Bergische Universität Wuppertal

Fakultät für Mathematik und Naturwissenschaften

Institute of Mathematical Modelling, Analysis and Computational Mathematics (IMACM)

Preprint BUW-IMACM 22/13

Fabian Heldmann, Sarah Treibert, Matthias Ehrhardt,
and Kathrin Klamroth

PINN Training using Biobjective Optimization: The Trade-off between Data Loss and Residual Loss

June 1, 2022

http://www.imacm.uni-wuppertal.de

PINN Training using Biobjective Optimization: The Trade-off between Data Loss and Residual Loss

Fabian Heldmann[b], Sarah Treibert[a], Matthias Ehrhardt[a], Kathrin Klamroth[b]

[a] *University of Wuppertal, Chair of Applied Mathematics and Numerical Analysis, Gaußstrasse 20, 42119 Wuppertal, Germany*
[b] *University of Wuppertal, Chair of Optimization, Gaußstrasse 20, 42119 Wuppertal, Germany*

Abstract

Physics informed neural networks (PINNs) have proven to be an efficient tool to represent problems for which measured data are available and for which the dynamics in the data are expected to follow some physical laws. In this paper, we suggest a multiobjective perspective on the training of PINNs by treating the data loss and the residual loss as two individual objective functions in a truly biobjective optimization approach.

As a showcase example, we consider COVID-19 predictions in Germany and built an extended susceptibles-infected-recovered (SIR) model with additionally considered leaky-vaccinated and hospitalized populations (SVIHR model) to model the transition rates and to predict future infections. SIR-type models are expressed by systems of ordinary differential equations (ODEs). We investigate the suitability of the generated PINN for COVID-19 predictions and compare the resulting predicted curves with those obtained by applying the method of non-standard finite differences to the system of ODEs and initial data.

The approach is applicable to various systems of ODEs that define dynamical regimes. Those regimes do not need to be SIR-type models, and the corresponding underlying data sets do not have to be associated with COVID-19. The severe acute respiratory syndrome type 2 (SARS-CoV-2) was chosen as the

Email addresses: heldmann@uni-wuppertal.de (Fabian Heldmann), streibert@uni-wuppertal.de (Sarah Treibert), ehrhardt@uni-wuppertal.de (Matthias Ehrhardt), klamroth@uni-wuppertal.de (Kathrin Klamroth)

thematic scope because it is a topical issue.

Keywords: physics-informed neural networks, compartment models, COVID-19, SARS-CoV-2, epidemiology, loss function, multiobjective optimization, weighting parameters, Pareto front

---

1. Introduction

Physics informed neural networks (PINNs) [1], also called theory-inspired machine learning [2], have recently become a popular method for solving differential equations. By incorporating the residual of the differential equation into the loss function of a neural network-based surrogate model, PINNs can seamlessly combine measured data with physical constraints given by differential equations. PINNs can also be viewed as a *surrogate model* for solving differential equations by incorporating additional data or as a data-driven correction (or even discovery) of the underlying physical system.

We are currently situated in the fourth wave of the coronavirus pandemic, when interventions and protection measures are lifted towards the summer of the year 2022 in multiple countries, and vaccines against future mutations as well as the predominant B.1.1.529 (omicron) variant are worked on by big biotechnology companies, e.g. BioNTechPfizer, ImmunityBio, Sanofi, Valneva, Moderna or AstraZeneca [3]. 75.8 % of the German population is initially immunized and 59.5 % have received a booster vaccination by now.

The B.1.617.2 (delta) variant of SARS-CoV-2, which is characterized by a higher contagiosity than the previous B.1.1.7 (alpha), B.1.351 (beta) and P.1 (gamma) variants, has been observed in Germany since March 2021 and was the predominant variant in Germany during several months in the year 2021 [4]. In the autumn of 2021, the new omicron variant was detected and classified as concerning by the World Health Organization. Three sublines (BA.1, BA.2, BA.3) and a transmission advantage with respect to the delta variant were attributed to the omicron variant at the end of November 2021 [4]. The omicron variant quickly spread worldwide. Three recombinations of the omicron and

delta variant (XD, XE, XF) have already been registered as sublines [4].

