National Technical University of Athens
School of Electrical and Computer Engineering
MSc Data Science and Machine Learning

Application of Physics Informed Neural Networks in the development of Physiologically Based Kinetic Models

Diploma Thesis

of

VASILEIOS MINADAKIS

Supervisor 1: Konstantina Nikita
Professor, School of Electrical and Computer Engineering, NTUA

Supervisor 2: Haralambos Sarimveis
Professor, School of Chemical Engineering, NTUA

Athens, October 2023

National Technical University of Athens
School of Electrical and Computer Engineering
MSc Data Science and Machine Learning

DISCLAIMER ON ACADEMIC ETHICS AND INTELLECTUAL PROPERTY RIGHTS

Being fully aware of the implications of copyright laws, I expressly state that this diploma thesis, as well as the electronic files and source codes developed or modified in the course of this thesis, are solely the product of my personal work and do not infringe any rights of intellectual property, personality and personal data of third parties, do not contain work / contributions of third parties for which the permission of the authors / beneficiaries is required and are not a product of partial or complete plagiarism, while the sources used are limited to the bibliographic references only and meet the rules of scientific citing. The points where I have used ideas, text, files and / or sources of other authors are clearly mentioned in the text with the appropriate citation and the relevant complete reference is included in the bibliographic references section. I fully, individually and personally undertake all legal and administrative consequences that may arise in the event that it is proven, in the course of time, that this thesis or part of it does not belong to me because it is a product of plagiarism.

(Signature)

. . . . . .
Vasileios Minadakis

Abstract

Physiologically Based Kinetic models are mathematical models of differential equations and are used to predict the processes of administration, distribution, metabolism and excretion (ADME) of pharmaceutical or toxic substances to which organisms are exposed. PBK models incorporate details from the physiology of the examined organisms, such as the mass of the organs and tissues, the blood flow rates of the organs etc. Therefore, PBK models are often used along with biodistribution experimental data to estimate the value of kinetic parameters that are difficult, or even infeasible to estimate experimentally. Consequently, the estimation of these parameters using experimental data is an optimization problem. This optimization process may present various obstacles, among which the most common is overparameterization of the model. The redundant parameters usually provoke identifiability issues to the model, meaning that the values of the estimated parameters have no physical significance.

There are various approaches on how to estimate the parameters of a PBK model. A recently proposed approach in optimization problems of dynamic systems involves the use of Physics Informed Neural Networks (PINNs). This approach exploits the ability of the artificial neural networks to perform very well as function approximators and combines that with information directly derived from the differential equations that describe the examined dynamic system. PINNs have been applied in two types of problems. In forward problems, they are trained to predict the solution of the differential equations, and they have proven to be very efficient in cases where the equations are difficult to solve using numerical solvers. The second type of problems is the inverse problems where PINNs are employed to estimate the unknown parameters of dynamic systems, using available data.

This diploma thesis will focus on the development of a PBK model that predicts the biodistribution of five perfluoroalkyl substances (PFAS) in rainbow trout fish, which consume food rich in PFAS. These chemicals consist a large class of synthetic chemicals that contain carbon-fluorine bonds, which are one of the strongest chemical bonds and that makes PFAS very resistant to degradation. To estimate the values of the model's unknown parameters, two different approaches will be tested. In the first approach, the unknown parameters will be estimated by implementing an optimization workflow, aimed at minimizing the value of an objective function that quantifies the agreement between the model's predictions and experimental data. The second approach will implement a PINN to estimate the unknown parameters of the model. Moreover, identifiability analysis will be performed using a workflow that exploits the profile likelihood method, to improve the robustness of the development process. Finally, a comparison of the two approaches will be provided, highlighting the advantages of each method. The efficiency of PINN workflows

in the development of PBK models will be particularly discussed.

Keywords

PBK, PFAS, identifiability analysis, Physics Informed Neural Networks, PINN, machine learning

Περίληψη

Τα κινητικά μοντέλα που βασίζονται στη φυσιολογία των οργανισμών είναι μαθηματικά μοντέλα διαφορικών εξισώσεων που χρησιμοποιούνται για την πρόβλεψη και περιγραφή της απορρόφησης, βιοκατανομής, μεταβολισμού και απέκκρισης φαρμακευτικών ή τοξικών ουσιών, στις οποίες οι οργανισμοί εκτίθενται. Τα κινητικά μοντέλα φυσιολογίας εμπεριέχουν πληροφορία και λεπτομέρειες από την φυσιολογία των οργανισμών, όπως οι μάζα των οργάνων και των ιστών, οι ρυθμοί ροής αίματος στα όργανα κ.ά. Επομένως, τα κινητικά μοντέλα φυσιολογίας χρησιμοποιούνται συχνά σε συνδυασμό με πειραματικά δεδομένα βιοκατανομής μίας ουσίας στους ιστούς ενός οργανισμού, προκειμένου να εκτιμηθούν κινητικές παράμετροι, των οποίων ο πειραματικός υπολογισμός είναι δύσκολος ή και ανέφικτος. Επομένως, η διαδικασία εκτίμησης αυτών των παραμέτρων ανάγεται σε ένα κλασικό πρόβλημα βελτιστοποίησης. Η διαδικασία βελτιστοποίησης μπορεί να παρουσιάσει πολλαπλά εμπόδια, με το συχνότερο από αυτά να είναι η υπερ-παραμετροποίηση του μοντέλου. Οι πλεονάζουσες παράμετροι συχνά προκαλούν προβλήματα ταυτοποιησιμότητας στο μοντέλο, που σημαίνει ότι οι εκτιμηθείσες παράμετροι δεν έχουν καμία φυσική σημασία.

Για την εκτίμηση των παραμέτρων ενός μοντέλου υπάρχουν διάφορες προσεγγίσεις. Μία μεθοδολογία που έχει προταθεί τα τελευταία χρόνια είναι η χρήση νευρωνικών δικτύων που βασίζονται στη φυσική του συστήματος που μελετάται. Αυτή η μεθοδολογία εκμεταλλεύεται την ικανότητα των τεχνητών νευρωνικών δικτύων να προσεγγίσουν οποιαδήποτε άγνωστη συνάρτηση και τη συνδυάζει με πληροφορία που προέρχεται απευθείας από τις διαφορικές εξισώσεις, που περιγράφουν τη δυναμική του εκάστοτε συστήματος. Η μεθοδολογία αυτή έχει χρησιμοποιηθεί σε δύο ειδών προβλήματα. Η πρώτη κατηγορία αφορά προβλήματα στα οποία τα νευρωνικά δίκτυα εκπαιδεύονται ώστε να μάθουν να προβλέπουν τη λύση των διαφορικών εξισώσεων κι εν τέλει να λειτουργήσουν ως εναλλακτική μέθοδος επίλυσης διαφορικών εξισώσεων. Η δεύτερη κατηγορία προβλημάτων είναι εκείνη στην οποία αυτές οι αρχιτεκτονικές νευρωνικών δικτύων χρησιμοποιούνται για την εκτίμηση των άγνωστων παραμέτρων των διαφορικών εξισώσεων ενός δυναμικού συστήματος, χρησιμοποιώντας διαθέσιμα πειραματικά δεδομένα.

Η παρούσα διπλωματική εργασία πρόκειται να εστιάσει στην ανάπτυξη ενός κινητικού μοντέλου φυσιολογίας που προβλέπει τη βιοκατανομή πέντε διαφορετικών υπερφθοριωμένων αλκυλιωμένων ουσιών στο είδος ψαριού που ονομάζεται ιριδίζουσα πέστροφα, το οποίο έχει εκτεθεί σε αυτές τις ουσίες μέσω της διατροφής. Ως υπερφθοριωμένες αλκυλιωμένες ουσίες θεωρείται μία μεγάλη κατηγορία συνθετικών χημικών, τα οποία περιέχουν δεσμούς άνθρακα-φθορίου, που τα καθιστούν ιδιαίτερα ανθεκτικά σε αποσύνθεση. Για την εκτίμηση των αγνώστων παραμέτρων του μοντέλου θα χρησιμοποιηθούν δύο διαφορετικές προσεγγίσεις. Στην πρώτη προσέγγιση, οι άγνωστες παράμετροι θα εκτιμηθούν υλοποιώντας έναν αλγόριθμο

βελτιστοποίησης, ο οποίος θα ελαχιστοποιεί την τιμή μίας αντικειμενικής συνάρτησης, η οποία θα υπολογίζει το πόσο καλά προβλέπει το μοντέλο τα πειραματικά δεδομένα. Η δεύτερη προσέγγιση θα αφορά την υλοποίηση ενός τεχνητού νευρωνικού δικτύου ενισχυόμενο με πληροφορία από τις διαφορικές εξισώσεις του μοντέλου, με σκοπό την εκτίμηση των αγνώστων παραμέτρων. Επιπλέον, το ζήτημα της ταυτοποιησιμότητας των παραμέτρων θα προσεγγιστεί αξιοποιώντας την μέθοδος υπολογισμού του προφίλ της πιθανοφάνειας των παραμέτρων, ενισχύοντας έτσι την ευρωστία του μοντέλου. Καταλήγοντας, θα γίνει μία σύγκριση μεταξύ των δύο μεθοδολογιών και θα παρουσιαστούν τα προτερήματα της καθεμιάς σε σχέση με την άλλη.

Λέξεις Κλειδιά

κινητικά μοντέλα φυσιολογίας, τεχνητά νευρωνικά δίκτυα, υπερφθοριωμένες αλκυλιωμένες, ανάλυση ταυτοποιησιμότητας, μηχανική μάθηση

Acknowledgements

I would like to take this opportunity to extend my heartfelt appreciation to those who have made significant contributions to the completion of my MSc thesis. First and foremost, I wish to express my profound gratitude to Professor Konstantina Nikita. Our collaboration and communication were flawless throughout the entire duration of this thesis.

I am equally thankful to Professor Haralambos Sarimveis, who served as my co-supervisor, providing valuable insights and expertise related to the field of artificial neural networks. I also extend my gratitude to Periklis Tsiros, a PhD candidate at the School of Chemical Engineering (NTUA), whose guidance and advice were invaluable in overcoming various challenges during the development of this thesis.

I would like to express my appreciation to Assistant Professor Athanasios Voulodimos for his participation on the examination committee, alongside Professors Nikita and Sarimveis. His constructive feedback and suggestions provided guidance for future research in this scientific field.

Furthermore, I am deeply indebted to those who provided unwavering emotional support throughout this journey. I cannot thank Anastasia, my partner, enough for her continuous encouragement and assistance, which played a pivotal role in numerous moments during the thesis's development. I would also like to extend my gratitude to my parents, Giorgos and Eleni, for their support and for always standing by me, supporting my decisions.

Athens, October 2023

Vasileios Minadakis

Table of Contents

List of Figures

List of Images

List of Tables

Chapter 1

Introduction

In recent years, the application of machine learning has blossomed in multiple scientific fields, revolutionizing the way we approach complex problems and extract insights from data. Two major factors have contributed to this progress. The first one is the increased availability and accessibility to data through the internet. The other factor is the progress made in the field of hardware, which facilitated more complex computations. Noteworthy is the role of machine learning in the fields of medical science, chemistry and pharmaceuticals. Especially artificial neural networks (ANNs), which are a branch of machine learning techniques, have emerged as one of the most popular methods in addressing complex computational problems across various domains. Their ability to learn patterns from extensive datasets has driven significant advancements in fields such as computer vision, natural language processing and robotics.

One of the most interesting applications of artificial neural networks in recent years is the development of Physics Informed Neural Networks (PINNs). They represent an innovative synergy between the traditional physics-based modeling and cutting-edge machine learning techniques. These networks leverage the capability of neural networks to approximate any complex function while enforcing the fundamental laws of physics of the examined dynamic system as constraints. PINNs consist of two structural units. The first one is an artificial network, which is usually a feedforward network. The second structural unit consists of the differential equations that describe the physics of the examined dynamic system. PINNs are capable of estimating the values of the derivatives using automatic differentiation and comparing them with those provided by the differential equations. Automatic differentiation is used to estimate the values of the derivatives of the output with respect to the input of the neural network, exploiting the chain rule methodology. Therefore, the learning algorithm aims to minimize the loss function, which is estimated based on the discrepancy between the value of the derivative estimated from the automatic differentiation process and the one estimated from the equations of the dynamic system.

As a result, PINNs can be applied to two different tasks. The first one involves solving the differential equations of a dynamic system. PINNs can be trained similarly to a simple feedforward neural network to act as a function approximator. However, in the case of PINNs, the physical laws of the physical system are incorporated into the network, by satisfying the initial and the boundary conditions of the system. Consequently, the PINN can provide an alternative way to solve differential equations instead of using classic

numerical solvers. The second category of problems that utilize PINNs, is the estimation of the unknown parameters of a dynamic system, using experimental data. PINNs are trained using the experimental data, the initial and the boundary conditions and update the values of the unknown parameters along with the weights and biases of the neural network. Consequently, PINNs have garnered attention from various scientific fields and are used for a wide range of problems.

This diploma thesis focuses on exploring the capabilities of PINNs to estimate the unknown parameters of a Physiologically Based Kinetic (PBK) model, using relative experimental data. PBK models originate from the scientific field of pharmacokinetics and pharmacodynamics. They are compartmental models that predict the administration, distribution, metabolism and excretion (ADME) of pharmaceutical or toxic substances. PBK models consist of first-order differential equations that take into account the mass equilibrium of the examined substances. Moreover, PBK models incorporate important information regarding the physiological properties of the examined organism. The scientific literature provides an extensive amount of data regarding the physiological properties of both humans and animals. Therefore, incorporating physiological data into PBK models becomes straightforward, enabling the inclusion of information such as accurate estimations for the mass or blood flow rates of individual organs. Additionally, many PBK models consider the growth of organisms, particularly in simulations covering extended periods, utilizing relevant data on the organisms' growth rates. Consequently, PBK models can provide comprehensive physiological representation of the organisms when suitable experimental data and a-priori knowledge about the kinetic and physiological parameters are available.

This thesis develops a PBK model designed to predict the biodistribution of five distinct perfluoroalkyl substances (PFAS) across the organs of rainbow trout. Rainbow trout is a frequent subject in experimental studies, and numerous biodistribution studies exist concerning PFAS exposure. The dataset employed for this PBK model relates to a 28-day dietary exposure of rainbow trout to PFAS, followed by an equal duration of a depuration phase. Besides data on PFAS concentrations in various organs, the study also provided information on the fish's total mass throughout the experiment. This information was essential for accurately estimating the fish's mass, thereby updating the organ-specific physiological parameters (like mass and blood flows) at each simulation step. However, the study didn't include any PFAS measurements in the fish's excreta. This omission presents a challenge in PBK model development, as direct estimation of PFAS elimination rates becomes complex, necessitating the sourcing of such kinetic parameters from analogous models.

Another aspect that this diploma thesis focuses on, is the identifiability of the parameters of the PBK model. Identifiability issues frequently arise in PBK models, primarily due to factors, such as model overparameterization and the use of noisy or sparse data for parameter estimation. As a result, the development process needs to employ a tool that can provide feedback on which parameters are non-identifiable. The methodology followed in this thesis to address this problem is the profile likelihood method. Estimating the profile likelihood of each estimated parameter of the model offers multiple advantages. Firstly, it is

an easily applied methodology even in large models with numerous parameters. Moreover, it provides feedback for each parameter, classifying them as structurally non-identifiable, practically non-identifiable or identifiable. When a model is deemed structurally non-identifiable, it suggests inherent issues within the model's framework. Addressing this requires changes to the model's structure, as providing more or superior data won't rectify this intrinsic non-identifiability. On the other hand, practical non-identifiability pertains to the data itself. For instance, data might be too noisy for precise parameter estimation or may necessitate augmentation with supplementary measurements.

Taking all these into consideration, the first part of this diploma thesis provides a detailed explanation of the differential equations of the PBK model, as well as its physiological parameters. For the estimation of the unknown parameters of the model, two different approaches are presented and tested. The first one, the optimization workflow, exploits an optimization algorithm to minimize the value of an objective function that estimates the discrepancy of the model's predictions with the corresponding experimental data. This workflow also employs the identifiability analysis tool described above to detect the non-identifiable parameters of the model. The implementation of this workflow was carried out using the statistical programming language R, which is a common choice for the development of PBK models. Additionally, some parts of the code of this workflow, which were used repeatedly, were organised into a custom R package to streamline the use of these functions. For instance, the library contains various metric functions, some of which are presented and used later in this diploma thesis. Additionally, the implementation of the identifiability analysis tool and the profile likelihood are also included in this package. This approach allows for easy use of the identifiability analysis tool in conjunction with any analyzed dynamic system.

The second approach, the PINN workflow, exploits the capabilities of PINNs in solving inverse problems. It aims to determine the unknown parameters of a dynamic system using limited experimental data. The training process of the PINN, especially the tuning of hyperparameters, is elaborated in detail. The development of the PINN was implemented using Python and a set of modules, specifically the DeepXDE module, which has been specifically developed for training PINNs using libraries like PyTorch or Tensorflow, among others. The final objective of this thesis is to compare the PINN workflow with the conventional optimization workflow that is usually employed in the development of PBK models. The advantages and disadvantages of the two methodologies are discussed, and specific recommendations for when to use each methodology are provided. Finally, some possible directions to extend the research in the application of PINNs in PBK development are proposed, given the promising results this methodology has shown.

Chapter 2

Artificial Neural Networks

2.1 Introduction

Artificial Neural Networks (ANNs) stand as a distinct subset within the expansive realm of machine learning and they are tightly interwined with applications of artificial intelligence. Over recent years, the field of machine learning, with a particular emphasis on neural networks, has undergone a profound evolution. This revolution has been boosted principally by the spread use of the internet, which increased the availability and the sharing of enormous amount of data all around the world. Simultaneously, advances in computer hardware have played a pivotal role in facilitating the computational demands of complex neural network architectures. In this chapter, a short historical progression of neural networks will be presented, tracing their development. This historical context sets the stage for a comprehensive exploration of the theoretical background that enables neural networks to exhibit such remarkable efficacy in diverse applications. The detailed examination of their inner mechanisms will unravel how the artificial neural networks achieve their remarkable capabilities.

2.2 Historical background

The name of the artificial neural networks derives from their structural resemblance to biological neurons and their mode of operation. ANNs consist of multiple nodes, typically referred to as neurons, that imitate the biological neurons. Biological neurons encompass dendrites, soma, axon and synapses or (axon terminals), each with a distinct function. The dendrites of the neurons are responsible for receiving signals from other neurons in their close environment, while the soma processes the signal received from the dendrites. Subsequently, the axon transmits the processed signal from the soma to the opposite end of the neuron, called synapse. Finally, the synapse is connected to the dendrites of other neurons to transmit the signal to them. McCullouch and Pitts in 1944 [6] influenced by the biological neuron system, asserted that a net with thresholds and weights could perform any function achievable by a digital computer.

However, the first trainable neural network architecture was introduced by Frank Rosenblatt in 1957 [7], the Perceptron. It represents the simplest architecture of artificial networks, which consists of a single layer of neurons and is suitable for binary classification

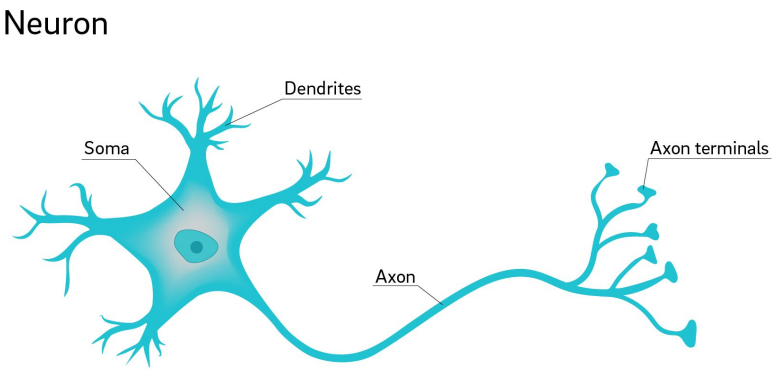

Image 2.1: *Illustration of a biological neuron [2]. The main parts of the neuron are the dendrites, the soma, the axon and the synapses. The synergy of them lead facilitates receiving signal from the environment, processing them and finally transmitting the output of the transformation process. The artificial neural networks imitate the process of the biological neurons.*

problems, where the two classes are linearly separable. As depicted in image 2.2, the first structural unit of Perceptron is a layer of n input nodes, which is called input layer, which provides Perceptron with the input values. Subsequently, the sum of the input values, each multiplied by its corresponding weight, is computed and forwarded to the activation function. The activation function that was originally employed in Perceptron is the step function, which is well-suited for binary classification problems. The step function returns a value of 0 or 1 if the weighted sum of inputs is lower or higher of 0.5, respectively. Then, the output variable Y, receives either 0 or 1 value and determines the class of the given input. Despite the innovative nature of Perceptron's concept, its limitations were evident in its inability to tackle more complex problems. Nevertheless, the Perceptron's conceptual foundation proved pivotal, serving as the precursor to the evolution of deep neural networks.

