <http://dx.doi.org/10.3109/02656736.2010.501511>. – DOI 10.3109/02656736.2010.501511. – ISSN 1464-5157 (Electronic); 0265-6736 (Print); 0265-6736 (Linking)
- [68] TANCIK, M. ; SRINIVASAN, P. P. ; MILDENHALL, B. ; FRIDOVICH-KEIL, S. ; RAGHAVAN, N. ; SINGHAL, U. ; RAMAMOORTHI, R. ; BARRON, J. T. ; NG, R. : Fourier Features Let Networks Learn High Frequency Functions in Low Dimensional Domains. In: *CoRR* abs/2006.10739 (2020). <https://arxiv.org/abs/2006.10739>
- [69] TREFNÁ, H. D. ; CREZEE, H. ; SCHMIDT, M. ; MARDER, D. ; LAMPRECHT, U. ; EHmann, M. ; HARTMANN, J. ; NADOBNY, J. ; GELLERMANN, J. ; VAN HOLTHE, N. ; GHADJAR, P. ; LOMAX, N. ; ABDEL-RAHMAN, S. ; BERT, C. ; BAKKER, A. ; HURWITZ, M. D. ; DIEDERICH, C. J. ; STAUFFER, P. R. ; VAN RHOON, G. C.: Quality assurance guidelines for superficial hyperthermia clinical trials: I. Clinical requirements. In: *International Journal of Hyperthermia* 33 (2017), Nr. 4, S. 471–482. <http://dx.doi.org/10.1080/02656736.2016.1277791>. – DOI 10.1080/02656736.2016.1277791

- [70] VAN DER GAAG, M. L. ; DE BRUIJNE, M. ; SAMARAS, T. ; VAN DER ZEE, J. ; VAN RHOON, G. C.: Development of a guideline for the water bolus temperature in superficial hyperthermia. In: *International Journal of Hyperthermia* 22 (2006), Nr. 8, S. 637–656
- [71] VERNON, C. C. ; HAND, J. W. ; FIELD, S. B. ; MACHIN, D. ; WHALEY, J. B. ; ZEE, J. van d. ; PUTTEN, W. L. ; RHOON, G. C. ; DIJK, J. D. ; GONZÁLEZ GONZÁLEZ, D. ; LIU, F. F. ; GOODMAN, P. ; SHERAR, M. : Radiotherapy with or without hyperthermia in the treatment of superficial localized breast cancer: results from five randomized controlled trials. International Collaborative Hyperthermia Group. In: *Int J Radiat Oncol Biol Phys* 35 (1996), Jul, Nr. 4, S. 731–744. [http://dx.doi.org/10.1016/0360-3016\(96\)00154-x](http://dx.doi.org/10.1016/0360-3016(96)00154-x). – DOI 10.1016/0360-3016(96)00154-x. – ISSN 0360–3016 (Print); 0360–3016 (Linking)
- [72] WANG, S. ; SANKARAN, S. ; WANG, H. ; PERDIKARIS, P. : *An Expert's Guide to Training Physics-informed Neural Networks*. 2023
- [73] WANG, S. ; YU, X. ; PERDIKARIS, P. : When and why PINNs fail to train: A Neural Tangent Kernel perspective. In: *Journal of Computational Physics* (2021), 10, S. 110768. <http://dx.doi.org/10.1016/j.jcp.2021.110768>. – DOI 10.1016/j.jcp.2021.110768
- [74] WEINBAUM, S. ; JIJI, L. M.: A new simplified bioheat equation for the effect of blood flow on local average tissue temperature. In: *J Biomech Eng* 107 (1985), May, Nr. 2, S. 131–139. <http://dx.doi.org/10.1115/1.3138533>. – DOI 10.1115/1.3138533. – ISSN 0148–0731 (Print); 0148–0731 (Linking)
- [75] YADAN, O. : *Hydra - A framework for elegantly configuring complex applications*. Github. <https://github.com/facebookresearch/hydra>. Version: 2019
- [76] ZEE, J. van d. ; GONZÁLEZ, D. ; RHOON, G. C. ; DIJK, J. D. ; PUTTEN, W. L. ; HART, A. A.: Comparison of radiotherapy alone with radiotherapy plus hyperthermia in locally advanced pelvic tumours: a prospective, randomised, multicentre trial. In: *The Lancet* 355 (2000), 2025/02/20, Nr. 9210, 1119–1125. [http://dx.doi.org/10.1016/S0140-6736\(00\)02059-6](http://dx.doi.org/10.1016/S0140-6736(00)02059-6). – DOI 10.1016/S0140-6736(00)02059-6. ISBN 0140–6736
- [77] ZHANG, A. ; LIPTON, Z. C. ; LI, M. ; SMOLA, A. J.: Dive into Deep Learning. In: *arXiv preprint arXiv:2106.11342* (2021)
- [78] ZHANG, J. ; SANDISON, G. A. ; MURTHY, J. Y. ; XU, L. X.: Numerical simulation for heat transfer in prostate cancer cryosurgery. In: *Journal of Biomechanical Engineering* 127 (2005), Nr. 2, S. 279–294