Local peaks during the fourth COVID-19 wave in Germany were reached on November 28^{th} 2021 with 693, February 14^{th} 2022 with 2,434 and March 20^{th} 2022 with 2,619 daily infections per 1 million people. All of these peaks were larger than the global peaks of the three previous waves experienced in spring 2020 (69 on April 2^{nd} 2020), winter 2020/2021 (305 on December 22^{nd} 2020) and spring 2021 (257 on April 25^{th} 2021) [5].

The mathematical model used in this work to describe the population dynamics of COVID-19 is the *susceptible-vaccinated-infected-hospitalized-recovered* (SVIHR) model. Here, pre-symptomatic individuals are merged with symptomatic people in the infected compartment, so that we have a single infected compartment of people not hospitalized.

The contribution of our work consists of three large parts: Firstly, we establish the SVIHR model and build upon this a *Physics-Informed Neural Network* (PINN). The PINN method uses certain physics-informed constraints, expressed e.g. by differential equations, as part of the loss function of a corresponding deep neural network. Thus, the system of ODEs plays a crucial role in the training (i.e., the optimization) of the neural network.

PINNs were first introduced in the work of Raissi et al. and since then used to solve different forward and inverse problems [1]. The PINN approach trades off between the data-based and physical loss functions in the training process. This reduces the space of reasonable solutions to those that satisfy a 'physical law' to some degree, i.e., an SVIHR compartmental model in this case. The loss function of the PINN is based on reported data of recent infection events (*data loss part*), and a system of ODEs inheriting transition and transmission dynamics, from which so-called residual networks are computed using automatic differentiation (*residual loss part*). The data loss makes PINNs a data-driven technique. The PINN involves several fixed model parameters, from which transition parameters are computed. The uniqueness of our PINN approach lies in the fact that we apply it to the established SVIHR model and differentiate between training data covering the time since the outbreak of the pandemic in Germany and training

data involving exclusively one achieved peak.

Secondly, the constructed PINN is applied to pandemic-related data referring to a time-series of up to two years between 2020 and 2022. Two model parameters called *trainable parameters*, which determine the force of disease transmission and generated the PINN, are identified per run of the PINN. Multiple validation runs are performed to adapt the network structure and improve predicted scenarios, minimizing the error between forecasts of the pandemic wave and the corresponding reported data. The PINN is highly flexible with respect to the modification of model restrictions.

The third contribution of our work is the optimization of the parameter weighting the relation between the data and residual loss part. To achieve this, we interpret the training process as a biobjective optimization problem, where the residual loss and the data loss are considered as two independent and generally conflicting objective functions. Rather than combining these two objectives with a pre-determined and fixed weighting parameter, we identify suitable weighting parameters by generating a (rough) approximation of the Pareto front. For the training process, we adopt a scalarization-based approach that transforms the biobjective problem into a series of weighted-sum scalarizations. Favorable solutions are identified by repeated training runs with adaptively selected scalarization parameters. The resulting approximation of the Pareto front provides valuable information on the trade-off between data loss and residual loss. On one hand, this information can be used to assess the suitability of the employed physical model. On the other hand, a thorough analysis of the (approximated) Pareto front supports an informed selection of a suitable compromise, focusing more on the data or more on the physical model depending on the decision makers preferences and beliefs.

1.1. Related Research

Since the outbreak of the COVID-19 pandemic, a variety of compartmental models have been introduced as enhanced susceptible-infected-recovered (SIR) compartment models to study various aspects of the spread of SARS-CoV-2.

PINNs have been applied to compartment models and studied in the context of the COVID-19 pandemic as well.

For instance, Malinzi et al. applied a PINN to a susceptible-infected-recovered-deceased (SIRD) model in order to identify the behavioural dynamics of COVID-19 in the Kingdom of Eswatini between March 2020 and September 2021. They found that their PINN outperformed all other data analysis models even when given minimal quantities of training data [6].