2.3 Deep Neural Networks

The artificial neural networks have attained the interest of many scientific fields because of their ability to solve a large variety of problems. The fundamental concept underlying the operation of a neural network is its ability to approximate any function, denoted as $f(x)$. The complexity of these functions exhibit large variability. Some examples of simple functions are those described by basic mathematical operators. On the other hand, approximating certain functions can be a challenging task, because they can not be described precisely and step-by-step. For instance, the task of image recognition falls into this category of functions. Although, recognising objects by their image is a straightforward task for human beings, this process can not be described by a specific function $f(x)$. Hence, the neural networks are able to be trained or learn performing tasks of such operations,

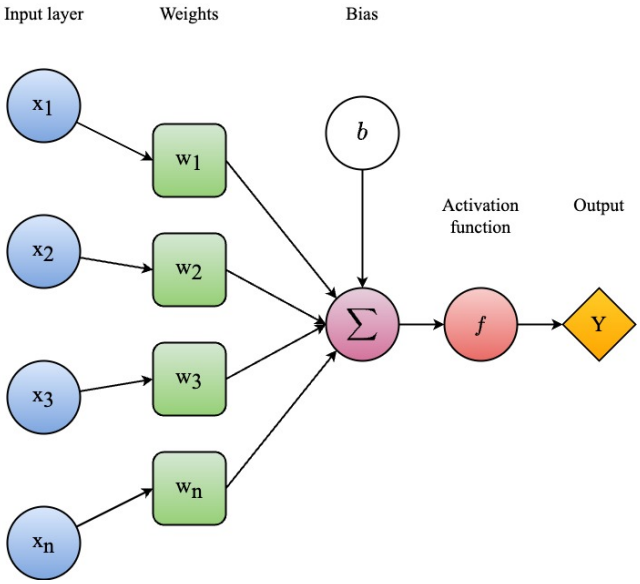

Image 2.2: *Structural representation of Perceptron. It consists of an input layer with n nodes, each one of them is multiplied with the corresponding weight w. The sum of all $w_i x_i$ is provided as input to the activation function, which is the step function. The output of the step function is either 0 or 1.*

that are not easily described. Therefore the core concept of **function approximation** by neural networks, is that any task can be represented by a function $f(x;\theta)$, where θ is a vector of **adjustable parameters**. Therefore, the term of **function approximation** describes the process of defining a function $f(x,\theta)$, as well as tuning the parameters θ in order to perform approximately as the desired function $f(x)$. The process of tuning of the parameters θ is called **learning algorithm** [8].

The ability of artificial neural networks approximating any function is linked to the arrangement of multiple layers of neuron and the interconnections of the neurons of different layers. Consequently, neural networks exhibit various architectures differing on the number of layers, the number of the nodes per layer and the activation function used. The layers existing between the input and the output layer are called hidden layers. The artificial neural networks can be classified into two classes, regarding the number of hidden layers. The shallow neural networks consist of a single (or few) hidden layers, while deep neural networks encompass those with multiple hidden layers.

2.3.1 Feedforward Networks

The most typical architecture of deep neural network is the **multilayer perceptron (MLP)**, which is built by stacking multiple layers of nodes. The inception of MLP was driven by the need to address the limitations of the perceptron, particularly its inability to solve non-linear problems. MLP comprises one or more hidden layers and operates exactly like perceptron. The MLP are commonly referred to as **Feedforward Neural Networks** (**FFN**), because they strictly propagate information from the input layer forward to the hidden layers and there are no feedback connections in which outputs of the model are fed back into itself. MLP are often termed as **Fully-Connected Networks (FCN)** since

every neuron in a layer is connected with all neurons in the subsequent layer [8].

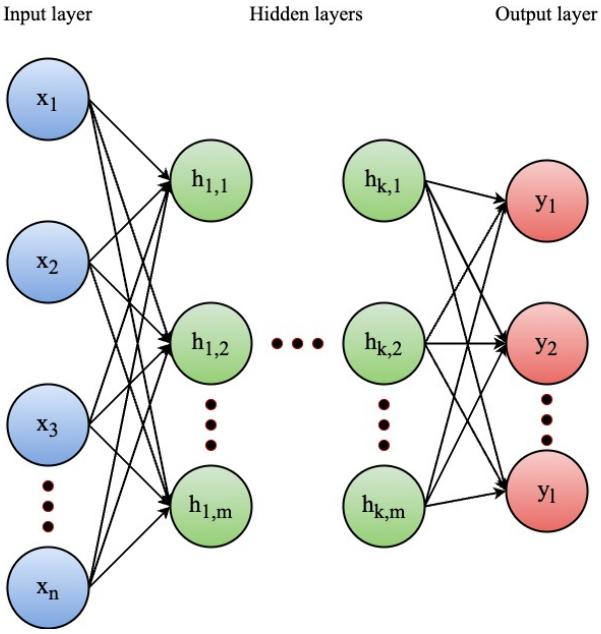

Image 2.3: *Structural representation of a Feedforward Network (FFN). The input layer consists of n nodes. There are k hidden layers that consist of m nodes each. Finally, the output layer consists of l nodes. In feedforward networks, each node of any layer is connected with all nodes of the next layer.*

Feedforward neural networks are called **networks** because their architecture encompass a directed acyclic graph that describes how the functions are composed together, For instance, in case we have three different function$f^{(1)}, f^{(2)}$ and $f^{(3)}$ creating a chain $f(x) = f^{(3)}(f^{(2)}(f^{(1)}(x)))$. This kind of chained functions are commonly represented in neural networks. In this example, $f^{(1)}$ represents the first layer of the network, $f^{(2)}$ the second network, and so on. The total number of chained functions defines the **depth** of the neural network. The final layer of a feedforward network is called **output layer**. The data used for the training of a feedforward network consist of multiple data points $\boldsymbol{x}$, accompanied by a label $y \approx f^*(\boldsymbol{x})$, where $f^*(\boldsymbol{x})$ is considered to be the output of the network. Therefore, during the training of the network, each instance of the training dataset indicates directly to the output layer that it must return a value that is as close as possible to y. However, as the training data do not explicitly indicate the behavior of the rest layers, the learning algorithm must decide how to use these layers to best implement an approximation of f^*. That is the reason why the layers between the input and the output layer are called hidden layers [9].

The other characteristic property of feedforward networks is its **width**. The width is defined by the number of nodes (or neurons) consisted in each hidden layer. Each node of the layers resembles a neuron, as it receives a vectorised input from many other nodes, transforms it into a scalar and finally estimates its own activation value. Nevertheless, the principal target of the feedforward network is not to operate identically as the neural network of the brain. Instead, we should consider the feedforward networks as function approximation machines used to achieve statistical generalization.

Except FFN also exist other architectures of neural networks that perform better at specific tasks. Thus, the second family of neural networks are the **Convolutional Neural Networks (CNN)**. CNNs are particularly suited for tasks involving image, speech, audio recognition where they outperform FFNs. Another family are the **Recurrent Neural Networks (RNN)** and they are linked with the processing of sequential data. RNNs are networks specialized for processing a sequence of values $x^{(1)}, ..., x^{(\tau)}$ because of the concept of parameter sharing across employs only FFN architectures; therefore, the theoretical exploration of CNNs and RNNs will not be further expanded upon [9].

2.3.2 Gradient Descent

The training process of the feedforward networks consists of the forward propagation the backward pass and a learning algorithm that updates the parameters of the network. The learning algorithm used to estimate the values of the parameters of a feedforward network is actually an optimization algorithm. One of the most commonly used optimization algorithms is the **gradient descent**. Suppose we have a function $y = f(x)$ and the derivative is denoted as $f'(x)$ or $\frac{dy}{dx}$ and it is equal to the slope of $f(x)$ at the point x. In other words, it indicates how to scale a small change in the input to obtain the corresponding change in the output $f(x + \epsilon) \approx f(x) + \epsilon f'(x)$. The derivative here is useful because it indicates how to change the value of x to achieve small improvements in y and minimize it. Therefore, the process of reducing the value of $f(x)$ by moving the value of x in small steps with opposite sign of the derivative is called gradient descent. Consequently, the **gradient descent algorithm** suggests that the new value of $\boldsymbol{x}$ is estimated as

$$\boldsymbol{x}' = \boldsymbol{x} - \epsilon \nabla_{\boldsymbol{x}} f(\boldsymbol{x}) \tag{2.1}$$

where ϵ is the learning rate. The algorithm terminates when every element of the gradient is zero or lower than a user-defined threshold. However,the main limitation of the gradient descent is that the algorithm is susceptible into detecting local minima instead of global minimum. Consequently, the gradient descent is very effective algorithm in convex problems, but the majority of the optimization problems are non-convex.

An important limitation of the gradient descent algorithm is the high computational cost it has, when large training datasets are used. Generally, the cost functions used by a feedforward network are actually sums of the differences between the networks predictions and the output variables of the dataset. Therefore, using large training datasets explodes the computational cost to perform one iteration of the gradient descent algorithm. When implementing the gradient descent algorithm in the training of a neural network, we want to minimize the value of loss function

$$J(\boldsymbol{\theta}) = \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}(\mathbf{x}^{(i)}, y^{(i)}, \boldsymbol{\theta}) \tag{2.2}$$

where $\boldsymbol{\theta}$ are the weights and the biases of the network. So the aim is to minimize find the

optimal $\boldsymbol{\theta}$ that minimize the gradient.

$$\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) = \frac{1}{n} \nabla_{\boldsymbol{\theta}} \sum_{i=1}^{n} \mathcal{L}(\mathbf{x}^{(i)}, y^{(i)}, \boldsymbol{\theta}) \tag{2.3}$$

2.3.3 Stochastic Gradient Descent

Stochastic gradient descent (SGD) is an extension of the gradient descent algorithm and is a very commonly used optimization algorithm in feedforward networks. In SGD the gradient is not calculated based on all data points. Instead, the gradient is considered as an expectation and this is calculated based only on one data point. In other words, instead of estimating the gradient $\sum_{i=1}^{n} \nabla_{\theta} \mathcal{L}_i$, we estimate only $\nabla_{\theta} \mathcal{L}_i$ of one data point. Consequently the computational cost is significantly decreased.

Another alternative version of the SGD is the **mini-batch gradient descent**. In each iteration of this algorithm, a small **minibatch** of examples $\mathbb{B} = \boldsymbol{x}^{(1)}, ..., \boldsymbol{x}^{(m')}$ is sampled uniformly from the training set. Therefore, the estimation of the gradient is formed as

$$\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) = \frac{1}{m'} \nabla_{\boldsymbol{\theta}} \sum_{i=1}^{m'} \mathcal{L}(\mathbf{x}^{(i)}, y^{(i)}, \boldsymbol{\theta}). \tag{2.4}$$

2.3.4 Back-Propagation

The training of the neural network is divided into two separate stages. The first one is the **forward propagation**. During the forward propagation, the input $\boldsymbol{x}$ provides the initial information and then propagates up to the hidden units at each hidden layer and finally returns $\hat{\boldsymbol{y}}$. Therefore, the input is transferred through the hidden layers to the output layer and it is transformed into a scalar cost $J(\boldsymbol{\theta})$.

Then the **back-propagation** algorithm [10] is used to pass backward the information from the cost function, in order to compute the gradient of the cost function with respect to the parameters of the network, which is required from the learning algorithm. The back-propagation algorithm is a simple procedure to estimate the gradients, employing the concept of the chain rule to estimate the derivatives iteratively from the output layer towards the input layer. This way, the computational cost remains at low levels, compared to numerical evaluation of the gradients. The back-propagation algorithm is generally used to estimate the derivatives of any given function and it is not explicitly used in the training process of feedforward networks [9].

2.3.5 Activation Functions

The selection of the activation function used by the nodes is important for the operation of a feedforward network, as they define if a node will be activated or not. Although there are multiple functions that have been used as activation functions in the hidden units of feedforward networks, there is nit a straightforward method to select the optimal function

for each problem. The list of functions presented here is not exhaustive and contains only the functions that were examined during the production of the results of this thesis.

Step function

The step function was previously referred as the activation function used in perceptron. Although this function has historical significance due to perceptron, is never used in deep neural networks. The step function is given by

$$\sigma(z) = \begin{cases} 1, & z \geq 0 \\ 0, & z < 0. \end{cases} \tag{2.5}$$

The input provided to step function is compared to a threshold value (usually it is 0). If the input is greater than threshold then the node is activated and passes the information to the following nodes, otherwise it is deactivated. The most important disadvantage of this function is that its derivative is zero, so it cannot be employed along with backpropagation.

Sigmoid

The next is the Sigmoid (or Logistic) function given by

$$\sigma(z) = \frac{1}{1+e^{-z}} = \frac{1}{2} + \frac{1}{2}\tanh\left(\frac{z}{2}\right). \tag{2.6}$$

This function takes as input and tranforms it into a value in the range of 0 to 1. Consequently, the sigmoid function usually selected when the output of the model must predict probabilities. The advantages of sigmoid function are that it is continuous and preserves information around the region of $z = 0$. However, while $z \to +\infty$ then $\sigma(z) = 1$ and when $z \to -\infty$ then $\sigma(z) = 0$, so it becomes more like step activation function. Therefore, the value of the gradients are very small. Very small values of gradients is an obstacle during the training of the networks and this problem is commonly known as vanishing gradient.

Hyperbolic Tangent

Hyperbolic tanget

$$\sigma(z) = \tanh(z) = \frac{e^{z} - e^{-z}}{e^{z} + e^{-z}} = \frac{e^{2z} - 1}{e^{2z} + 1} \tag{2.7}$$

is a very popular choice as activation function in deep neural networks. The tanh is actually a shifted sigmoid function and has the important property of $\sigma(0) = 0$.

Hyperbolic tangent faces the same problem with sigmoid function, the vanishing gradient problem. Functions that suppress a large input space into a smaller space, such as sigmoid does between 0 and 1 or tanh between -1 and 1 are characterized by this problem. Consequently, whenever the neuron takes a value close to the edge values of the output

space, a large change in the input causes a small change on the output because the derivative has a small value. So the this effects the training of the model and the update of the weights. This happens because during backpropagation multiple gradients with very small values are multiplied, which causes very slow update of the weights close to the input layer. The vanishing gradient effect is more common to neural networks with larger number of hidden layers.

ReLU

The Rectified Linear Unit (ReLU) is a widely used activation function, especially in CNNs, and the main advantage is the low coputational cost. Although it gives the impression that ReLU is a linear function, it is not true. Moreover, ReLU is a good choice to face the problem of vanishing gradients (a problem linked with sigmoid and tanh). ReLU ranges from 0 to infinite and is estimated as

$$\sigma(z) = max(0, z). \tag{2.8}$$

Both ReLU and its derivative are monotonic functions. However, the main disadvantage o ReLU is that for negative values of z, ReLU turns into zero, which affects the mapping of negative values, in case the majority of neurons are negative. When the most neurons have output zero, then the gradients fail to flow and the weights are not updated anymore and the learning process is practically terminated. This effect is called Dying ReLU and is usually tackled by using variants of ReLU, the Leaky RelU and GeLU.

Chapter 3

Physics - Infromed Neural Networks

3.1 Introduction

In recent years **Physics Informed Neural Networks (PINN)** have drawn attention as an approach to solve a large variety of problems. PINNs were originally introduced by Maziar Raissi in 2017 for forward [11, 12] and inverse [12, 13] problems involving nonlinear differential equations. Compared to data-driven deep learning, PINNs leverage physics laws to train the network, along with data. In fact, PINNs are designed to integrate scientific computing equations, such as ordinary differential equations (ODE) or partial differential equations (PDE), into the training of a deep learning network. The main difficulty that PINNs are destined to tackle is that deep neural networks are not an effective method to approximate the governing equations of a physical, biological or engineering system because in most cases the availability of data is limited, thus the neural networks can not be trained well enough and concerns about robustness of the models are risen.

The core concept behind the PINNs is that they encompass important information about the dynamics of the studied system directly from the differential equations that describe it. Therefore, they can effectively approximate the mapping between the input and the output variables of a system even when the available data are sparse. So, PINNs have been proven to be a good alternative choice to solve nonlinear differential equation (forward problem), which tend to be difficult to solve in terms of computational cost, or to estimate the unknown parameters of a model (inverse problem) as an alternative to classic optimization techniques.

PINNs are actually ANNs, that after the training, they return an approximation of the derivatives of the output with respect to the input of the network, for multiple points in the integration domain (collocation points), as well as the approximated solution of the differential equations. This part of the PINNs is commonly called *surrogate network* or *approximator*. PINNs also use a *residual network* that encodes the differential equations that govern the examined dynamic system. The residual network is used to estimate the derivatives based on the differential equations of the system, given the output of the deep learning network at each collocation point. Finally, the training of PINNs focuses on minimizing the loss function of the estimated derivatives from the neural network and the corresponding values estimated from the equations.

In this chapter a detailed theoretical overview of the PINNs will be provided. The focus

will be on how the PINNs are used for solving inverse problems. This will clarify how this methodology was used in later stages to estimate the parameters of the physiologically-based kinetics model developed in this diploma thesis.

3.2 Problem setup

First, consider a set of nonlinear differential equations of the general form

$$u_t + \mathcal{N}[u;\lambda] = 0,\ x \in \Omega,\ t \in [0,T], \tag{3.1}$$

where $u(t,x)$ denotes the latent (hidden) solution, $N[\cdot;\lambda]$ is a nonlinear operator with parameters λ and Ω is a subset of $\mathbb{R}^D$. Equation 3.1 represents a large variety of systems described by a set of differential equations, as well as chemical reaction systems, fluid dynamics problems and kinetic equations. Next we define the left hand side of of equation 3.1 as $f(t,x)$

$$f := u_t + \mathcal{N}[u], \tag{3.2}$$

where $u(t,x)$ are the values of the state variables of the system and are approximated by a deep neural network. The estimation of $u(t,x)$ by a deep neural network along with equation 3.2 formulate a physics informed neural network $f(t,x)$. In order to make this structure work, it is necessary to use automatic differentiation and estimate the value of the differential equations given the values of the output variables of the neural network with respect to its input. The automatic differentiation leverages the chain rule method, applied to the operations defined on the network nodes, to estimate the derivatives from the surrogate network.

Therefore, the parameters λ of the PINN contain all the parameters (weights) of the deep neural network that predicts the $u(t,x)$ along with the unknown parameters of the differential equations. The parameters λ can be estimated by minimizing the mean squared error loss

$$MSE = MSE_u + MSE_f \tag{3.3}$$

where

$$MSE_u = \frac{1}{N_u}\sum_{i=1}^{Nu} |u(t_u^i, x_u^i) - u^i|^2, \tag{3.4}$$

and

$$MSE_f = \frac{1}{N_f}\sum_{i=1}^{N_f} |f(t_f^i, x_f^i)|^2. \tag{3.5}$$

In equation 3.4, $\{t_u^i, x_u^i, u^i\}_{i=1}^{N_u}$ denote the initial and boundary conditions as well as the experimental data that may be available. In equation 3.5, the set of $\{t_f^i, x_f^i\}_{i=1}^{N_f}$ denote the collocation points for $f(t, x)$. Therefore, the minimization of MSE_u enforces the model to fit on the experimental data and the boundary conditions, while the minimization of the MSE_f enforces the model to follow the dynamics imposed by the differential equations of the system at a finite set of collocation points. The minimization of the total loss MSE leads to the estimation of both weights and biases of the deep neural network, as well the unknown parameters of the differential equations. However, for specific problems, it is possible to use weights to estimate the total loss MSE, thereby training the PINN more efficiently. So, equation 3.3 is transformed into

$$MSE = \omega_u MSE_u + \omega_f MSE_f \tag{3.6}$$

where ω_u and ω_f are extra hyperparameters that must be tuned.

3.3 Applications of PINNs

PINNs have already found applications in a wide range of computational problems, serving as both solvers for differential equations and tools for estimating unknown parameters in dynamic systems. Initially, PINNs methodology was demonstrated on Burgers' equation, which is used in various areas, such as fluid mechanincs, gas dynamics, acoustics and traffic flow, as well as on the Schrödinger equation [11, 12, 13]. Furthermore, PINNs have been applied to solve stiff differential equation arising from chemical kinetics problems [14]. Epidemiological modeling has also adopted the PINN architecture, where PINNs were leveraged to estimate model's parameters using available epidemiological data. Another field that has implemented the PINN architecture is the development of epidemiological models [15]. In this work, the PINNs were exploited to estimate parameters of the model using available epidemiological data. Another intriguing application of the PINN architecture lies in the estimation of blood pressure [16]. However, in this paper the surrogate network was a CNN, exploiting the type of the available data that were in the form of time-series.

Moreover, the medical field has witnessed numerous applications of PINNs. For instance, in [17], PINNs were used to quantify kinetic parameters such as blood flow from dynamic contrast-enhanced magnetic resonance images (MRI). The field of drug compartmental kinetic models has also started to explore the potential of PINNs [18], although there are not many applications in this field. However, there is currently no published article that investigates the application of PINNs in conjunction with PBK models. In conclusion, PINNs hold promise in addressing challenges related to PBK development, particularly when dealing with complex systems of differential equations that are typically challenging to solve or when estimating model parameters presents significant obstacles.

Chapter 4

Pharmacometrics

4.1 Introduction

Pharmacometrics is the field of science that quantifies drug, disease and trial information to aid efficient drug development and/or regulatory decisions, as defined by the Food and Drug Administration (FDA) of United States [19]. It involves in the quantitative analysis of pharmacological and clinical data using mathematical and statistical models. It encompasses a multidisciplinary approach to quantify the interactions between drugs or toxic substances, the living organisms, and diseases, by interlinking the biology, physiology, and pharmacology with disease condition through mathematical models.

Pharmacometrics is a term to describe two different major families of models, which are called **Pharmacokinetic** (PK) and **Pharmacodynamic** (PD) models. These models fulfill diverse objectives, including the formulation of customised dosing plans for each drug and patient, as well as providing support for regulatory and drug development decisions [19]. PK models describe the absorption, distribution, metabolism, and excretion properties (**ADME**) of a drug from an organism during and after exposure, while the PD models describe the organism's response to this drug in terms of biochemical or molecular interactions. Therefore, ensembles of PK and PD models are usually used to characterize a drug, design successful dosing plans or to understand the concentrations-effects of a substance and an organism.

4.2 Pharmacokinetic Models

Pharmacokinetics defines this sub-category of pharmacometric models that are used to describe the ADME processes of a drug into an organism. These models present various structural differences between them and therefore, they are divided in other classes.

The first one that should be referred are the **Non-Compartmental** (NC) models. The NC models are usually used to estimate kinetic parameters from available concentration-time profile data of a drug, when the kinetics follow a first order pattern. The advantage of NC models is that they are based on the application of the trapezoidal rule in order to estimate the **Area Under the Curve** (AUC) of the concentration versus time. This feature makes them have minimal computational cost, rendering them accessible and easy to use. For instance, a NC model could predict the concentration of a drug in the blood,

when the metabolism processes of this drug follow first order kinetics. However, NC models should be avoided to be used as predictive tools for kinetics processes of higher order and cannot produce more detailed results such as concentration-time profiles for specific tissues of the organism [20].

This gap is partially addressed by the **compartmental** analysis. The compartmental models maintain a relatively simple structure, consisting of only a few compartments and the equations that govern the whole system are of low complexity. To elaborate further, compartmental models describe the fate of the drugs as they are distributed by the blood flow at the organs, but the representation of the organs as compartments remains rudimentary, such as the organ of interest could be modelled as a separated compartment and the rest organs are represented by a common compartment. This category of models have been proved to be useful in preliminary stages of the characterization of a drug and its kinetics. A more detailed approach of the compartmental analysis is the **Physiologically-Based Kinetics** (**PBK**) modelling.

4.3 Physiologically-Based Kinetics Models

Physiologically-based kinetic models form a distinct subset within compartmental models, that are characterized of elevated structural complexity and and integration of physiological information. These models simulate the ADME processes of various substances, whether they are drugs or possibly toxic compounds, across multiple organs and tissues within organisms. The first reported attempt for the development of a PBK model is attributed to Torsten Teorell in 1937 [21]. Since then, the development of PBK models has gone through various stages. Nowadays, this field of science has made significant progress, mainly because of the available and enhanced computational power and abundant availability of data accessible through the internet.

The term PBK is frequently encountered as PBPK, standing for physiologically-based pharmacokinetics, when the substance of interest is a drug. Another usual alternative is the PBTK, which stands for toxicokinetics when the substance is not a drug but possible toxic compounds. It is apparent that while the PBK methodology was originally developed for drugs applications, it has since found utility in a wider field of applications, such as nanomaterials [22] or Per- and polyfluoroalkyl substances (PFAS) [23], among others.

PBK models consist of multiple compartments, with each one representing an organ or tissue or a group of organs (e.g. the liver or the muscles or the gastrointestinal tract). Each compartment has specific physiological properties, like mass and blood flow. The primary goal of PBK models is to describe the ADME processes for the entire organism, and predict the concentration-time profiles for the different compartments equally well. While physiological parameters are usually available in literature [24, 25, 26] for a wide range of organisms, the PBK modelling methodology is often used to approximate parameters that are difficult or impossible to estimate experimentally. Therefore, by exploiting PBK models, researchers can leverage their predictive capabilities to gain insights into the complex ADME processes.

The distribution dynamics of drugs in the blood can be modelled with various ways.

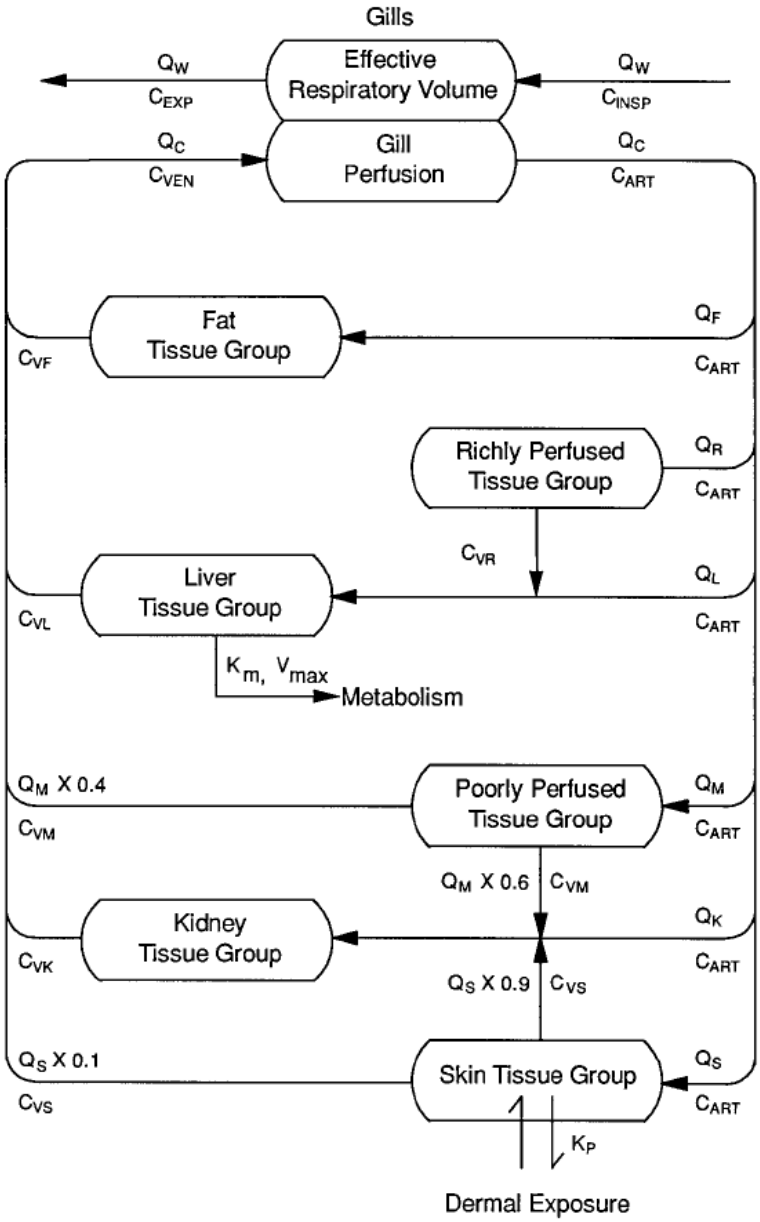

Figure 4.1. *Structural representation of the PBK developed in [1]. It is a model to simulate the dermal exposure of fish to various organic chemicals. The model consists of 7 compartments. The five of them (Fat, Liver, Kidney, Skin and Gills) are explicitly representing a unique organs, while the rest two compartments represent two different groups of organs. The organs are separated into the two groups based on the level of blood perfusion of the organ. The compartment of Gills is modelled with more details, as it is divided into two sub-compartments to include both the respiration and the perfusion processes.*

The two most common approaches to model the distribution in blood dynamics are the **flow-limited** (also known as perfusion-limited) and the **permeability-limited**.

The flow-limited approach is used when the substance diffuses from the capillary blood to the interstitial space and the only limiting factor is the blood flow in the capillaries of the specific organ. That means that the membranes of the capillaries offer negligible resistance at the diffusion of the substance from the blood into the interstitial space of the organ. Therefore, the tissue concentration is controlled by a constant parameter, called

tissue:blood partition coefficient [27].

The permeability-limited (or membrane-limited) approach is used when the limiting factor of the dynamics processes is the resistance that the substances face to transfer from the capillary blood. Moreover, there are differences at the level of the resistance between the organs, due to the differences of the cell membrane properties of each organ. The resistance to penetration of the substance is affected by multiple physicochemical properties [27].

The PBK models may differ in the number of the compartments and the structure of them. Some models consist of compartments that represent an organ as a total, while other models follow a more sophisticated design, where each compartment is divided into more sub-compartments and simulates the processes inside the organ with more details. Moreover, the group of organs represented into a PBK model is not standard. The selection of the organs that will be represented is usually a selection of which organs present higher interest for exposure at a specific drug or toxic compound. Another crucial factor in this decision is the availability and the quality of available biodistribution data that facilitate the inclusion of extra compartments. A frequent modelling decision is to group some organs into one compartment. This happens due to lack of experimental data (to model each one of these organs as a single compartment) and their biological processes are important to understand the ADME of a substance, as for example the organs of the gastrointestinal tract are often modelled as a single compartment because it is responsible for the fecal elimination of the substances [1]. Figure 4.1 is the structural representation of the model developed in [1] and includes some of the modelling approaches described above.

Chapter 5

Identifiability Analysis

5.1 Introduction

Mathematical modelling has evolved into a very powerful approach to simulate a large variety of scientific issues in silico, such as biological processes, chemistry problems, as well as pharmacokinetics modelling among others, and overcome the disadvantages of conducting experiments. However, biokinetics models, and notably PBK models, usually consist of a large set of parameters, which could either be physiological parameters or rate constants or partition coefficients that describe the kinetic of the substances. While physiological parameters for the majority of the species can be readily found in the literature, the same is not true for the kinetics parameters. These parameters tend to present high variability from substance to substance, or from species to species, or even between sexes of the same species. The values of these parameters are typically impractical to estimate experimentally, but they can often be estimated in silico using appropriate data. PBK modelling has proven to be an effective method for estimating these parameters with relatively low computational cost, provided that a sufficient amount of high-quality experimental data measured under specific experimental conditions is available.

However, throughout the process of PBK modeling, it is not always clear weather the available experimental data are adequate, in terms of quality and quantity, to estimate the entire set of unknown parameters in the model. Stated otherwise, given a biodistribution dataset and a PBK structure, there exists a trade-off between the number of the parameters that can be estimated using this dataset and the quality of the estimation of the parameters. This is due to the common issue that not all parameters can be estimated unambiguously, under specific circumstances. This issue is widely known as **non-identifiability** of the parameters. Therefore, the concept of identifiability of parameters answers the question whether all free parameters of a model can be uniquely derived from the available data, given a specific model structure. If the free parameters of the model are non-identifiable then the predictions of the model are not trustworthy and structural adjustments should be applied on the model, to address this issue [28]. If the free parameters of the model are non-identifiable then the predictions of the model are not trustworthy and structural adjustments should be applied on the model, to address this issue. Evidently, the identifiability of the parameters is a concern that should always be taken in consideration during model development, and a methodology should be applied to illuminate this issue.

Going deeper into the concept of identifiability, a definition should be given. In order to present the terminology, an example of chemical reaction model will be used as example model. Given a model $\mathcal{M}$ describing n species concentrations x_i in a chemical reaction network, described by a system o ordinary differential equations (ODE)

$$\begin{aligned} \dot{\vec{x}}(t) &= f(\vec{x}(t), \vec{u}(t), \vec{p}) \\ \dot{\vec{y}}(t) &= g(\vec{x}(t), \vec{s}) + \vec{\epsilon}(t) \end{aligned} \tag{5.1}$$

where $\vec{x}(t)$ are the internal states of the model, $\vec{u}(t)$ an externally given stimulus, $\vec{p}$ are the parameters of the model, g is an m-dimensional mapping of the internal states $\vec{x}$ to the observable variables $\vec{y}(t)$ involving scaling and offset parameters $\vec{s}$. $\vec{\epsilon}(t)$ represents noise of the measurements in the experimental dataset, that is assumed to be normally distributed around the mean values of the measurements. Given the initial conditions of the ODE system $\vec{x}(0)$, then the set of parameters that fully describes the $\mathcal{M}$ model is

$$\hat{\theta} = \{\vec{p}, \vec{x}(0), \vec{s}\} \tag{5.2}$$

Then, consider a metric $\chi^2(\theta)$ function used to measure the agreement of the model $\mathcal{M}$ output with the experimental data. This metric is **Weighted Sum of Squared Residuals** (WSSR)

$$\chi^2(\theta) = \sum_{k=1}^{m} \sum_{l=1}^{d} \left(\frac{y_{kl}^D - y_k(\theta, t_l)}{\sigma_{kl}^D} \right)^2 \tag{5.3}$$

where y_{kl}^D indicate the d data-points for each one observable output variable k, measured at specific time points, t_l. As σ_{kl}^D are noted the measurement errors, while $y_k(\theta, t_l)$ are the model predictions for a given set of parameters θ, at specific time points t_l. Therefore, an optimization problem has been set up, where the values of the parametric set θ can be estimated as

$$\hat{\theta} = \arg\min\left[\chi^2(\theta)\right]. \tag{5.4}$$

In case of the noise of measurements is normally distributed, $\vec{\epsilon} \sim N(0, \sigma^2)$ then equation 5.4 is equivalent to the maximum likelihood estimation (MLE) of θ:

$$\chi^2(\theta) = \text{const} - 2 \cdot \log \mathcal{L}(\theta) \tag{5.5}$$

where $\mathcal{L}(\theta)$ is the likelihood of θ. Therefore, the value of the constant does not have any impact on the values of $\hat{\theta}$, so the minimization of $\chi^2(\theta)$ is equivalent to the maximization of the likelihood. For this reason the χ^2 will be used as a placeholder for the likelihood from now on.

Having set up all the necessary mathematical notations about the model $\mathcal{M}$ and its parameters θ, as well as the optimized values $\hat{\theta}$, the definition of the identifiable parameter

can be given. A parameter θ_i is **identifiable**, if the confidence interval $[\sigma_i^-, \sigma_i^+]$ of its estimate $\hat{\theta}$ is finite. Otherwise, two different types of non-identifiability can be detected for any parameter θ_i; the structural non-identifiabity and the practical non-identifiabity (practical or numerical).

Finally, as confidence interval $[\sigma_i^-, \sigma_i^+]$ of a parameter $\hat{\theta}_i$ to a confidence interval α indicates that the true value θ_i^* is located within this interval with probability α. To estimate the confidence intervals the method of *finite sample confidence intervals* was applied, as presented in [29]. This methodology estimates the confidence intervals of the parameters by applying a threshold on the likelihood. Thus, the confidence intervals are estimated by defining the region

$$\{\theta | \chi^2(\theta) - \chi^2(\hat{\theta}) < \Delta_\alpha\} \text{ with } \Delta_\alpha = \chi^2(\alpha, df). \tag{5.6}$$

The threshold Δ_α indicates the α quantile of the χ^2 distribution with $df = 1$ for pointwise confidence intervals, that hold individually for each parameter or $df = \#\theta$ for simultaneous confidence intervals, that hold jointly for all parameters.

5.2 Structural non-identifiability

Following the definition given in [29], a parameter estimate θ_i is *structural identifiable*, if a unique minimum $\chi^2(\theta)$ with respect to θ_i exists. The structural non-identifiability issue arises from the over-parameterization of the model, which leads to a non-effective mapping g between the internal state variables $\vec{x}$ to the observable output variables $\vec{y}$. Consequently, the value of the metric $\chi^2(\theta)$ remains unchanged while the values of the ambiguous parameters $\theta_{sub} \subseteq \theta$ change. This means that there are multiple values of the θ parameters that can be used in the $\mathcal{M}$ model and receive exactly the same output. This situation indicates that there is high uncertainty on what is the true value of parameters θ, as well as what are their confidence intervals. Practically, when a parameter is structural non-identifiable then both confidence intervals are infinite $[-\infty, +\infty]$. Thus, the value of the parameter cannot be estimated at all.

At this point, it is important to clarify that structural non-identifiability is an issue derived exclusively from the mapping g between the internal state variable and the output variables and is not affected by the quantity and quality of the experimental data. Consequently, providing more data to estimate $\hat{\theta}$ would not aid to overcome the structural non-identifiability. Instead, only the modification of the g mapping can lead to make all the parameters structural identifiable. Modifying mapping g, could mean increasing the number of observable output y variables, or group some of the abundant parameters under a global parameter, decreasing this way the total number of parameters for estimation.

In the special case that the model $\mathcal{M}$ consists of only two parameters θ, then the $\chi^2(\theta)$ can be visualized as a landscape. If the two parameters are structurally non-identifiable, then the $\chi^2(\theta)$ is a totally flat valley infinitely extending in both directions of θ_1 and θ_2. In figure, panel A 5.1 reveals the shape of χ^2 and the functional relationship between the

parameters θ_1 and θ_2, in case of structural non-identifiability.

5.3 Practical non-identifiability

Following the definition given in [29], a parameter estimatation $\hat{\theta}_i$ is practically non-identifiable, if the likelihood confidence intervals region is infinitely extended in increasing or decreasing direction of θ_i, although the likelihood has a unique minimum for this parameter. The meaning of this parameter is that the confidence of practically non-identifiable are not both extended infinitely to both sides. Therefore, in the confidence interval $[\sigma_i^-, \sigma_i^+]$ either $\sigma_i^- = -\infty$ or $\sigma_i^+ = +\infty$. In the case of practical non-identifiable parameters, applying perturbations (towards the direction of the infinite confidence interval) on them results to negligible changes on the output variable of $\vec{y}$. In order to overcome practical non-identifiability issues, increasing the amount and/or decreasing the noise of the (already available) measurements can ultimately resolve these issues. Additionally, solving practical identifiability issues can provide useful insight in the process of experimental planning.

In the case of a two paremeters model $\mathcal{M}$ that the parameters are practically non-identifiable, the χ^2 is visualized like the in the panel B of figure 5.1. The confidence region of the parameters is infinitely extending to the direction of increasing parameters. Therefore, in the confidence interval $[\sigma_i^-, \sigma_i^+]$, the σ_i^- is finite and the σ_i^+ is infinite.

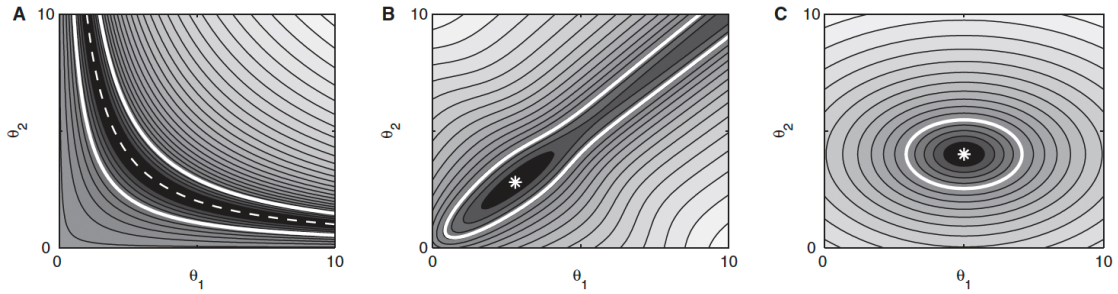

Figure 5.1. *Contour plots of $\chi^2(\theta)$ in case the parameter space of model $\mathcal{M}$ is two-dimensional. The colouring from white to black reveals the change of the χ^2 value from higher to lower values respectively. The thick white lines represent the likelihood-based confidence intervals. The white dashed line and the white asterisks represent the optimal value of χ^2. In panel A is represented the occasion that structural non-identifiability exists, where the optimal value of χ^2 infinitely extends, while the θ parameters increase and is not restricted into a specific area of the parametric space, so the confidence intervals of the parameters θ tend to infinite. Panel B illustrates the case of practical non-identifiable parameters. The optimal value is restricted into a specific area of the parametric space, but the likelihood-based confidence region is infinitely extended to one of the two directions. Panel C illustrates the case that both parameters are structurally and practically identifiable. The confidence region is finite for θ parameters into the parametric space.*

5.4 Profile Likelihood

It is obvious that the non-identifiability of the parameters of a model raises crucial trust concerns regarding the predictive ability of the model. Therefore, it is necessary to

approach this issue and apply a methodology to perform identifiability analysis and detect structural or practical non-identifiable parameters. For this purpose, the method that is applied in this diploma thesis is a methodology that exploits the **profile likelihood** method in order to perform the analysis, following the methodology published in [29]. The concept of the profile likelihood method can be summarised with the following equation

$$\chi^2_{PL}(\theta_i) = \min_{\theta_{j\neq i}} \left[\chi^2(\theta)\right] \tag{5.7}$$

implying that the profile likelihood of a parameter θ_i is estimated by re-optimizing the $x^2(\theta)$ with respect to every $\theta_{j\neq i}$. This process needs to be repeated for various values of θ_i in the parametric space around $\hat{\theta}_i$. Using a profile likelihood as approach to perform identifiability analysis has many advantages in terms of simplicity of the process and computational cost. The detailed steps of the identifiability analysis proposed in [29] using profile likelihood are the following:

1. Optimize the value of $\chi^2(\theta)$ and the optimal parameters $\hat{\theta}$
2. Define which θ_i parameter will be tested.
3. Estimate and apply θ_{step} on decreasing or increasing direction of θ_i.
4. Re-optimize $\chi^2(\theta)$ with respect to any $\theta_{j\neq i}$.
5. Repeat steps 3-4 until any of the following conditions is satisfied:
   - $\chi^2_{PL}(\theta_i)$ exceeded a threshold Δ_α,
   - Reached maximum number of iterations,
   - The value of θ_i exceeded its user-defined bounds.