Kharazmi et al. [7] considered different integer-order, fractional-order and time-delay models expressed as systems of ODEs. For the aim of analyzing the previous dynamics of COVID-19 in New York City, Rhode Island and Michigan states and Italy, they used PINNs, that were explained as able to perform parameter inference and simulation of the observed and unobserved dynamics simultaneously. Their results showed that purely statistical approaches were generally not well suited for long-term predictions of epidemiological dynamics, and integer-order models seemed to be more robust than fractional-order models, that were first developed by Pang et al. [8]. Moreover, they stated that no model could accurately capture all the dynamics that play out during an extended pandemic, but models with the ability to adjust key parameters during training could lead to more useful predictions [7].

Cai, Karnidadakis and Li calibrated the unknown model parameters of a *susceptible-exposed-infected-removed* (SEIR) model using the novel fractional physics-informed neural networks (fPINNs) deep learning framework in order to obtain reliable short-term predictions of the COVID-19 dynamics caused by Omicron variant [9]. Data from the National Health Commission of the People's Republic of China covering the 27^{th} February 2022 to end of April were used. For instance, predictions are able to capture sudden changes of the tendency for the new infected cases.

On the other hand, concerning the general PINN approach, we emphasize that the multiobjective nature of PINN training was recognized in several recent publications. Rohrhofer et al. [10] analyze the impact of different weights in a weighted sum objective of data loss and model loss by scanning the weight

interval. Also Jin et al. [11, Section 4.4.] studied the influence of weights in an experiment for turbulent channel flow, by manually tuning the weight in order to improve the results. Finally, Wang et al. [12, Algorithm 2.1] proposed an an adaptive rule, called 'learning rate annealing for PINNS', for choosing the weights online during the training process. The basic idea behind is to automatically tune the weights by using the back-propagated gradient statistics during model training to properly balance all terms in the loss function.

Their numerical results on diffusion equations and Navier-Stokes equations, respectively, impressively show the impact of the weight selection on the training success. Indeed, suboptimal results are obtained for several training runs, thus leaving room for improved multiobjective training approaches. When the physical model and the data are in good correspondence (this is, for example, the case when the data is artificially generated from the model at hand), an 'ideal' solution that simultaneously minimizes data and model loss can be sought.

Maddu et al. [13] suggest a multiobjective descent method that adaptively updates the weights using an inverse Dirichlet strategy to avoid premature termination. While they do not discuss convergence guarantees, their numerical results show a good performance in comparison with recent adaptations of multiobjective descent methods [14, 15] to PINN training [16]. Stochastic multiobjective gradient descent algorithms were introduced for general NN training in [17]. We also refer to self-adaptive PINNs [18] and to PINN training in which the loss weights are regarded as hyperparameters [19].

In a more general setting, multiobjective training approaches were suggested in [20] to trade-off between data loss and regularization terms in the context of image recognition. The different characteristics (slope and curvature) of the considered training goals are addressed by enhancing the stochastic multigradient descent approach [17] by pruning strategies, and by combining adaptive weighted-sum scalarizations with interval bisection. The latter supports the identification of favorable knee solutions on the Pareto front.

This paper is organized as follows: In Section 2, the compartment model for COVID-19 predictions is introduced. Firstly, the SIR model explained in

Section 2.1 to provide an insight into the basics of epidemic modelling. Then the system of ODEs of our compartment model, the SVIHR model, is defined along with the used transition rates and transmission rate in Section 2.2. All model parameters are listed in Table 1.

Section 3 is devoted to the methodological developments. The *Nonstandard Finite Difference* (NSFD) method is introduced in Section 3.1, where the concept of the scheme is explained, the so-called *denominator function* is derived and the NSFD scheme for the SVIHR model is established. Section 3.2 provides an introduction to *physics-informed neural networks* (PINNs) with a focus on the loss function and the suggested neural network structure. Here, a brief distinction between supervised and unsupervised neural networks in made. Section 3.3 introduces some aspects of biobjective optimization needed to examine the Pareto front that is obtained by biobjective PINN training approaches. Finally, in Section 3.4, we introduce a dichotomic search scheme aiming to quickly find Pareto optimal solutions and supporting an informed decision on the preferable trade-off between the data loss and the residual loss.