There are multiple ways to estimate the θ_{step} of every iteration. The simpler one is to set θ_{step} to a constant value and take equal steps around $\hat{\theta}_i$. However, this approach does not take into consideration the steepness of the likelihood. In particular, θ_{step} should take higher values when exploring regions of θ_i at which the likelihood is flat. On the contrary, the steps should be smaller in regions where the likelihood increases. This way, the method does not spend computational sources to explore the area where the likelihood is flat and heads faster to regions where the likelihood increases and reaches the threshold. The alternative, suggested in [29], is to select the θ_{step} in an adaptive manner in order to accomplish what was previously described. Therefore, θ_{step} should fulfill the following equation

$$\chi^2(\theta_{last} + \theta_{step}) - \chi^2(\theta_{last}) = q \cdot \Delta_\alpha \tag{5.8}$$

where θ_{last} are the parameter values of the previous iteration and q takes a constant value in the range $[0, 1]$. The meaning of q is that when testing an identifiable parameter there

will be required at least $1/q$ steps to exceed the threshold Δ_α. Notably, larger values on q force to larger steps. In order to estimate θ_{step} of each iteration, it is necessary to numerically minimize the value of the function

$$f(\theta_{step}) = \chi^2(\theta_{last} + \theta_{step}) - \chi^2(\theta_{last}) - q \cdot \Delta_\alpha. \tag{5.9}$$

Finally, by following the profile likelihood approach, previously described, to perform identifiability analysis on the parameters of any model, the method classifies the parameters as:

- Structural non-identifiable: both σ_i^- and σ_i^+ are infinite, or
- Practical non-identifiable: only on of the σ_i^- and σ_i^+ is infinite, while the other's value has been estimated, or
- Identifiable: both σ_i^- and σ_i^+ are finite and have been estimated.

In conclusion, there are some significant advantages to using profile the likelihood method for conducting identifiability analysis. Firstly, the concept of the profile likelihood and the steps to apply it are as simple as possible. Additionally, this method is feasible to be applied even in cases of models with larger number of parameters, ensuring that the computational runtime remains within reasonable bounds.

Chapter 6

Perfluoroalkyl and Polyfluoroalkyl Substances - PFAS

6.1 Introduction

Perfluoroalkyl and Polyfluoroalkyl (PFAS) substances are a group of organic substances that have multiple fluorine atoms attached to an alkyl chain, creating very strong C-F bonds, such so these compounds contain at least one moiety of C_nF_{2n+1}- or -CF_2- (fully fluorinated methyl or methylene carbon atom, without any H/Cl/Br/I atom attached to it) [30]. The number of the fluorinated carbon atoms in PFAS varies from 4 to 17 for non-polymeric substances and have a great impact on their physicochemical properties, bioaccumulation characteristics and their protein-binding processes.

6.2 Classification of PFAS - Properties & Applications

According to [31], PFAS can be classified into non-polymeric and polymeric. The non-polymeric PFAS can be further divided into **perfluoroalkyl** and **polyfluoroalkyl** substances. The perfluoroalkyl PFAS consist only of fully fluorinated carbon atoms, whose chain is connected with a carboxylate (COO^-) or a sulfonate (SO_3^-) or a phosphate (OPO_3^-). On the contrary the polyfluoroalkyl PFAS contain at least one carbon (but not all) atom which is not fully fluorinated and its missing fluorine is replaced by an oxygen or hydrogen atom.

Fluorinated polymers are defined as the polymers in which one or more of the monomer units contains a fluorine atom. Therefore, not all fluorinated polymers belong to the large family of PFAS. We consider as polymeric PFAS the fluoropolymers (substances where the majority of the hydrogen atoms have been replaced by fluoride atoms), the side-chain fluorinated polymers (substances on fluorinated carbons attached to poly- or perfluoroalkylic side chains) and the perfluoropolyethers, where the carbon atoms of the main chain are connected directly to oxygen and fluoride atoms [3, 31].

PFAS substances posses material properties that have led to their widespread use in multiple applications [31]. Due to the strength of the C-F bond, the carbon-chain of the PFAS are hydrophobic and lipophobic substances. Those properties cause a wide range of PFAS substances to be highly effective surfactants, due to their ability to reduce the surface

tension of water effectively [31, 32]. Additional properties of PFAS that resulted in using them in a wide range of applications are the low reactivity, the good heat conductivity and the non-flammability, among others. Consequently, the most prevalent applications of PFAS substances include the production of fire-fighting foams, pesticide production, and in the aerospace, aviation, and automotive industries. They are also used in the paint, coating, and varnish industry, the medical field, the paper and packaging industry, electronics and semiconductors production, as well as food processing. The complexity of the PFAS have created a large group of different substances, which are employed in more than 100 sectors of industrial or consumer's products [3, 32].

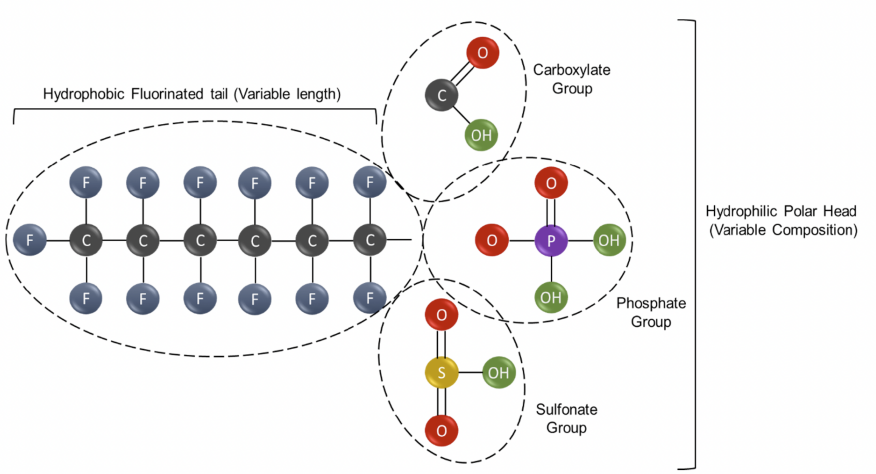

Image 6.1: *Example of non-polymeric perfluorinalkyl substance. They consist of two parts. The first one is the fully fluorinated carbon-chain, which is of variable length, and the second part which is a polar group, such as a carboxylate or a phosphate or a sulfonate group. The fluorinated carbons formulate a hydrophobic tail, while the polar group is hydrophilic [3].*

The extended use of PFAS in a large variety of applications has been a significant factor for the existence of multiple sources of pollution in the environment. Notably, the industrial production of fluoropolymers, the waste-water treatment plants and the remaining industrial processes where PFAS are involved, have contributed to significant releases of PFAS both in the atmosphere and the freshwater systems. Another major source of soil and groundwater pollution are the industrial plants, which are responsible for the recycling and the incineration of products that contain PFAS in their composition. Additionally, PFAS have been detected in multiple freshwater ecosystems around the world. It's clear that contamination of freshwater sites with persistent and non-degradable substances like PFAS raises concerns for the health of humans and animals. Those that consume water from these ecosystems and are part of the same food chain are potentially at risk [3].

6.3 Concerns regarding Health and Environment

Among the numerous PFAS substances, the two most common PFAS substances used in the industry and in the literature are the perfluorooctane sulfonic acid (**PFOS**) and the perfluorooctanoic acid (**PFOA**) [31]. The extensive use of PFAS has raised concerns

regarding potential negative effects on human health and the environment. Notably, the abnormally high concentrations of carbon fluorochemicals spotted and measures in human serum samples of non-industrially exposed individuals as early as 1970 [33] were attributed to PFAS, as reported in a 2001 publication [34].

Following those publications, many researches have studied the effects of PFAS in human health. Significant associations between the exposure to long-chain PFAS (compounds with more than 6 fluorinated carbons) and various human liver adverse effects have been detecteed, such as hepatic fat infiltration [4, 35] (also known as metamorphosis or steatosis), a liver disease where excessive amount of triglycerides are accumulated within the cytoplasm of the hepatocytes and is linked to obesity, diabetes mellitus and alcoholic liver disease [36]. Additionally, PFAS are connected with hepatocellular adenomas and cancers, the disruption of fatty acid trafficking, the induction of CYP P450 enzyme and hepatocyte apoptosis [4, 37].

Another organ which is instantly affected by the exposure to PFAS is the kidney, as PFOA and PFOS have been associated with the development of kidney toxicities. Histological experimental studies conducted on animals or human cell cultures have shown that the most commonly observed adverse effects on the kidney are tubular epithelial hypertrophy or even hyperplasia leading to increased kidney mass. Other observed effects include papillary necrosis, glomerular changes and even kidney failure when the exposure concentration was high [5].

Except the adverse effects regarding the kidney, it is important to refer to the PFAS re-absorption mechanisms occurring in the kidneys due to binding with specific proteins. Figure 6.2 illustrates the entire process of elimination and reabsoprtion of PFAS through the kidneys and urine. The re-absorption mechanism is strongly associated with the high half-life times of PFAS in organisms, leading to extended residence times and slower elimination processes. This re-absorption mechanism and the extended half-life times affect the kinetics and as a result it is often considered in the development of PBK models [23, 38, 39].

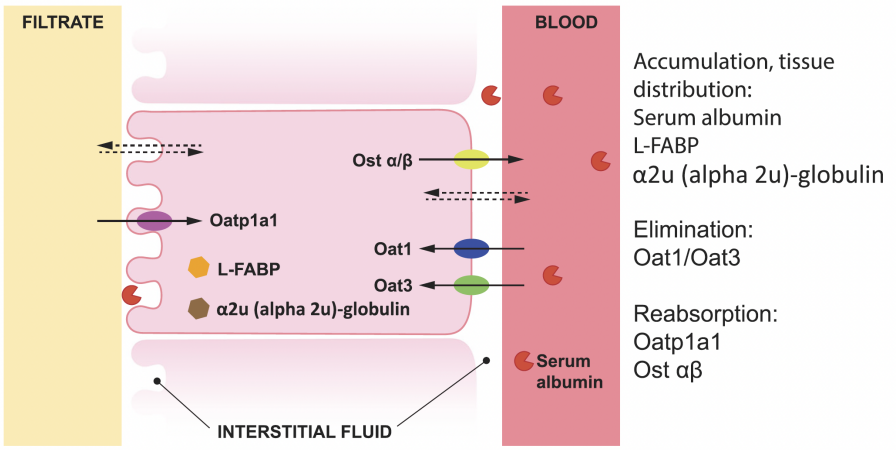

Image 6.2: *Schematic representation of the elimination and re-absorption mechanisms. The proteins responsible for the active transfer of PFAS from blood to the kidney tissue are the Oat1 and Oat3 transporters. One part of the amount of PFAS is successfully filtrated and excreted with the urine, while the rest of it is re-absorbed by binding to the Ost α/β proteins and being transferred back to the blood (to be distributed to the rest organs). Notably, the Oatp1a1 proteins are responsible for the re-absorption of PFAS from the urine back to the interstitial fluid of the kidney. An intriguing observation within this mechanism is the greater intensity of re-absorption observed in human males compared to females, exemplifying a difference in kinetics between the genders[4, 5].*

Chapter 7

Rainbow trout PBK

7.1 Introduction

In this chapter will present in detail the PBK model developed in this thesis, to predict the tissue-distribution of various PFAS in Rainbow trout (Oncorhynchus mykiss) after consumption of food spiked with these substances. The structure of the model is similar to the structure of the PBK model presented in [40]], albeit with some modifications. In addition, the model reported in [39] was taken into consideration, to simulate the renal excretion of PFAS. The experimental data used to estimate the kinetic parameters of the model, have been extracted from an experimental biodistribution study on Rainbow trout exposed to five different PFAS substances [41]. The first section of this chapter is about the biodistribution experimental data that are used for the estimation of the parameters of the model.

7.2 Biodistribution Data for Dietary Exposure of R.trout to PFAS

The development and training of PBK models requires data that provide tissue-specific concentration-time profiles. It is also important to provide detailed information about the exposure conditions of the organisms. Additionally, it is necessary for the experiment to have long enough duration to reveal the underlying kinetics of the ADME processes.

The biodistribution data that are used in this diploma thesis are published in [41]. The researchers examined the distribution of 5 different PFAS substances in the tissues and blood of dietary exposed rainbow trout. The examined substances are perfluorobutane sulfonic acid (PFBS), perfluorohexane sulfonic acid (PFHxS), perfluorooctane sulfonic acid (PFOS), perfluorooctanoic acid (PFOA) and perfluorononanoic acid (PFNA). The fish were exposed in parallel to those substances for a period of 28 days and they were fed daily with food mass equal to the 2.6% of their mean weight. The concentration of PFAS in the food was 500 μg/kg of food. The mean temperature of water was 15°C. The exposure period was followed by a depuration phase, which also lasted 28 days. Incorporating the depuration period into the experiment is significant, because this way it is feasible to understand and quantify the kinetics behind the clearance processes. Measurements for liver, muscle, skin, gills, kidney blood and carcass were received at 7, 14, 28, 31, 35, 42 and 56 days from the

beginning of the experiments. So, the dataset contains 3 data points during the exposure period and 4 data points during the depuration. No data were available for the excretion of PFAS, such as urine or fecal concentrations.

The data were available in plots and not in digital format. For this reason the digitization of the plots was achieved using a free plot-digitizing software named Graph Grabber (version 2.0.2) [42].

7.3 PBK Structure and ODEs

As previously documented, the main structure of the model is similar to the model developed in [40] incorporating some important modifications in the modelling process. The model consists of 7 compartments that represent the tissues of the fish, alongside two extra compartments for the arterial and venous blood. The tissues that are modeled as compartments are the Gills, Viscera, Liver, Kidney, Muscle, Skin and Carcass. The viscera compartment refers to a group of organs, which includes stomach, intestine, pyloric cecum and spleen. Although there was no biodistribution data for these compartments, it was considered that they have a significant contribution in the distribution of PFAS among the tissues, especially in scenarios involving PFAS exposure through dietary intake. Finally, the Carcass compartment represents the remaining tissues of the fish that are not explicitly represented as compartments. A detailed structural representation of the model is presented in figure 7.1.

In the following differential equations both concentrations and absolute mass of PFAS are estimated for each compartment of the model. The concentration of PFAS is estimated as

$$C_i = \frac{M_i}{w_i} \tag{7.1}$$

where w_i is the mass of the compartment i.

However, physiological parameters, like weights and the blood flows are not constant in the model. Instead, the growth of the fish is taken into account by the model. The aim behind this modelling decision is to estimate the concentrations based on the correct tissues weights, as the biodistribution data span a long term, during which the fish are increasing their mass significantly. As a result , the article containing the biodistribution data [41], also reports the mean total weight of the fish at the beginning, middle and end of the experiment. Assuming that the body mass growth of the fish is a linear process, it was easy to estimate and update the fish body weight at each time of the simulation, using linear interpolation. This way, the physiological parameters that are dependent on the body weight are updated for each time point of the simulation.

Concerning the estimation of the total blood flow Q_{total}, it is estimated as proposed in [25] and [43] taking into consideration the effect of experimental temperature as an additional factor, according to Arrhenius equation [44]. The Arrhenius equation is

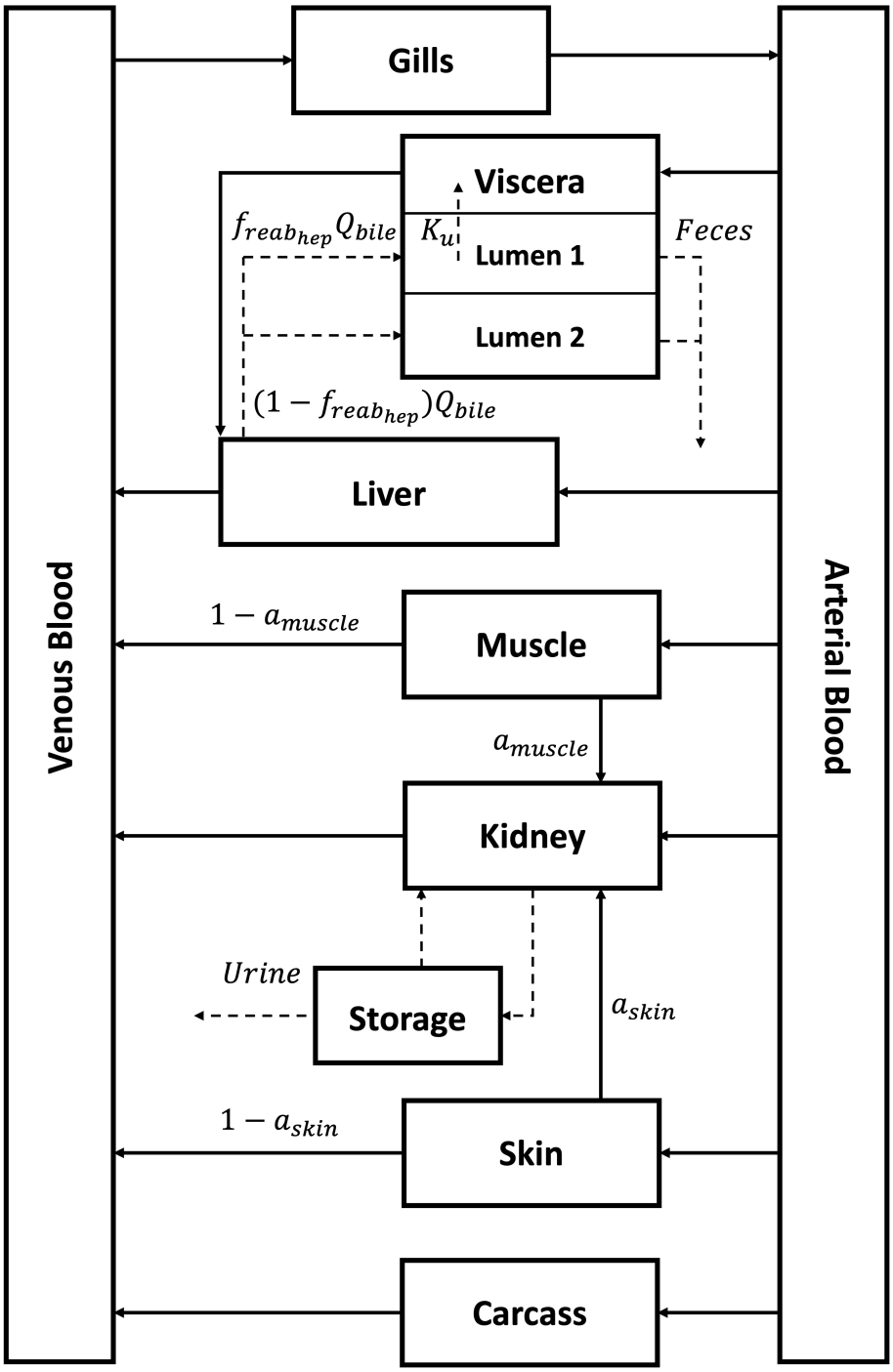

Figure 7.1. *Schematic representation of the PBK model developed for Rainbow trout exposed to PFAS through the food. The model consists of 8 compartments, including the blood which is divided into two compartments for the arterial and the venous blood. The compartment of viscera is used to model the stomach, intestine, pyloric cecum and spleen as a group. The exposure to PFAS through the food consumption is considered to occur at the Lumen 1 sub-compartment of viscera, by adding the amount of PFAS that is considered to be eaten. This PBK models two elimination pathways; through the urine and the feces. Moreover, reabsorption of PFAS from urine back to blood through the kidney is supposed to occur. The enterohepatic circulation of PFAS has also been modeled. Finally, the Carcass compartment represents the rest organs and tissues that have not been modeled explicitly.*

$$A_T = exp(\frac{T_A}{T_r} - \frac{T_A}{T_{exp}}) \tag{7.2}$$

where T_{exp} is the water temperature (°K), T_r is the reference temperature (°K) at which the reference value of the total blood flow is measured and T_A is the Arrhenius temperature (°K). As a result, the total blood flow of the fish is estimated using the terms for the temperature and mass correction through the equation

$$Q_{Total} = Q_{Total_{ref}} \times A_T \times \left(\frac{BW}{BW_{ref}}\right)^{-0.1} \times BW \times plasma \tag{7.3}$$

where $Q_{Total_{ref}}$ is the total blood flow reference value, BW is the body weight of the fish, BW_{ref} is the body weight at which the $Q_{Total_{ref}}$ is reported and $plasma$ is the percentage of the plasma volume to the total blood volume. Since the PFAS do not partition to red blood cells, the distribution is assumed to be based only on the plasma flow through the tissues, so we had to estimate the plasma flow rate instead of the total blood flow [40].

The equation that describes the distribution of PFAS in compartments that do not have any uptake or any elimination is

$$\frac{dM_i}{dt} = Q_i(C_{art} - \frac{C_i}{P_i}) \tag{7.4}$$

where M_i is the mass of the PFAS in compartment i, Q_i is the arterial plasma flow of compartment i, C_{art} is the concentration of PFAS in arterial blood, C_i is the concentration of PFAS in tissue i and P_i is the partition coefficient of compartment i. The compartments of the model that are described from this equation are Muscle, Skin and Carcass.

As it was previously stated, the gastrointestinal tract of fish was modeled as a single compartment, named Viscera, which represents the stomach, the intestine, the pyloric cecum and the spleen. These organs contribute significantly in the absorption of PFAS from food and they are recirculate and eliminate PFAS through the enterohepatic circulation and the fecal elimination. This compartment is divided into 3 sub-compartments. The first sub-compartment is used to model the amount of PFAS inside the lumen of the viscera. This amount is considered to be the total amount of PFAS that become available for absorption to the organism with the amount received from the food and the amount that becomes available from the enterohepatic circulation. This amount is described as **Lumen 1** in the model.

The second sub-compartment of the viscera represents the amount of PFAS which is inside the bile flow and is unavailable for reabsorption. This sub-compartment is called **Lumen 2** inside the model and the equations. It is important to clarify that the fecal elimination of PFAS takes part in both **Lumen 1** and **Lumen 2**.

The third sub-compartment models the tissue of the viscera. It is an important sub-compartment because through this the PFAS are absorbed from the food which is inside the lumen and then are distributed to the other tissues and organs.

The differential equation for Lumen 1 sub-compartment is

$$\frac{dM_{lumen_1}}{dt} = f_{reab_{hep}} Q_{bile} C_{liver} - K_u \alpha M_{lumen_1} - Cl_{feces}(1-\alpha) M_{lumen_1} \tag{7.5}$$

where $f_{reab_{hep}}$ is the percentage of PFAS in bile (Q_{bile}) which becomes available for reabsorption, K_u is the uptake rate of PFAS from lumen to the tissue of the viscera, α is the assimilation efficiency of PFAS from the food and Cl_{feces} is the fecal elimination rate constant of PFAS amount.

Considering that the Lumen 2 sub-compartment is used to model the amount of PFAS from enterohepatic circulation that is not available for reabsorption and is eliminated through the feces, the correspondent equation is the following

$$\frac{dM_{lumen_2}}{dt} = (1 - f_{reab_{hep}}) Q_{bile} C_{liver} - Cl_{feces} M_{lumen_2}. \tag{7.6}$$

The last sub-compartment of viscera represents the tissues of the visceral organs and is modelled with an equation similar to the general equation presented previously for muscle, skin and carcass, but has an additional term describing the absorption of PFAS from food. So the equation for the tissue of the viscera is

$$\frac{dM_{viscera}}{dt} = Q_{viscera}(C_{art} - \frac{C_{viscera}}{P_{viscera}}) + K_u M_{lumen_1}. \tag{7.7}$$

The next differential equation pertains to the Liver. This equation includes additional terms: one for the incoming flow from the viscera and another for the outflow of PFAS through the bile. Consequently, the equation for the liver is as followes:

$$\frac{dM_{liver}}{dt} = Q_{liver} C_{art} + Q_{viscera} \frac{C_{viscera}}{P_{viscera}} - (Q_{liver} + Q_{viscera}) \frac{C_{liver}}{P_{liver}} - Q_{bile} C_{liver} \tag{7.8}$$

where Q_{bile} refers to bile flow, not the blood blood.

The following equation describes the kinetics of the kidney compartment.

$$\begin{aligned} \frac{dM_{kidney}}{dt} =& Q_{kidney} C_{art} - (Q_{kidney} + a_{muscle} Q_{muscle} + a_{skin} Q_{skin}) \frac{C_{kidney}}{P_{kidney}} \\ &+ a_{muscle} Q_{muscle} \frac{C_{muscle}}{P_{muscle}} + a_{skin} Q_{skin} \frac{C_{skin}}{P_{skin}} - Cl_{urine} CLU_{coef} M_{kidney} \\ &+ f_{reab_{urine}} M_{storage} \end{aligned} \tag{7.9}$$

It's worth noting that the kidney compartment receives two additional plasma flows apart from the arterial flow, which come from the muscle and skin compartments. This is modeled to reflect that a relatively high percentage of the venous blood (blood outflow) from these compartments ultimately flows directly to the kidney [1, 44].

The mean maximum volume of the urinary bladder is considered as storage. Therefore, $M_{storage}$ is the amount of PFAS present in the urine the urinary bladder, which is available either for excretion from the organism or for reabsorption back to the blood stream. Thus, the differential equation for the storage compartment is as follows:

$$\frac{dM_{storage}}{dt} = Cl_{urine}CLU_{coef}M_{kidney} - f_{reab_{urine}}M_{storage} - Q_{urine}C_{storage} \tag{7.10}$$

where, CL_{urine} is the urinary elimination rate, CLU_{coef} is a correction coefficient of the CL_{urine}, because its value was taken from another published article (more details in section 7.5), $f_{reab_{urine}}$ is the reabsorption rate of PFAS from urine to the organism, Q_{urine} is the outflow of urine and $C_{storage}$ is the concentration of PFAS inside the urine contained in the urinary bladder and it is calculated as

$$C_{storage} = \frac{M_{storage}}{V_{storage}} \tag{7.11}$$

where $V_{storage}$ is the maximum volume of the urinary bladder. The $f_{reab_{urine}}$ is estimated relying on a theoretical constant coefficient comparing the reabsorption rate to the elimination rate. So the relationship between the two rates is

$$K_{urine} = \frac{CLU_{coef}Cl_{urine}}{f_{reab_{urine}}}. \tag{7.12}$$

Therefore, the ODE for the PFAS eliminated through the urine is

$$\frac{dM_{urine}}{dt} = Q_{urine}C_{storage}. \tag{7.13}$$

The next differential equation is used to estimate the PFAS excreted with the feces. So the total amount of PFAS excreted through the feces is

$$\frac{dM_{feces}}{dt} = Cl_{feces}((1-a)M_{lumen_1} + M_{lumen_2}) \tag{7.14}$$

which is actually the sum of the unavailable for absorption PFAS amount from the compartment Lumen 1 and the amount of the compartment Lumen 2, which is the amount of PFAS in bile that is not available for reabsorption.

The last two remaining equations are to estimate the PFAS in arterial and venous blood. As figure 7.1 indicates, the Venous Blood compartment feeds the gills compartment with the total blood flow, while it receives blood flows from the Liver, Muscle, Kidney, Skin and Carcass compartments. Therefore, the equation for the estimation of PFAS in Venous Blood is

$$\begin{aligned}\frac{dM_{venous}}{dt} = &- Q_{total}C_{venous} + (Q_{liver} + Q_{viscera})\frac{C_{liver}}{P_{liver}} + \\ &(Q_{kidney} + a_{muscle}Q_{muscle} + a_{skin}Q_{skin})\frac{C_{kidney}}{P_{kidney}} + \\ &(1 - a_{muscle})Q_{muscle}\frac{C_{muscle}}{P_{muscle}} + (1 - a_{skin})Q_{skin}\frac{C_{skin}}{P_{skin}} + \\ &Q_{carcass}\frac{C_{carcass}}{P_{carcass}}.\end{aligned} \tag{7.15}$$

In contrast, the Arterial blood compartment receives the total blood flow from the Gills compartment and feed the rest organs, so the correspondent equation is

$$\frac{dM_{art}}{dt} = Q_{gills}\frac{C_{gills}}{P_{gills}} - (Q_{viscera} + Q_{liver} + Q_{kidney} + Q_{muscle} + Q_{skin} + Q_{carcass})C_{art}. \tag{7.16}$$

Concluding, those were all the equations that the model consists of. Those equations are in total agreement with the mass and the flow equilibrium of the problem, which is an important condition to be satisfied in the development process of PBK models.

7.4 Physiological Parameters

The current PBK model consists of physiological parameters, which of course are independent of the substance of exposure. On the other hand, some of the parameters of the model are PFAS-specific. Consequently, the value of these parameters are unique for every PFAS substance. While the total of the physiological parameters were estimated exclusively using the literature sources, this is not true for the PFAS-specific parameters too. In this section a detailed description of the physiological parameters of the model will be provided with their units.

A short description of each physiological parameter of the model, their symbols and their units are given in the table 7.1. All parameters reported in this table are considered to be independent of the substance of exposure.

The mass of each tissue i is estimated as

$$w_i = f_{w_i} \times BW \tag{7.17}$$

where f_{w_i} is the weight of tissue i as a fraction to the total body weight BW of the fish. The correspondent blood flow is estimated as

$$Q_i = f_{b_i} \times Q_{Total} \tag{7.18}$$

where f_{b_i} is the blood fraction of the blood flow of tissue i to the total blood flow Q_{total}.

Table 7.1. *Definitions, symbols and units of the parameters used in the R. trout PBK model.*

Definition	Parameter	Unit
Body weight	BW	g
Body weight reference value	BW_{ref}	g
Total blood flow reference value	$Q_{Total_{ref}}$	ml/h/g
Water temperature	T_{exp}	°K
Arrhenius temperature	T_A	°K
Reference temperature	T_{ref}	°K
Tissue weight coefficients (as fraction of total body weight)	f_{w_i}	-
Mass of compartment i	w_i	g
Regional blood flow as fraction of total blood flow	f_{b_i}	-
Regional blood flow of compartment i	Q_i	ml/h
Fraction of PFAS available for reabsorption coming from enterohepatic circulation	$f_{reab_{hep}}$	-
Fecal elimination rate	Cl_{feces}	1/h
Renal elimination to reabsorption ratio	K_{urine}	-
Urinary elimination rate	Cl_{urine}	1/h
Correction factor of the urinary elimination rates	CLU_{coef}	-
Reabsorption rate of PFAS from urine	$f_{reab_{urine}}$	1/h
Urine flow rate coefficient	$Q_{urine_{coef}}$	ml/g/h
Maximum urine volume coefficient inside urinary bladder	$V_{urine_{coef}}$	ml/g
Bile flow rate coefficient	$Q_{bile_{coef}}$	ml/g/h
Fraction of skin blood flow that flows directly to kidney	a_{skin}	-
Fraction of muscle blood flow that flows directly to kidney	a_{muscle}	-
Fraction of plasma volume to total blood volume	$plasma$	-
Uptake rate from lumen_1 to viscera tissue	K_u	1/h
Partition coefficient of compartment i	P_i	-
Assimilation efficiency	a	-
Assumed density of blood and tissues = 1	ρ	g/ml

Therefore, to estimate the plasma flow instead of blood flow, then the equation 7.18 must be multiplied by the *plasma* fraction of blood and finally take

$$Q_i = f_{b_i} \times Q_{Total} \times plasma. \tag{7.19}$$

Finally, the weight of the Carcass compartment are estimated as the remaining mass of the organism,

$$\begin{aligned} w_{carcass} =& BW - (w_{blood} + w_{liver} + \\ & w_{skin} + w_{muscle} + w_{gills} + \\ & w_{kidney} + w_{viscera} + w_{lumen}) \end{aligned} \tag{7.20}$$

and the blood flow of the remaining organism is

Table 7.2. *Values and references of the physiological parameters values.*

Parameter	Value	Reference
BW_{ref}	270.1 (for T=6 °C)	[25]
	296.4 (for T=12 °C)	[25]
	414.5 (for T=18 °C)	[25]
$Q_{Total_{ref}}$	1.188 (for T=6 °C)	[25]
	2.322 (for T=12 °C)	[25]
	3.750 (for T=18 °C)	[25]
T_{exp}	288	[41]
T_A	6930	[44]
T_{ref}	279	[44]
	285	[44]
	291	[44]
$f_{b_{liver}}$	0.0035	[40]
$f_{b_{skin}}$	0.073	[40]
$f_{b_{muscle}}$	0.655	[40]
$f_{b_{gill}}$	0.0002	[40]
$f_{b_{kidney}}$	0.071	[40]
$f_{b_{viscera}}$	0.069	[40]
$f_{w_{liver}}$	0.012	[40]
$f_{w_{blood}}$	0.045	[40]
$f_{w_{skin}}$	0.064	[40]
$f_{w_{muscle}}$	0.566	[40]
$f_{w_{gill}}$	0.02	[40]
$f_{w_{kidney}}$	0.016	[40]
$f_{w_{viscera}}$	0.051	[40]
$f_{w_{lumen}}$	0.012	[40]
$Q_{bile_{coef}}$	0.000075	[45]
$Q_{urine_{coef}}$	2.76	[46]
$V_{urine_{coef}}$	0.0022	[46]
a_{skin}	0.9	[1, 44]
a_{muscle}	0.6	[1, 44]
$plasma$	0.7	[26, 47, 48]

$$\begin{aligned} Q_{carcass} =& Q_{Total} - (Q_{liver} + Q_{skin} + \\ & Q_{muscle} + Q_{kidney} + Q_{viscera}). \end{aligned} \tag{7.21}$$

The values of all physiological parameters as well as the reference source are provided in table 7.2.

7.5 PFAS-Specific Parameters

In this section a detailed presentation of the PFAS-specific parameters will be provided. The PFAS-specific parameters, refer to the special feature of these parameters that they

have a distinct value for each one of the PFAS considered in this model. For some of the PFAS-specific parameters it was feasible to find relative values in the literature. The parameters that were given a constant value based on available literature were the assimilation efficiency, a, the coefficient of the enterohepatic circulation, $f_{reab_{hep}}$, the ratio of the renal elimination/reabsorption rates, K_{urine} and the renal elimination rates Cl_{urine}. The values of these parameters for each one of the PFAS as well as the reference literature are provided in table 7.3.

Table 7.3. *The values of the PFAS-Specific parameters and the corresponding literature sources.*

Parameter	PFOA	PFNA	PFBS	PFHxS	PFOS	Reference
α	0.138	0.522	0.0598	0.558	0.721	[49, 50]
$f_{reab_{hep}}$	0.30	0.340	0.230	0.30	0.420	[51]
K_{urine}	2.080	1.350	10.410	5.880	1.350	[39, 50]
Cl_{urine}	104.40	180.0	82.80	82.80	180.0	[39, 50]

7.5.1 Assimilation Efficiency

The assimilation efficiency values were set to constant values, using the values given in [49] in which the experimental conditions were very similar to those in the paper [41], from which the experimental data used for the model presented in this thesis. Additionally, those values were also used in food-web bioaccumulation model for fish [50]. As indicated in equation 7.5, the assimilation efficiency is multiplied by the K_u and the M_{lumen_1}, in order to describe the mass of PFAS transferred to the viscera tissue per time unit. Considering that lack of available data for the concentration of PFAS into the viscera tissue, it was clear that a modelling approach with using constant assimilation efficiency parameters fixed using the literature, could lead to a feasible estimation of a K_u parameter which is common for all PFAS substances. In other words, the assimilation efficiencies incorporate all the differences that exist between the PFAS, regarding the kinetic processes during their transfer from the lumen space to the viscera tissue. Finally, instead of estimating 5 different parameters (one for each PFAS substance) to describe the rate constant of transfer of PFAS from lumen to tissue, only one common K_u needs to be estimated from the data.

7.5.2 Enterohepatic Circulation Coefficient

The $f_{reab_{hep}}$ describes the fraction of PFAS that becomes available for absorption due to the enterohepatic circulation. The term of enterohepatic circulation refers to the secretion of bile acids (acidic steroids formed from cholesterol in the liver) from the liver, via bile flow into the intestine and some of them are reabsorbed by the blood and the liver [50, 52, 53]. The similarity of the PFAS and the bile acids was studied in [51] in the light of enterohepatic circulation may affect the distribution of PFAS. The main reason is that some PFAS can connect (as the bile acids do) to the key transport proteins that regulate the enterohepatic circulation. Notably, the binding affinity of various PFAS with the

transport proteins named apical sodium-dependent bile acid transporter (ASBT) and the Na+ taurocholate transport polypeptide (NTCP) was estimated. These proteins were selected as they play a key role for the secretion of bile acids at the enterohepatic circulation. Then the estimated binding affinities of the PFAS were directly compared to the binding affinity of taurocholic acid (TCA) with the two transporter proteins, which is the primary component of bile acids and is known to be highly reabsorbed through the enterohepatic circulation (up to 95%). Therefore, the coefficients of reabsorption were estimated by making this comparison, considering the TCA as a "benchmark" substance that has $f_{reab_{hep}}$ equal to 0.95. The values for the 5 PFAS are reported in table 7.3. Reading the given values, it seems that the PFAS are connected to some extent to the transporter proteins and there is a high possibility to be reabsorbed through the enterohepatic circulation. The most likely to be reabsorbed is the PFOS, while the PFBS has the lower coefficient.

7.5.3 Renal Elimination-to-Reabsorption Coefficient

The reabsorption of PFAS from the urine back to the kidney tissue is a very common mechanism for various organisms, especially for humans, increasing this way the half-life (even to a range of years [5]) of the PFAS significantly. As it was previously described in image 6.2, the elimination and the reabsorption of PFAS through the urine is a protein regulated process and the transporter proteins with the higher impact are the Oat1 and Oat3 on the clearance process and Oatp1a1 on the reabsorption process. The developers of the PBK model published in [39] estimated first order urinary clearance and reabsorption rates based on the uptake rates of the Oat1, Oat3 and Oatp1a1 proteins. Therefore, it was feasible to estimate the elimination/reabsoprtion ratio, noted as K_{urine} in table 7.3. Using those elimination/reabsorption coefficient values provides significant insight into the protein facilitated reabsorption mechanism of PFAS, especially taking into account that the dataset used to fit the parameters of the model did not include any data for PFAS concentrations in urine.

7.5.4 Renal Elimination Rate Constants

The final group of PFAS-Specific parameters that were fixed to constant values are the urinary elimination rate constants Cl_{urine}. As it was previously noted, the biodistribution data did not contain any information about the elimination of PFAS, either fecal or urinary. This obstacle led to search for already estimated renal elimination rates of PFAS in fish. Therefore, the values for the urinary elimination rate constants were taken by [50], considering that their model refers to rainbow trout.

7.6 Parameters Estimation - Optimization workflow

It has already been evident that deciding about which parameters to fit using the biodistribution data is a complex task. It is necessary to keep a balance between the number of parameters and the goodness of fit of the model. Overloading the model with

excessive parameters to enhance its predictive ability can lead to significant identifiability issues, as discussed in Chapter 5.

Therefore, the first step involved defining a set of free parameters that will be fitted using the data. To estimate those parameters, the selection of an optimizing algorithm and a fitness metric is required. For this task, a custom metric was developed . The metric function is called **Physiologically-Based Kinetics Objective Function** or **PBKOF** and is a modified version of the PBPK Index presented in [54]. Considering the PBK models consist of multiple output variables that are fitted on the experimental dataset, N_{comps} is the number of compartments for which the data are available. Additionally N_{obs} indicates the number of measurements available for any given compartment. Therefore, the value of the PBKOF metric for any compartment j of interest is estimated as

$$I_j = \sqrt{\frac{\sum_{i}^{N_{obs_j}} \left(\frac{y_i - \hat{y}_i}{y_i}\right)^2}{N_{obs_j}}} + \sqrt{\frac{\sum_{i}^{N_{obs_j}} \left(\frac{y_i - \hat{y}_i}{\hat{y}_i}\right)^2}{N_{obs_j}}} \tag{7.22}$$

Finally, the overall goodness of fit of the model, regarding all the total compartments is equal to

$$\text{PBKOF} = \sum_{j=1}^{N_{comps}} I_j \frac{N_j}{N_{Total}} \tag{7.23}$$

where N_{Total} is the sum of observations for all compartments, thus $N_{Total} = \sum_{j=1}^{N_{comps}} N_{obs_j}$. The weighting factor N_j/N_{Total} is used in order to give more focus on compartments with higher number of observations than others.