We present our results in Section 4. They comprise the validation of the prediction for the calendar weeks 33 in 2021 to 5 in 2022 in Section 4.1 with a focus on the effects of altered weighting parameters, and the forecast for the calendar weeks 44 in 2021 to 8 in 2022 in Section 4.2 with a focus on a more precise prediction. They are based on distinct training data sets, meaning different extracts from the reported COVID-19 data used by us. Finally, we present the results of the biobjective training in Section 4.3. A conclusion is drawn and an outlook to future work is given in Sec. 5.

2. A Compartment Model for COVID-19 Predictions

The compartment model used to compute the residual loss during PINN training in this paper is the *susceptible-vaccinated-infected-hospitalized-recovered* (SVIHR) model, which was proposed by Treibert and Ehrhardt in [21]. It is briefly derived in Section 2.2 again for the sake of completeness. Building upon

the basic *susceptible-infectious-recovered* (SIR) model introduced by Kermack and McKendrick in 1927 [22], the SVIHR model enhances the SIR model to include a vaccinated and a hospitalized compartment. A general short introduction to SIR models in mathematical epidemiology is provided in Section 2.1.

A comparison between the data-driven PINN approach with a distinct training data set and the NSFD method based on the SVIHR model in regard of infection and hospitalization numbers was made in [21]. With the PINN itself as well as the NSFD scheme, we computed future COVID-19 scenarios using PINN-optimized parameters.

Regarding the application of the SVIHR model in this paper, the used data and thus parameter values and compartment sizes are updated based on the data sources [23, 24, 25] compared to [21]. Whereas the focus on the comparison with the NSFD scheme is stronger in [21], we include validation runs with an additional *NSFD-induced* loss term in Section 4 of this paper. This new loss is based on iterations of the NSFD scheme and thus underlying system of ODEs representing the SVIHR model and the initial compartment sizes. We weight the resulting loss term with a second hyperparameter. The most relevant difference between our validation runs in this paper and [21] is that the PINN and so the compartment model are applied to different time periods with respect to training data. Whereas the training set covers the beginning of the outbreak in Germany until end of 2021 in [21], we focus on the training sets mentioned in Section 4.1 and Section 4.2 to focus on the difference between a long- and short-term prediction here.

An intensive focus on the performance of the NSFD scheme for a *susceptible-vaccinated-infected-intensive care-deceased-recovered* (SVICDR) model was put by Treibert and Ehrhardt in [26]. Here, the impact of parameter boundary modifications on predicted prevalence was investigated, taking into account data on Germany in the pandemic, an exponential increasing vaccination rate in the considered time window and trigonometric contact and quarantine rate functions. Our results showed that the NSFD can predict a global peak based solely on the mathematical model and defined parameters, but independently of a previously

experienced behavior of the infectious disease.

As determining the proportion of asymptomatic individuals in the total infected population is not our goal at this point, we do not incorporate a separate compartment of asymptomatic infected individuals, but assume at least very mild symptoms in infected individuals. The degree of infectivity of infected individuals can be regulated by adjusting the transmission rate in the model. Our model is adaptable to different vaccination and transmission scenarios.

2.1. The SIR Model in Mathematical Epidemiology

The basic SIR model consists of three compartments of susceptible (S), infected (I), and recovered (R) individuals. We denote with $K(t)$ the size of a compartment $K \in \{S, I, R\}$ at time t, where a time unit equals a week. Susceptible individuals have not yet become infected but may become ill. In the basic SIR model, infected individuals are also capable of infecting susceptible persons, i.e. they are assumed to infectious (without any delay) and may or may not have symptoms. Recovered individuals have overcome the disease and are neither infectious nor ill.