The next step of the workflow is to create a function that takes as input the values of the free variables. These values will be used internally to solve the differential equations. Subsequently, the results of the ODEs will serve as input to the PBKOF metric to evaluate their alignment with the experimental data. Finally, the function will return the PBKOF metric value. This function will be used from the optimizing algorithm to minimize the PBKOF and estimate the values of the unknown parameters from the data.

The next part of the process is to perform identifiability analysis using the optimal values of the parameters and detect any non-identifiable parameters. If non-identifiable parameters exist, the model structure must be modified and repeat the whole process until the parameters become identifiable and the goodness of fit is satisfactory.

The described workflow was implemented in R (version 4.3.1) [55]. For the solution of the ODEs the deSolve [56] R package (version 1.36) was used, while the nloptr [57] (version 2.0.3) was exploited for the optimization problem. Finally, an implementation of the identifiability analysis concept was written in R code, exploiting the profile likelihood method that was described in chapter 5.

Finally, during this diploma thesis a custom R package was created to automate the call of functions that are used frequently. Notably, the package contains functions to estimate

profile likelihood and perform identifiability analysis on any ODE model the user desires. Moreover, the package contains a group of metrics, that were developed in colaboration with Periklis Tsiros, PhD candidate. This metrics are used to evaluate the goodness of fit of regression models (as well as PBK modelling). An implementation for local sensitivity analysis has also been added in the package. The purpose of this package is to contain various tools that are used in kinetics and PBK modelling. The name of the package is **PBKtools** and is available on Github at https://github.com/ntua-unit-of-control-and-informatics/PBK_modelling_tools.git

7.7 Parameters Estimation - PINN workflow

PINNs can easily be applied to estimate the unknown parameters of any nonlinear dynamic system, such as the PBK model developed in this thesis. To evaluate the PINNs as an alternative approach for parameters estimation, a new workflow using PINNs was set up. Firstly, in this section it was considered that the common parameters Ku, CLU_{coef} and CL_{feces} are constant and set equal to the values estimated using the optimization workflow described in section 7.6. Additionally, the simple scenario of estimating the partition coefficients only for the PFOS substance was considered. This decision was made to test the ability of PINNs in estimating parameters on an easier to solve problem. Additionally it is more common estimating unknown parameters of a PBK model for a single substance, than for multiple substances simultaneously. However, addressing this problem as a subsequent step is highly interesting, because in these problems the computational cost of the optimization workflow is increased. Consequently, a PINN was used to estimate the partition coefficients $P_{liver}, P_{muscle}, P_{kidney}, P_{skin}, P_{gills}, P_{carcass}$ and $P_{viscera}$ only for the PFOS experimental data.

The training of the PINN model was accomplished using the **DeepXDE** [58, 59], a python library for solving computational problems, forward or inverse, using PINNs. DeepXDE facilitates the coding of the PINNs and is suitable for both educational and research purposes.

To estimate the parameters of the model, a PINN was structured using a feed forward network. The decision to use this structure was based on various applications of PINNs in other publications, that showed that FFN achieve satisfying results without the need to use more complex networks in problems similar to this one. Training the PINN requires tuning the neural network's hyperparameters. After executing some preliminary tests, the learning rate, the optimizer and the activation function were fixed to constant values that provided satisfying results. Therefore, the parameters that needed tuning were the number of the hidden layers (N_{hidden}), the number of nodes per layer ($N_{neurons}$), the number of collocation points ($N_{collocation}$), and the value of the weight ω_u that is multiplied with the MSE_u term in equation 3.3.

The best set of hyperparameters was defined after applying the grid search technique. However, the simultaneous grid search of four parameters would lead to a huge number of FFN structures to be tested. For this reason, tuning was realised in two stages. At the first stage only the N_{hidden}, $N_{neurons}$ and $N_{collocation}$ were tuned. Notably, three values for

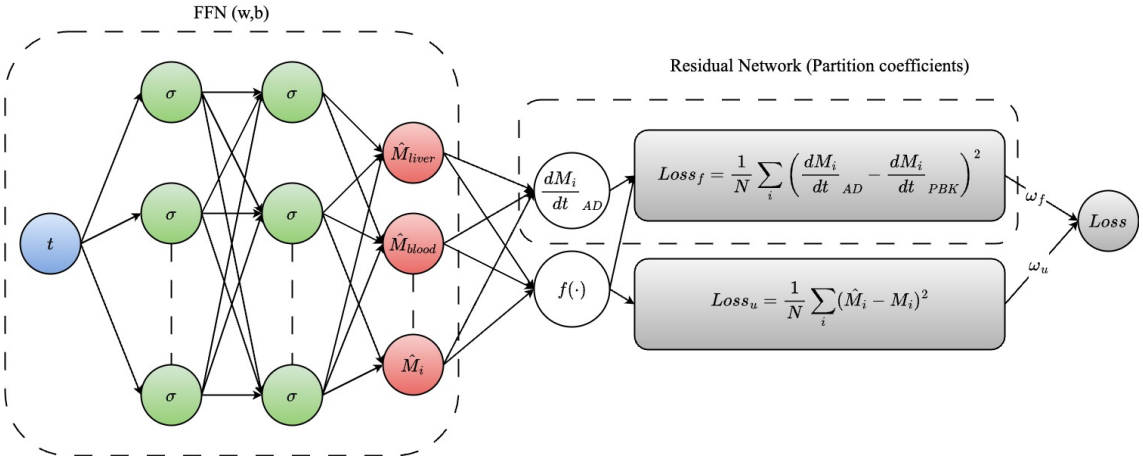

Figure 7.2. *Schematic representation of PINN used to estimate the partition coefficients of the PBK model. The first part of the PINN is a feedforward network, which receives as input only time. The output of the network are the predicted values of the state variables of the PBK model $\hat{M}_i$ Those values are used to estimate the agreement with the experimental data as $Loss_u$. The predictions are also provided to the residual network. The residual network estimates the derivatives of each output of the FFN with respect to input t and compares it with the corresponding values from the PBK model. The evaluation of this difference (in terms of mean squared error) formulates the second loss term, the $Loss_f$. Finally, the two loss terms are scaled with the values ω_f and ω_u and they are summed to calculate the total loss. The minimization of $Loss_u$ is achieved with respect to the weights w and biases b of the FFN. Instead, the minimization of $Loss_f$ is achieved with respect to the partition coefficients of the PBK model too.*

each parameter were tested in the grid, so there were 27 possible networks to train. The parameter values of the first stage of this tuning process are reported in table 7.4

The final selection of these parameters was determined through a grid search. The tested values for these parameters are provided in table 7.4. With three values for each hyperparameter, this resulted in a set of 27 different PINN models. It's worth noting that all these models were trained with $\omega_u = 2$. This value was selected based on the preliminary tests, which indicated that a slightly increased weight on the data loss should be given. The chosen PINN structure was ultimately determined by evaluating the goodness of fit using the MSE score.

Table 7.4. *Grid of the hyperaparameters used in the first stage of tuning for the PINN.*

Parameters	**Values**		
N_{hidden}	2	5	10
$N_{neurons}$	10	20	30
$N_{collocation}$	50	200	500

At the second stage only the tuning process only the values of the ω_u were tested, in order to investigate which ratio of loss weights can give the best results. Although it is obvious that this parameter affects the results of the first stage tuning, it would be inefficient to include it in the grid search. Thus, the five different values of ω_u that have been tested are $[0.01, 0.1, 1, 2, 5, 10, 100]$. Seven PINNs were retrained (using the already

optimized weights and biases from the first tuning stage) with these values for ω_u. However, it was not possible to compare these models in terms of MSE score, as weighting the terms of the loss function made the comparison infeasible. The estimated partition coefficients of each model, were provided to the PBK model. Then, the PBK model was used to provide predictions at the experimental time points. Consequently, those predictions were compared to the experimental values to evaluate the goodness of fit. The metrics that were selected for this task are the $PBKOF$ described in equation 7.23, the absolute average fold error (AAFE) and the root mean squared error (RMSE). The AAFE and the RMSE metrics are estimated as

$$AAFE = 10^{\frac{1}{N}\sum_{i=1}^{N} |\log\left(\frac{\hat{y}_i}{y_i}\right)|} \tag{7.24}$$

and

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(\hat{y}_i - y_i)^2}{N}} \tag{7.25}$$

respectively, where $\hat{y}_i$ is the predicted value and y_i is the observed value at time i. These metric functions have been implemented in the **PBKtools** library.

Chapter 8

Results and Discussion

8.1 Introduction

In this chapter, the outcomes resulting from implementing the modeling workflows described in chapter 7 will be presented and discussed. We will start by addressing the findings of the identifiability analysis, shedding light on which parameters can be freely estimated from the data during the optimization workflow, without encountering identifiability issues. We will proceed with presenting the results from the optimization workflow. The values of the parameters will be reported in detail, as well as the relevant plots, which illustrate how well the fitted model predicts the experimental data. In the following section, the findings obtained from the PINN workflow will be reported. This section will include results from the grid search hyperparameter tuning. Concluding this chapter, we will compare the two workflows and discuss the advantages of each one.

8.2 Optimization Workflow: Results

The optimization workflow could be considered as a two-step process. The first step is to minimize the value of the objective function, with regard to the free parameters of the model, in order to estimate their optimal value. The second stage, is using these optimal values to perform identifiability analysis to detect any identifiability issues. Of course, this process is a trial and error process, as it is difficult to define a PBK structure from the first try, whose unknown parameters are identifiable. Often, some parameters emerge as non-identifiable, so we need to perform structural modifications to the model, in case of structural non-identifiability, or increase the number of the available data, in case of numerical non-identifiability. At the beginning of this model's development, most of the non-identifiable parameters were structurally non-identifiable. As a result, we needed to adjust the model's parameterization to tackle these issues. In the section, we will present the final parameterization of the model, as well as the resulting plots of the identifiability analysis for this modelling approach.

8.2.1 Estimated Parameters

The implementation of the optimization workflow, indicated that the optimal approach is to assign common values to K_u, CLU_{coef} and Cl_{feces} accross the different PFAS. This

approach comes with certain assumptions. The first assumption is that the rate constant of absorption of PFAS from the lumen contents through the stomach walls to blood remains consistent across substances. The same assumption was made for Cl_{feces}, as there was no information about the amount of PFAS excreted through feces. As for CLU_{coef}, is considered to be constant across all PFAS because it is a correction factor. CLU_{coef} is used to correct the values of the Cl_{urine} rate constants, which are received directly from the model of [50]. We decided on the need for this correction because these parameters expressed the rate constant of urinary elimination with respect to the total body burden of the fish, as it was used in [50], while in our model it expresses the same constant rate with respect to the PFAS concentration in the kidney tissue.

Apart from these three parameters, which are the same across all PFAS, we estimated the set of 7 partition coefficients, exclusively for each substance. The availability of concentration-time profile data for gills, liver, muscle, kidney, skin and carcass allowed the estimation of different partition coefficients for each compartment and PFAS. Additionally, one partition coefficient (for every substance) was estimated for the viscera compartment. Although there was no concentration data for this compartment, it was necessary to have an estimation for this parameter, based on the rest of the experimental data, as this compartment plays a significant role on the absorption and elimination of the PFAS. Considering all these modelling decisions, the total number of estimated parameters was 38 (3 common parameters for all PFAS, plus 7 partition coefficients for every PFAS).

The values of the three common parameters are presented in Table 8.1. Those values are estimated from the simultaneous fit of the model to the experimental data of all PFAS substances.

Table 8.1. *Estimation of common parameters.*

K_u	CLU_{coef}	Cl_{feces}
1.4669	5.7190E-04	1.3065

Table 8.2. *Estimation of partition coefficients for every PFAS.*

Substances	P_{liver}	P_{muscle}	P_{kidney}	P_{skin}	P_{gills}	$P_{carcass}$	$P_{viscera}$
PFOS	1.5685	0.1132	0.4399	0.2716	0.2292	0.1074	3.6991
PFOA	2.0036	0.0369	0.8512	0.3188	0.3428	0.18	0.5637
PFBS	1.7415	0.1387	0.7631	0.2242	0.2772	0.1161	1.0770E-05
PFHxS	1.6979	0.0434	0.3608	0.2935	0.1538	0.0412	8.8461E-06
PFNA	0.8033	0.0649	0.2461	0.2335	0.2204	0.1135	1.2747

The optimized values of the partition coefficients are reported in Table 8.2. Notably, the values of the $P_{viscera}$ across the PFAS substances has a large variability, relatively to the other parameters. Additionally, the values of P_{liver} are larger for every PFAS (except the PFOS where $P_{viscera} > P_{liver}$. This is something that can be justified, as the liver is the organ that has the higher concentration of PFAS.

8.2.2 Identifiability Analysis Results

Following the optimization workflow, the next step after estimating the parameters, was performing identifiability analysis, by implementing the methodology described in chapter 5. The three common parameters were excluded from the analysis. Including these parameters in the identifiability process, would increase significantly the computational cost. Additionally, given that these parameters are common for the five substances it was considered that there is enough information in the experimental data to estimate them. Therefore, the analysis was performed only on the seven partition coefficients of the model. Moreover, only the experimental data and the partition coefficients of PFOS were needed to perform identifiability analysis. That is true because using different set of values of partition coefficients, is equivalent to having 5 different models, which are identical in terms of structure and the only differences between them are the values of the parameters. So, given that the amount of data for each PFAS is the same, we can perform identifiability analysis only for the PBK and dataset for PFOS and then expand the results to the other models too.

Table 8.3. *Identifiability analysis results and likelihood-based confidence intervals* $\sigma^{\pm}$ *derived from the profile likelihood method.*

Parameters	Non-Identifiability Issue	Optimal Value	Lower Bound	Upper Bound
P_{liver}	Identifiable	1.5685	0.00312	558.183
P_{muscle}	Identifiable	0.1132	0.00028	31.506
P_{kidney}	Identifiable	0.4399	0.0013	25.727
P_{skin}	Identifiable	0.2716	0.00057	162.598
P_{gills}	Identifiable	0.2292	0.00048	159.814
$P_{carcass}$	Identifiable	0.1074	0.00023	57.641
$P_{viscera}$	Practical	3.6991	0	412.162

The conclusions from the identifiability analysis were derived from the profile likelihood plots of the parameters. The plots for all partition coefficients are shown in figure 8.1. Additionally, table 8.3 reports the upper and lower bounds of the parameters, determined using the profile likelihood method, and highlights which parameters are non-identifiable. These plots depict the estimated likelihood profiles, as detailed in chapter 5.

By observing the plots it is obvious that all parameters, except $P_{viscera}$, are identifiable. Meanwhile, the profile likelihood of $P_{viscera}$ shows different curvature than the other parameters. Notably, the profile likelihood exceeds the defined threshold only from the right side of optimal value, while the left side of the profile reaches a plateau, for any value of $P_{viscera}$. The curvature of this profile indicates that $P_{viscera}$ is a practical non-identifiable parameter. The fact that the experimental data used to fit the model contained no data for concentration-time measurements inside the organs modeled through the viscera compartment, comes in agreement with the fact that this parameter is not totally identifiable and could be somehow predicted prior the optimization process. $P_{viscera}$ being practical non-identifiable means that feeding the optimization workflow with some exper-

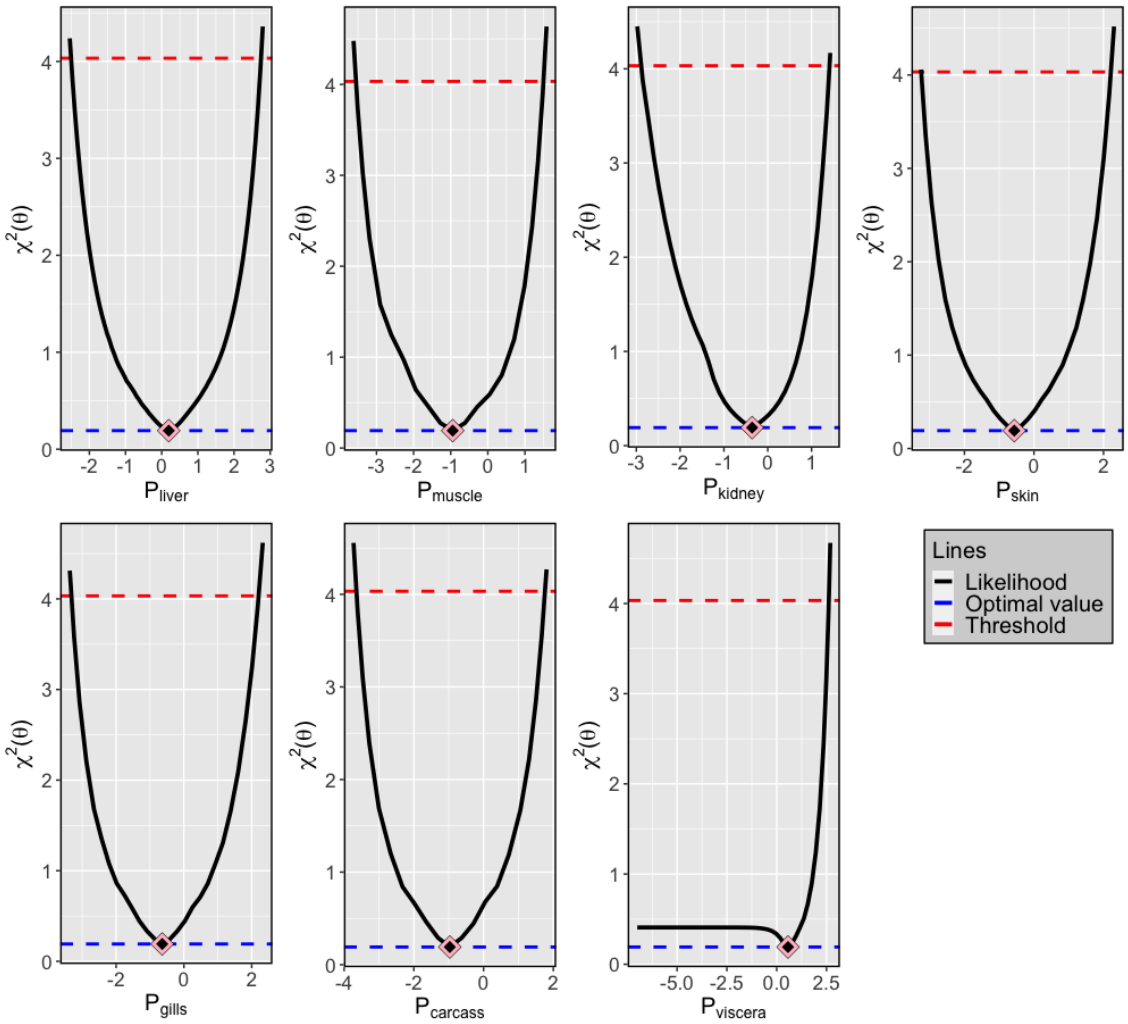

Figure 8.1. *Identifiability analysis for the Partition coefficients of the model. The plots show with black continuous lines the estimated profile likelihood of the parameters. The horizontal blue line represents the minimized value of the objective function $\chi^2(\theta)$. The red dashed line represents the user defined threshold, which defines if any parameter is identifiable or not. The threshold is calculated as $Threshold = \chi^2(\hat{\theta}) + \Delta_\alpha$, where Δ_α is the 95^{th} quantile of the χ^2 distribution with degree of freedom $df = 1$. The rhombus symbol represents the optimal value of each parameter, after the optimization of the $\chi^2(\theta)$. Notably, for all parameters the black line exceeds the threshold, so they can be considered as identifiable. However, the black line of $P_{viscera}$ exceeds the threshold only from the right side, while $\chi^2(\theta)$ takes a constant (under the threshold) value while $P_{viscera}$ extends to values lower than the optimal. Consequently, $P_{viscera}$ is practical non-identifiable parameter. The values of the x-axis are in log scale.*

imental data point of concentration of PFAS measured in the visceral organs, would turn this parameter into totally identifiable. It was clear that obtaining additional experimental data was not feasible. However, $P_{viscera}$ was retained in the set of free parameters. The primary reason was the inability to find a value for this parameter in the literature, which would have allowed fixing the parameter to a constant value. Furthermore, this parameter

is significant, as the dynamics of the viscera directly influence the dynamics of the other compartments. Therefore, it was essential to keep this parameter adjustable to ensure a better fit of the model.

8.2.3 Concentration - Time Profiles

After having completed the identifiability analysis, the next step was to evaluate the goodness of fit of the model to the training experimental data. In figure 8.2, each plot presents the predictions of the model versus the experimental measurements for all organs and tissues. To produce the plots, the optimal values reported in tables 8.1 and 8.2 were used in the PBK model. Then, the ODEs of the model (provided with the appropriate initial conditions) were solved using an ODE solver library of R and the results of the solution were plotted along with the experimental data of each substance.

Observing the concentration profiles across the different organs and across the different PFAS substances, the model is aligned with the experimental data, with the curves consistently mirroring the data trends. However, a notable disagreement emerges between the model's predictions for PFBS and its corresponding dataset. Notably, the model cannot predict at a satisfying level the data especially for the blood compartment, during the feeding phase (left of the vertical black line). Additionally, the model struggles to accurately predict the kidney data in the case of PFNA, resulting in an underestimation of the measurements. Nevertheless, even for those substances, the model captures the underlying data dynamics, with any discrepancy of the model predictions from the data points remaining at an acceptable level.

It remains uncertain whether the distinctions observed in the PFBS dynamics, relative to the other PFAS, stem from unaccounted biological mechanisms in the modelling framework or from potential measurement errors. For instance, the third measurement of blood for PFBS records an unexpected decline, despite the fact that the fish were being fed daily up to that point. Therefore, we would expect the third measurement of blood to be the highest, or at least at the same level with the first two measurements. Instead, the data show a sharp decrease of the concentration after the second measurement, posing intriguing questions regarding potential underlying factors.

Concluding, model predictions of the experimental data are satisfying, considering that there are three important parameters with common values for all PFAS. For instance, the K_u, that expresses the rate constant of PFAS transportation from the stomach to the blood circulation, might not be the same for all PFAS, as different substances may bind with different rates at specific proteins, responsible for the absorption of PFAS. It is sensible that using a fixed value to model the different absorption kinetics may limit the predictive ability of the model. For instance, in preliminary tests, where a different K_u was tested for each substance, the model could predict the kidney data more efficiently. However, in this modeling approach the identifiability analysis indicated that using different K_u values for each substance, turned the K_u parameter into non-identifiable, thus the estimation of these values was characterized with increased uncertainty.

The same could be said for the Cl_{feces} parameter. This parameter was estimated

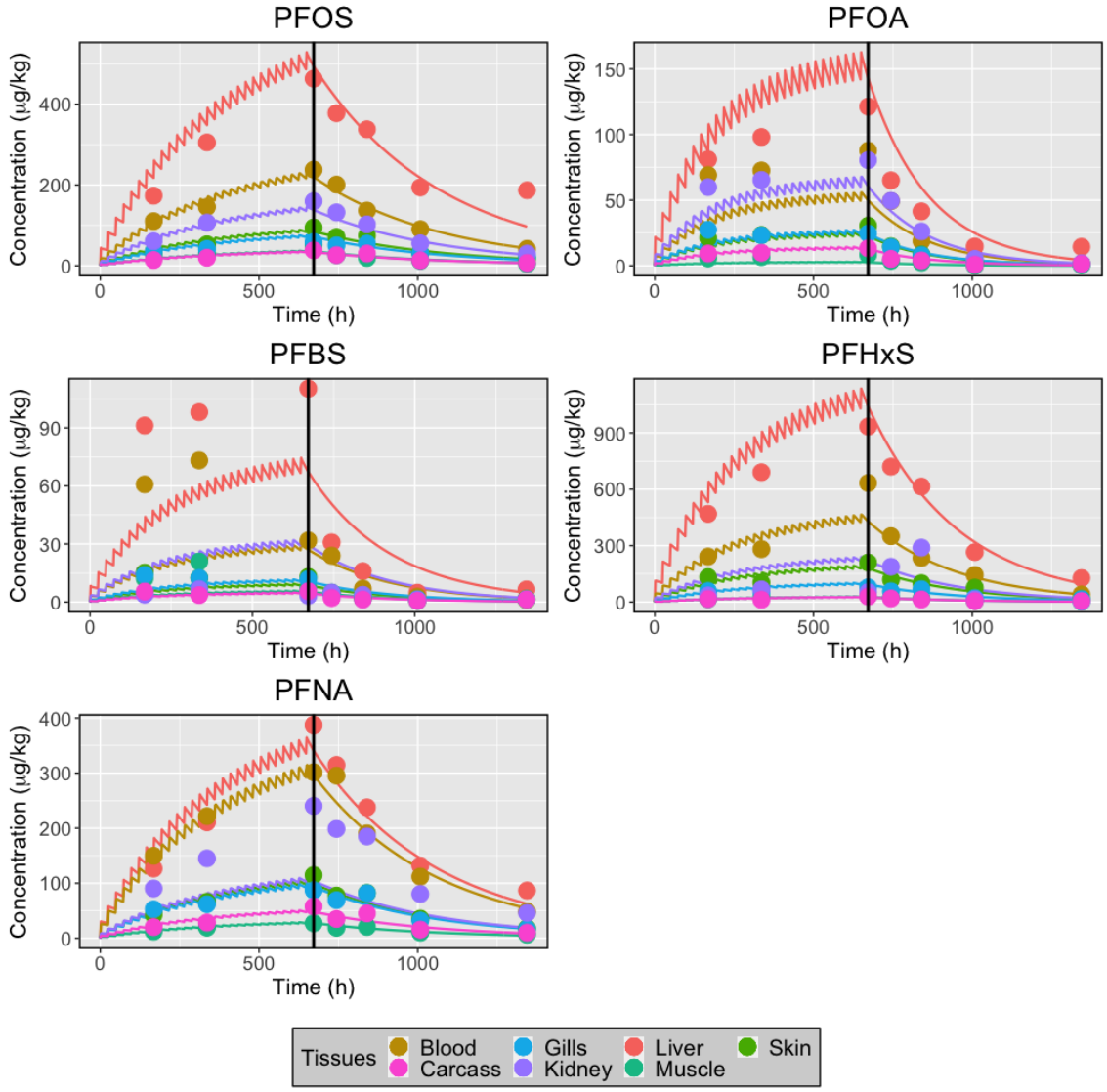

Figure 8.2. *Predictions of the concentration of PFAS in each compartment over time. The lines correspond to the predicted concentrations, while the points are used for the experimental measurements. Each plot features a vertical black line marking the time point when the fish ceased being fed with PFAS-spiked food. Beyond this line, the fish were not exposed to any amount of PFAS, indicating the depuration period.*

without having available experimental data for the excreted PFAS amounts through the feces. In case those data were available, the estimation of the Cl_{feces} would have been more accurate and they would improve the predictions of the model. However, this is an advantage of the PBK models, as they enable the estimation of many unknown kinetic parameters using a small number of sparse experimental data, exploiting the information that stems from the physiology of the organism. Just like K_u, a different value for Cl_{feces} was estimated for each substance. Although this approach improved the concentration plots, it created identifiability issues with this parameter. Therefore, using an increased number of parameters in a PBK model is an easy method to improve its predictive ability, but is not a good modelling practice, as it creates identifiability issues and the estimated

parameters are characterized by high uncertainty. At this point, the importance of a tool for easily conducting identifiability tests during the model development process is evident, as it helps avoid over-parameterization of the model.

8.3 PINN Workflow: Results

Utilizing PINNs to estimate the parameters of the differential equations in a dynamic system offers numerous advantages. Considering this PBK model as a case study, it is evident that even simple feedforward networks with a reasonably small number of layers networks are sufficient to perform this task. During the simulation phase, it became clear that this stage didn't present significant challenges, and it was possible to achieve satisfactory results within reasonable simulation durations and test counts. Therefore, following the tuning two-stages strategy presented in section 7.7, a PINN network was trained to estimate the parameters of the model.

8.3.1 Hyperparameters Tuning

The first stage involves selecting the number of hidden layers (N_{hidden}), the number of neurons per layer ($N_{neurons}$) and the number of collocation points ($N_{collocation}$). After testing all possible parameter combinations, the complete set of different PINNs was trained. The training results are documented in table 8.4, expressed in terms of MSE. All the models reported in table 8.4 were trained for 400,000 iterations, with learning rate $lr = 0.001$, using the $tanh$ as activation function and the algorithm "Adam" as optimizer to update the weights and the biases.

From the results of table 7.4, it turns out that that using five hidden layers is optimal. The networks with two hidden layers did not manage to achieve a low loss value in most cases. That leads to the conclusion that two hidden layers are not sufficient to model the kinetics governed by the differential equations of the PBK model. On the other hand, neither the PINNs with ten layers achieved low loss values. Considering that all the networks were trained for 400,000 epochs, it is possible that the networks with ten hidden layers needed more iterations to be trained. Therefore, using five hidden layers seems to be the optimal selection for the FFN, as these networks consistently scored low loss values.

Considering the results of table 8.1, it is obvious that trying to train a PINN with ten or more hidden layers for a larger number of iterations is not necessary, as the results given from the networks are already satisfactory in terms of goodness of fit with respect to the experimental data. Going deeper in the structures with five hidden layers, we observe that the model with 30 neurons per layers and 500 collocation points achieved significantly lower losses than the rest of the networks with five layers. Therefore, model 18 was selected to proceed to the next level of tuning the weights of the loss function.

8.3.2 Tuning the Weights of the Loss Function

Model 18 from the previous stage was used in this stage to tune an additional parameter, the weight value ω_u of the loss, which refers to the experimental data in the loss function

Table 8.4. *Loss values for the PINNs structures tested during the first stage of the hyperparameters tuning.*

Model id	N_{hidden}	$N_{neurons}$	$N_{collocation}$	Loss
1			50	6.91
2		10	200	4.81
3			500	60.6
4			50	4.71
5	2	20	200	60.8
6			500	60.3
7			50	60.8
8		30	200	60.4
9			500	4.08
10			50	4.2
11		10	200	4.82
12			500	4.78
13			50	5
14	5	20	200	4.67
15			500	5
16			50	4.74
17		30	200	4.35
18			500	2.96
19			50	61.4
20		10	200	6.71
21			500	19
22			50	8.66
23	10	20	200	51.5
24			500	60.3
25			50	7.74
26		30	200	60.8
27			500	60.1

of the PINN. The effect of the relationship between the weights of the two terms in the loss function was considered crucial to assess whether different weights are necessary. The parameter ω_u is of utmost importance, as it influences the training of the model, directing the model's focus either towards the experimental data or the information derived from the differential equations. Using a high value for ω_u might cause the PINN to overfit, neglecting the valuable information provided by the equations. Conversely, a very low ω_u value might result in the PINN not being adequately trained to predict the experimental data. Clearly, this parameter is essential in the hyperparameter calibration process.

As a result, model 18 was used to continue the training from the point it stopped in the previous stage. So the already updated values of weights and partition coefficients were loaded on a structure identical to this of model 18. As we want to examine the effect of the ω_u/ω_f on the training of the PINN, there was no reason to change both parameters at each test. Thus, at the following tests, the value of ω_f was fixed at 1, in the first stage of tuning. Consequently, the model was retrained for 200,000 iterations using 7 different values of ω_u. In these tests we included both cases where the $\omega_u > \omega_f$ and the opposite. After training

the models, the estimated values of the partition coefficients from each PINN were provided to the PBK model to compute the model's predictions and evaluate the goodness of fit, using multiple metric functions. Therefore, the metrics used to evaluate the models are the PBKOF, AAFE, RMSE and R^2. Obviously, lower values of PBKOF, AAFE and RMSE indicate better fit on the experimental data, while the opposite is true for R^2. The results stem from each model are reported in table 8.5.

Table 8.5. *Scores of the PINNs trained with different values of ω_u. The model was retrained for 200,000 iterations, using the already updated network weights from the previous hyperparameters tuning process. In all tests reported here $\omega_f = 1$.*

ω_u	**PBKOF**	**AAFE**	**RMSE**	$\mathbf{R^2}$
0.01	0.915	1.635	54.184	0.809
0.1	0.965	1.657	55.298	0.805
1.0	0.940	1.638	51.391	0.810
2.0	1.054	1.695	54.229	0.794
5.0	0.977	1.661	55.563	0.806
10.0	0.933	1.690	65.929	0.778
100.0	1.580	1.983	69.073	0.710

Observing the values of the metrics reported in table 8.5 it is obvious that the PINN trained with $\omega_u = 100.0$ achieved the worst scores in all metrics. That means that the ω_u forced the PINN to strongly ignore the information provided by the PBK model and operated more like a simple feedforward network. Generally, increasing the value of the ω_u/ω_f ratio, leads the PINN to increasingly ignore the information provided by the differential equations of the dynamic system.

However, the rest of the models have similar scores. Going deeper into the results reported in table 8.5, the predictions of the model whose parameters come from the PINN with $\omega_u = 0.01$ have achieved the best score in terms PBKOF and AAFE, while it is the second best score in terms of RMSE and R^2. Giving more focus on the value of the PBKOF is a reasonable choice. The PBKOF is a custom metric function, developed especially for the validation of PBK models. So considering that the model of ω_u provided a parametric set that led to the best PBKOF score, we should consider this model as the best. Additionally, even in terms of RMSE and R^2, this model has achieved the second best scores and the differences are not that significant, compared with the corresponding best values (achieved by the model with $\omega_u = 1.0$). After considering all these details, the conclusion is that the optimal value of $\omega_u = 0.01$. Finally, a common observation in the generated plots in figure 8.3 is that all models could estimate the partition coefficients sufficiently well. However, it's essential to utilize the metric functions to draw conclusions, as the plots alone do not highlight the differences between the models.

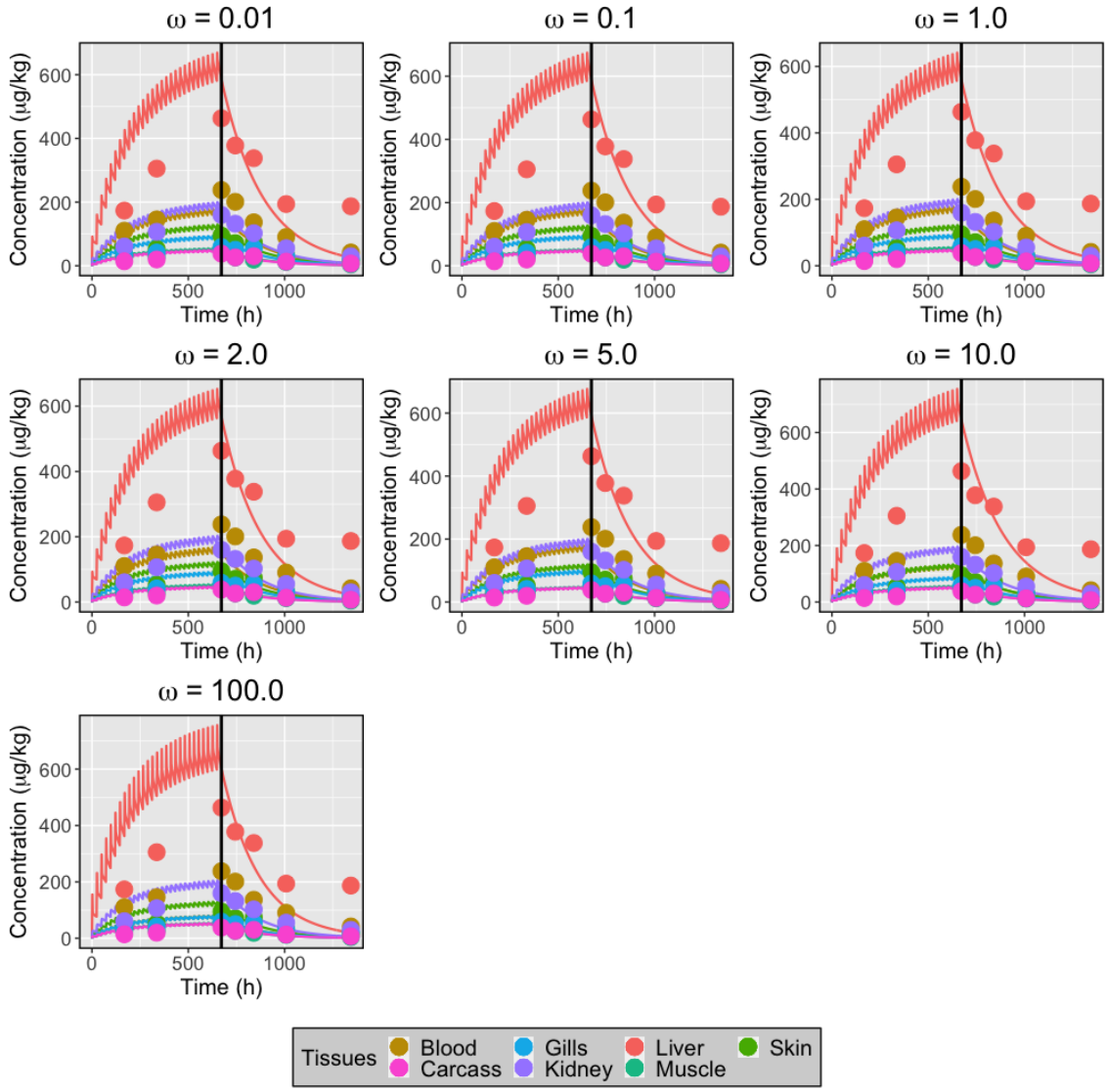

Figure 8.3. *Predicted concentrations of the PBK model and the experimental data. Each plot refers to the predictions produced using the partition coefficients estimated by the PINN that is trained using the corresponding value of ω_u. The lines correspond to the predicted concentrations, while points indicate experimental measurements. Each plot features a vertical black line marking the time point when the fish ceased being fed with PFAS-spiked food. Beyond this line, the fish were not exposed to any PFAS, marking the depuration period.*

Chapter 9

Conclusions

The development of PBK models has proven to be a useful tool in risk assessment workflows. This thesis focused on the development of a PBK model and the estimation of its unknown parameters using two different workflows. The first one estimated the parameters by minimizing the value of an objective function. The second workflow implemented a PINN to estimate the parameters.

Despite the small size of the dataset that was used to estimate the unknown parameters of the PBK model, it was feasible to achieve satisfying predictions of the experimental data. However, the quantity and the quality of the available experimental data both influenced the structural development of the model. For instance, the dataset included experimental measurements for 7 organs and tissues of the fish, but it did not contain any measurements of the excreted PFAS. This limitation restricted the number of unknown parameters that could be estimated through the data. Specifically, the lack of urine measurements resulted in fixing the Cl_{urine} parameters to a constant value, found in literature, instead of estimating the precise value of this parameter for each substance. Therefore, the development of a PBK model involved maintaining a balance between achieving a good fit and avoiding overparameterization of the model.

Avoiding model overparameterization was specifically addressed by building a workflow that performs identifiability analysis on the model's parameters based on the profile likelihood method. The main benefits of this method, were simplicity of its implementation and a reasonable computational cost. Identifiability analysis provided useful conclusions, not only regarding which parameters were non-identifiable but also whether they were structurally or practically non-identifiable. This classification indicated the steps that should be followed to overcome the identifiability issues. For instance, when a parameter was structurally non-identifiable, it was obvious that structural modifications to the model were necessary to either eliminate the need for estimating this parameter or to set it to a constant value. This approach led to fixing the Cl_{urine} to a constant value, as was previously mentioned. Moreover, it indicated that using a different K_u for each substance raised identifiability issues, since the experimental data were insufficient for estimating this parameter. Consequently, we estimated a common value for the K_u parameter which falls within the reasonable range defined by the data.

Regarding the optimization workflow, the selection of $PBKOF$ as objective function ensured that the concentration predictions matched the experimental data for all organs

equally well, even when measurements varied in scale. Other metric functions, like the root mean squared error did not provide good results, as this function focused more on compartments with higher measured concentrations, as liver and blood. So this process, indicated that using the PBKOF can significantly improve the optimization workflow, yielding better predictions for all observable variables.

However, the optimization workflow exhibited some weaknesses. Firstly, in some optimization problems the numerical solution of the differential equations can be a computationally costly task. Even though the system of the differential equations used in this model is relatively simple (as they are first order and linear) the computational cost increases significantly with the number of times that PFAS are administered (through the food). Notably, the current PBK modeled the daily dietary uptake of PFAS. Therefore, every 24 hours, a specific state variable (M_{lumen_1}) increased by a certain value. Imposing such changes on a state variable repeatedly during the integration time frame impacts the computational cost, and slows down the solution of the ODEs. Particularly, when the solution of the ODEs is incorporated into an optimization workflow, like the one developed in this thesis, and the ODEs are solved in each iteration, it is obvious that this makes the whole process much slower.

Another limitation of the optimization algorithm is that it performs satisfactorily only when estimating a reasonably small number of parameters (such five to ten). However, the workflow for the PBK model that was developed in this thesis, required estimating the three parameters considered common for all PFAS, and a set of seven partition coefficients for each of the five examined substances. This amounted to a total of 38 parameters to estimate. Together the aforementioned limitations resulted in long computational times for minimizing the $PBKOF$ value, which made difficult the experimentation with different model structures, and addressing possible identifiability issues.

The limitations of the optimization workflow led us to explore PINNs as an alternative approach for estimating the model's unknown parameters. We implemented the PINNs workflow using the DeepXDE module in Python. This library supports various backends, including TensorFlow and PyTorch. The primary advantage of this module is its ability to streamline the coding process for the entire workflow. With comprehensive documentation, it provides all the essential functions and guidance needed for this specific task.

While training the PINN model, we conducted hyperparameter tuning to identify the optimal number of hidden layers, neurons, and collocation points. Our grid search pinpointed the best parameter combination, with 5 hidden layers emerging as the most suitable. Networks with only 2 hidden layers failed to train properly, resulting in high loss values. On the other hand, networks with 10 hidden layers faced challenges in training effectively even after numerous iterations. Since networks with five hidden layers demonstrated promising outcomes, we did not investigate deeper networks further. As for the number of neurons per layer, we explored three different configurations, but the training loss did not display a consistent pattern, making it challenging to derive definitive conclusions.

The final parameter we examined during the grid search was the number of collocation points. This parameter is intrinsically linked to the insights obtained from the PBK

model's governing equations. This parameter is important because it defined the sampled time points in the time domain where one of the loss function's terms is assessed. This specific term aims to minimize the difference between the derivative values estimated via automatic differentiation and those derived directly from the ODEs. It's imperative to choose a value for this parameter that ensures comprehensive representation across the time domain. Based on established literature, we selected the number of 500 collocation points as optimal for training the neural network. Notably, this is the sole parameter during the initial tuning phase directly tied to the information from the PBK model's differential equations. Concluding this hyperparameter tuning stage, the best PINN structure was found to comprise 5 hidden layers, each with 30 neurons, and utilized 500 collocation points for training. During this phase, the ratio ω_u/ω_f was set at 2.0.

The second stage of tuning explored the ω_u/ω_f ratio, highlighting its significance and the sensitivity of the PINN's predictions to this parameter. We trained multiple PINN models using various values of ω_u, while keeping the ω_f fixed to a constant value ($\omega_f = 1$). The PBKOF metric was used to validate the models and the one with the lowest PBKOF value was selected. The optimal value $\omega_u = 0.01$, indicated that during the PINN training, more emphasis is given on the information provided by the dynamics of the model. This helps the PINN to avoid overfitting to experimental data. In contrast, larger values of the ω_u/ω_f values cause the PINN to overlook the differential equations, making it behave more like a basic feedforward network trained solely on the experimental data.

In general, the PINN implementation for solving inverse problems provides an alternative approach with certain benefits over the optimization workflow. The modeling of daily input of PFAS as an increment to a state variable was straightforward via a simple function and did not affect the computational burden of the simulations. Thus, PINNs effectively tackle an important problem that stems from the frequent forced changes on state variables.

The implementation of the PINN successfully produced a model that closely matched the experimental data while adhering to the differential equations of the PBK model. As such, the PINN approach demonstrated its capability to achieve a satisfactory goodness of fit for PBK models while keeping computational costs reasonable. However, it's important to highlight that the optimization workflow produced a superior goodness of fit compared to the PINN method. A significant factor contributing to this difference was the use of mean squared error as a loss function in PINNs. In contrast, the optimization workflow employed the PBKOF, which has been empirically shown to be a more effective objective function for training PBK models.

The main limitation of the PINN approach is the absence of an identifiability analysis process. This omission can result in overparameterization of the models, with the estimated parameter values lacking physical significance. Therefore, when choosing the PINN workflow over the optimization method, it's essential to supplement it with an identifiability analysis tool. However, the PINN model might be a more efficient and suitable method for tasks where identifiability analysis isn't required. For example, when using data similar to other PFAS not covered in this thesis, the PINN workflow can be used to re-estimate the values of some kinetics parameters. Another example is interpolating the

model to different species. In this case, the parameters of the model are updated using new experimental data, that refer to the species of the interest.

Taking all these into consideration, this diploma thesis could be extended to multiple directions. First of all, regarding the structure of the PBK model, it is suggested to examine the inclusion of a gills compartment as an additional elimination pathway. It is possible that a small amount of the PFAS is eliminated through the exhalation of the fish. However, due to inadequacy of experimental data related to this process, in this work it was considered that this amount is negligible. Moreover, the model could be extended to take into account the simultaneous uptake of PFAS both from consumed food and the water, through the inhalation process. Specifically, for this task the proposed approach is to search for experimental studies which expose the fish to PFAS in both ways. One more suggestion, is to re-estimate the parameters of the model which are related to the fecal and urinary elimination if more experimental data become available about these two elimination processes of PFAS. Having detailed concentration-time data of the PFAS into the excreta of fish, would lead to more accurate estimation of the parameters Cl_{urine} and Cl_{feces}.

In relation to the PINN approach, a logical next step following this diploma thesis would be to adapt the workflow to estimate the common parameters K_u, CLU_{coef} and Cl_{feces}, as well as all the partition coefficients for the five PFAS substances. If PINNs can efficiently estimate such a comprehensive set of parameters and produce satisfactory results, they may replace traditional optimization workflow for PBK development. Additionally, it's highly recommended to explore alternative loss functions such as $PBKOF$ which can consider the varying magnitudes of the model's output variables.

Bibliography

[1] J. W. Nichols, J. M. McKim, G. J. Lien, A. D. Hoffman, S. L. Bertelsen and C. M. Elonen. *A Physiologically Based Toxicokinetic Model for Dermal Absorption of Organic Chemicals by Fish*. *Fundamental and Applied Toxicology*, 31:229–242, 1996.

[2] Pavan Vadapalli. *Biological Neural Network: Importance, Components & Comparison*. https://www.upgrad.com/blog/biological-neural-network, 2021.

[3] Emiliano Panieri, Katarina Baralic, Danijela Djukic-Cosic, Aleksandra Buha Djordjevic and Luciano Saso. *PFAS Molecules: A Major Concern for the Human Health and the Environment*. *Toxics*, 10:44, 2022.

[4] Suzanne E. Fenton, Alan Ducatman, Alan Boobis, Jamie C. DeWitt, Christopher Lau, Carla Ng, James S. Smith and Stephen M. Roberts. *Per- and Polyfluoroalkyl Substance Toxicity and Human Health Review: Current State of Knowledge and Strategies for Informing Future Research*. *Environmental Toxicology and Chemistry*, 40:606–630, 2020.

[5] John W. Stanifer, Heather M. Stapleton, Tomokazu Souma, Ashley Wittmer, Xinlu Zhao and L. Ebony Boulware. *Perfluorinated Chemicals as Emerging Environmental Threats to Kidney Health: A Scoping Review*. *Clinical Journal of the American Society of Nephrology*, 13:1479–1492, 2018.

[6] Warren S. McCulloch and Walter Pitts. *A Logical Calculus of the Ideas Immanent in Nervous Activity*. *The Bulletin of Mathematical Biophysics*, 5:115–133, 1943.

[7] Frank Rosenblatt. *The perceptron: a Probabilistic Model for Information Storage and Organization in the brain.* *Psychological Review*, 65:386–408, 1958.

[8] Daniel A Roberts, Sho Yaida and Boris Hanin. *The Principles of Deep Learning Theory : an Effective Theory Approach to Understanding Neural Networks*. Cambridge University Press, 1η έκδοση, 2022.

[9] Ian Goodfellow, Yoshua Bengio and Aaron Courville. *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org.

[10] David E. Rumelhart, Geoffrey E. Hinton and Ronald J. Williams. *Learning representations by back-propagating errors*. *Nature*, 323:533–536, 1986.

[11] Maziar Raissi, Paris Perdikaris and George Em Karniadakis. *Physics Informed Deep Learning (Part I): Data-driven Solutions of Nonlinear Partial Differential Equations*. *ArXiv (Cornell University)*, 2017.

[12] M. Raissi, P. Perdikaris and G.E. Karniadakis. *Physics-informed Neural networks: a Deep Learning Framework for Solving Forward and Inverse Problems Involving Nonlinear Partial Differential Equations*. *Journal of Computational Physics*, 378:686–707, 2019.

[13] Maziar Raissi, Paris Perdikaris and George Karniadakis. *Physics Informed Deep Learning (Part II): Data-driven Discovery of Nonlinear Partial Differential Equations*. *ArXiv (Cornell University)*, 2017.

[14] Weiqi Ji, Weilun Qiu, Zhiyu Shi, Shaowu Pan and Sili Deng. *Stiff-PINN: Physics-Informed Neural Network for Stiff Chemical Kinetics*. *The Journal of Physical Chemistry*, 125:8098–8106, 2021.

[15] Viktor Grimm, Alexander Heinlein, Axel Klawonn, Martin Lanser and Janine Weber. *Estimating the time-dependent Contact Rate of SIR and SEIR Models in Mathematical Epidemiology Using physics-informed Neural Networks*. *ETNA - Electronic Transactions on Numerical Analysis*, 56:1–27, 2021.

[16] Kaan Sel, Amirmohammad Mohammadi, Roderic I. Pettigrew and Roozbeh Jafari. *Physics-informed Neural Networks for Modeling Physiological Time Series for Cuffless Blood Pressure Estimation*. *Npj Digital Medicine*, 6:1–15, 2023.

[17] Rudolf L.M.van Herten, Amedeo Chiribiri, Marcel Breeuwer, Mitko Veta and Cian M. Scannell. *Physics-informed Neural Networks for Myocardial Perfusion MRI Quantification*. *Medical Image Analysis*, 78:102399, 2022.

[18] Kanupriya Goswami, Arti Sharma, Madhu Pruthi and Richa Gupta. *Study of Drug Assimilation in Human System Using Physics Informed Neural Networks*. *International Journal of Information Technology*, 2022.

[19] *FDA - Division of Pharmacometrics*. https://www.fda.gov/about-fda/center-drug-evaluation-and-research-cder/division-pharmacometrics. Access date: 21-08-2023.

[20] Johan Gabrielsson and Daniel Weiner. *Non-compartmental analysis*. *Methods in Molecular Biology (Clifton, N.J.)*, 929:377–389, 2012.

[21] Torsten Teorell. *STUDIES ON THE DIFFUSION EFFECT UPON IONIC DISTRIBUTION*. *Journal of General Physiology*, 21:107–122, 1937.

[22] Wells Utembe, Harvey Clewell, Natasha Sanabria, Philip Doganis and Mary Gulumian. *Current Approaches and Techniques in Physiologically Based Pharmacokinetic (PBPK) Modelling of Nanomaterials*. *Nanomaterials*, 10:1267, 2020.

[23] Francesc Fàbrega, Vikas Kumar, Marta Schuhmacher, José L Domingo and Martí Nadal. *PBPK modeling for PFOS and PFOA: Validation with human experimental data*. *Toxicology Letters*, 230:244–251, 2014.

[24] Ronald P. Brown, Michael D. Delp, Stan L. Lindstedt, Lorenz R. Rhomberg and Robert P. Beliles. *Physiological Parameter Values for Physiologically Based Pharmacokinetic Models*. *Toxicology and Industrial Health*, 13:407–484, 1997.

[25] M. G. Barron, B. D. Tarr and W. L. Hayton. *Temperature-dependence of cardiac output and regional blood flow in rainbow trout, Salmo gairdneri Richardson*. *Journal of Fish Biology*, 31:735–744, 1987.

[26] W.H. Gingerich, R.A. Pityer and J. J. Rach. *Whole body and tissue blood volumes of two strains of rainbow trout (Oncorhynchus mykiss)*. *Comparative Biochemistry and Physiology Part A: Physiology*, 97:615–620, 1990.

[27] Joseph DiStefano III. *Dynamic Systems Biology Modeling and Simulation*, τόμος 8. Academic Press, 1η έκδοση, 2015.

[28] Lennart Ljung and Torkel Glad. *On global identifiability for arbitrary model parametrizations*. *Automatica*, 30:265–276, 1994.

[29] A. Raue, C. Kreutz, T. Maiwald, J. Bachmann, M. Schilling, U. Klingmüller and J. Timmer. *Structural and practical identifiability analysis of partially observed dynamical models by exploiting the profile likelihood*. *Bioinformatics (Oxford, England)*, 25:1923–1929, 2009.

[30] *OECD - Reconciling Terminology of the Universe of Per-and Polyfluoroalkyl Substances: Recommendations and Practical Guidance*. https://www.oecd.org/chemicalsafety/portal-perfluorinated-chemicals/terminology-per-and-polyfluoroalkyl-substances.pdf. Access date: 2023-08-25.

[31] Robert C Buck, James Franklin, Urs Berger, Jason M Conder, Ian T Cousins, Pimde Voogt, Allan Astrup Jensen, Kurunthachalam Kannan, Scott A Mabury and Stefan PJvan Leeuwen. *Perfluoroalkyl and polyfluoroalkyl substances in the environment: Terminology, classification, and origins*. *Integrated Environmental Assessment and Management*, 7:513–541, 2011.

[32] Juliane Glüge, Martin Scheringer, Ian T. Cousins, Jamie C. DeWitt, Gretta Goldenman, Dorte Herzke, Rainer Lohmann, Carla A. Ng, Xenia Trier and Zhanyun Wang. *An overview of the uses of per- and polyfluoroalkyl substances (PFAS)*. *Environmental Science: Processes & Impacts*, 22:2345–2373, 2020.

[33] W. S. GUY, D. R. TAVES and W. S. BREY. *Organic Fluorocompounds in Human Plasma: Prevalence and Characterization*. *ACS Symposium Series*, 28:117–134, 1976.

[34] Kristen J. Hansen, Lisa A. Clemen, Mark E. Ellefson and Harold O. Johnson. *Compound-Specific, Quantitative Characterization of Organic Fluorochemicals in Biological Matrices*. *Environmental Science & Technology*, 35:766–770, 2001.

[35] H. T. Wan, Y. G. Zhao, X. Wei, K. Y. Hui, J. P. Giesy and Chris K. C. Wong. *PFOS-induced hepatic steatosis, the mechanistic actions on β-oxidation and lipid transport*. *Biochimica et Biophysica Acta (BBA) - General Subjects*, 1820:1092–1101, 2012.

[36] Jae Mun Lee. *Fatty Infiltration of Liver*. *Computed Tomography*, σελίδες 116–119, 1996.

[37] Kan Li, Jie Sun, Jingping Yang, Stephen M Roberts, Xu Xiang Zhang, Xinyi Cui, Si Wei and Lena Q Ma. *Molecular Mechanisms of Perfluorooctanoate-Induced Hepatocyte Apoptosis in Mice Using Proteomic Techniques*. *Environmental Science & Technology*, 51:11380–11389, 2017.

[38] Anne E Loccisano, Jerry L Campbell, John L Butenhoff, Melvin E Andersen and Harvey J Clewell. *Evaluation of placental and lactational pharmacokinetics of PFOA and PFOS in the pregnant, lactating, fetal and neonatal rat using a physiologically based pharmacokinetic model*. *Reproductive Toxicology*, 33:468–490, 2012.

[39] Carla A Ng and Konrad Hungerbühler. *Bioconcentration of Perfluorinated Alkyl Acids: How Important Is Specific Binding?* *Environmental Science & Technology*, 47:7214–7223, 2013.

[40] Alice Vidal, Marc Babut, Jeanne Garric and Rémy Beaudouin. *Elucidating the fate of perfluorooctanoate sulfonate using a rainbow trout (Oncorhynchus mykiss) physiologically-based toxicokinetic model*. *Science of The Total Environment*, σελίδα 1297–1309, 2019.

[41] Sandy Falk, Klaus Failing, Sebastian Georgii, Hubertus Brunn and Thorsten Stahl. *Tissue specific uptake and elimination of perfluoroalkyl acids (PFAAs) in adult rainbow trout (Oncorhynchus mykiss) after dietary exposure*. *Chemosphere*, 129:150–156, 2015.

[42] *Graph Grabber 2.0.2 | Quintessa Limited | Scientific and Mathematical Consultancy*. https://www.quintessa.org/software/downloads-and-demos/graph-grabber-2.0.2. 2023-08-31.

[43] Chris M Wood and Graham Shelton. *Cardiovascular Dynamics and Adrenergic Responses of the Rainbow Trout In Vivo*. *Journal of Experimental Biology*, 87:247–270, 1980.

[44] Audrey Grech, Cleo Tebby, Céline Brochot, Frédéric Y. Bois, Anne Bado-Nilles, Jean Lou Dorne, Nadia Quignot and Rémy Beaudouin. *Generic physiologically-based toxicokinetic modelling for fish: Integration of environmental factors and species variability*. *Science of The Total Environment*, 651:516–531, 2019.

[45] M. Grosell, M. J. O'Donnell and C. M. Wood. *Hepatic versus gallbladder bile composition: in vivo transport physiology of the gallbladder in rainbow trout*. *American Journal of Physiology-Regulatory, Integrative and Comparative Physiology*, 278:R1674–R1684, 2000.

[46] B. James Curtis and Chris M Wood. *The Function of the Urinary Bladder In Vivo in the Freshwater Rainbow Trout*. *The Journal of Experimental Biology*, 155:567–583, 1991.

[47] E. D. Stevens. *The effect of exercise on the distribution of blood to various organs in rainbow trout*. *Comparative Biochemistry and Physiology*, 25:615–625, 1968.

[48] Richard W. Brill, Katherine L. Cousins, David R. Jones, Peter G. Bushnell and John F. Steffensen. *Blood Volume, Plasma Volume and Circulation Time in a High-Energy-Demand Teleost, the Yellowfin Tuna (Thunnus Albacares)*. *Journal of Experimental Biology*, 201:647–654, 1998.

[49] Ina Goeritz, Sandy Falk, Thorsten Stahl, Christoph Schäfers and Christian Schlechtriem. *Biomagnification and tissue distribution of perfluoroalkyl substances (PFASs) in market-size rainbow trout (Oncorhynchus mykiss)*. *Environmental Toxicology and Chemistry*, 32:2078–2088, 2013.

[50] Jennifer M. Sun, Barry C. Kelly, Frank A. P. C. Gobas and Elsie M. Sunderland. *A food web bioaccumulation model for the accumulation of per- and polyfluoroalkyl substances (PFAS) in fish: how important is renal elimination?* *Environmental Science: Processes & Impacts*, 2022.

[51] Huiming Cao, Zhen Zhou, Zhe Hu, Cuiyun Wei, Jie Li, Ling Wang, Guangliang Liu, Jie Zhang, Yawei Wang, Thanh Wang and Yong Liang. *Effect of Enterohepatic Circulation on the Accumulation of Per- and Polyfluoroalkyl Substances: Evidence from Experimental and Computational Studies*. *Environmental Science & Technology*, 56:3214–3224, 2022.

[52] Alan F. Hofmann. *Chemistry and Enterohepatic Circulation of Bile Acids*. *Hepatology*, 4:4S–14S, 1984.

[53] Jian Shan Cai and Jin Hong Chen. *The Mechanism of Enterohepatic Circulation in the Formation of Gallstone Disease*. *The Journal of Membrane Biology*, 247:1067–1082, 2014.

[54] Kannan Krishnan, Sami Haddad and Michael Pelekis. *A Simple Index for Representing the Discrepancy between Simulations of Physiological Pharmacokinetic Models and Experimental Data*. *Toxicology and Industrial Health*, 11:413–421, 1995.

[55] *R: The R Project for Statistical Computing*. https://www.r-project.org/. 2023-08-31.

[56] *R package deSolve*. https://desolve.r-forge.r-project.org/. 2023-08-31.

[57] *R Interface to NLopt*. https://astamm.github.io/nloptr/index.html. 2023-08-31.

[58] Lu Lu, Xuhui Meng, Zhiping Mao and George Em Karniadakis. *DeepXDE: A Deep Learning Library for Solving Differential Equations*. *SIAM Review*, 63:208–228, 2021.

[59] *DeepXDE — DeepXDE 1.6.0 documentation*. https://deepxde.readthedocs.io/en/latest/.

List of Abbreviations

AAFE	Absolute Average Fold Error
ADME	Administration, Distribution, Metabolism and Excretion
ANN	Artificial Neural Networks
ASBT	Apical Sodium-dependent Bile acid Transporter
AUC	Area Under the Curve
CNN	Convolutional Neural Network
CYP P450 enzyme	Cytochrome P450 enzyme
e.g.	exampli gratia
etc.	et cetera
FCN	Fully-Connected Network
FDA	Food and Drug Administration
FFN	Feedforward Network
GeLU	Gaussian Error Linear Unit
MLP	Multilayer Perceptron
MRI	Magnetic Resonance Image
NC	Non-compartmental
NTCP	Na+ taurocholate cotransport polypeptide
ODE	Ordinary Differential Equations
PBK	Physiologically-Based Kinetics
PBKOF	PBK Objective Function
PD	Pharmacodynamics
PDE	Partial Differential Equations
PINN	Physics Informed Neural Network
PK	Pharmacokinetics
PFAS	per- and polyfluoroalkyl substances
PFBS	perfluorobutane sulfonic acid
PFHxS	perfluorohexane sulfonic acid
PFNA	perfluorononanoic acid
PFOA	perfluorooctanoic acid
PFOS	perfluorooctane sulfonic acid
ReLU	Rectified Linear Unit
RMSE	Root Mean Squared Error
RNN	Recurrent Neural Network
SGD	Stochastic Gradient Descent
TCA	Taurocholic Acid

WSSR	Weighted Sum of Squared Residuals