The total size of the population at time t is denoted by $N(t)$. The satisfaction of the equation

$$N(t) = S(t) + I(t) + R(t) \quad \text{with } N\colon [0, T] \to \mathbb{N},$$

means that the number of individuals in the system is the sum of the compartment sizes at each considered time point $t \in [0, T]$. The system (2) must have initial conditions $S_0 = S(0)$, $I_0 = I(0)$, $R_0 = R(0) = 0$ to be well-defined [27, p. 11]. The population size $N(t)$ is constant, i.e. $N(t) = N$, if the derivative of $N(t)$ is zero, which means that the system does not contain a recruitment rate Λ or a natural death rate.

Let p be the probability rate that a contact with a susceptible results in a transmission, and let ζ be the per capita contact rate, i.e. the number of contacts made by one infectious individual. Then ζN is the number of contacts per unit of time this infectious individual makes, and $\zeta N \frac{S}{N}$ denotes the number

of contacts with susceptible individuals that one infectious individual makes per unit of time. Moreover, we define a transmission rate constant β [27, p. 10]:

$$\beta = p\,\zeta\,. \tag{1}$$

If $I(t)$ stands for the number of infected individuals at time t (*prevalence)*, then $\beta\,S\,I$ denotes the number of individuals who become infected per unit of time (*incidence*). If ω_I is the recovery rate, we obtain the following system of ODEs, that describes the SIR model [27, p. 11]:

$$\begin{aligned} \frac{\mathrm{d}S(t)}{\mathrm{d}t} &= -\beta\,I(t)\,S(t), \\ \frac{\mathrm{d}I(t)}{\mathrm{d}t} &= \beta\,I(t)\,S(t) - \omega_I\,I(t), \\ \frac{\mathrm{d}R(t)}{\mathrm{d}t} &= \omega_I\,I(t). \end{aligned} \tag{2}$$

For the model in Equation (2), the maximum number of infected people reached in the regarded epidemic is given by

$$I_{\max} = -\frac{\omega_I}{\beta} + \frac{\omega_I}{\beta}\,\ln\!\big(\frac{\omega_I}{\beta}\big) + S_0 + I_0 - \frac{\omega_I}{\beta}\,\ln\!\big(S_0\big)\,. \tag{3}$$

Let

$$F(t) = 1 - e^{-\omega_I t}, \quad t \geq 0 \tag{4}$$

be the probability of recovering/leaving the infectious class in the interval $[0, t)$ [27, p. 11]. The function $F(t)$, with $F(t) = 0$ for $t < 0$, is a probability distribution. Then $f(t) = \frac{\mathrm{d}F(t)}{\mathrm{d}t}$ is the respective probability density function:

$$f(t) = \omega_I\,e^{-\omega_I t}\,. \tag{5}$$

If X denotes the average time spent in the infectious class, the mean time spent in the infected class can be computed as the first moment

$$\mathbb{E}\big[X\big] = \int_{-\infty}^{\infty} t\,f(t)\,dt = \int_{-\infty}^{\infty} t\,\omega_I\,e^{-\omega_I t}\,dt = \frac{1}{\omega_I}\,. \tag{6}$$

For SARS-CoV-2, the mean time of infectiousness is not clearly defined. With a mild or moderate course of disease, contagiousness clearly declines within the ten days after symptom occurrence. Contagiousness has to be distinguished from

positive test results, that can occur several weeks after catching the infection, although the infectiveness is usually on a very low level then [28].

In this basic form of the SIR model, the population is assumed to be closed so that no individual enters or leaves a compartment from the outside, and recovered individuals are completely immune so that they can never be reinfected [27, p. 13].

The transmission rate constant (*transmission risk*) of the system in (2) can be incorporated into a transmission rate function $\theta(t)$:

$$\theta(t) := \beta\,\zeta\,\big(1-q\big)\,I(t)\,. \tag{7}$$

Here, ζ is the contact rate. The parameter q symbolizes the degree of intervention strength, quarantine, and isolation measures implemented. We note that time-dependent functions for the transmission risk $\beta(t)$, contact rate $\zeta(t)$ or isolation rate $q(t)$ are possible. As shown in [26], modifying parameters in trigonometric functions $q(t)$ and $\zeta(t)$ were investigated with respect to the NSFD scheme effects the height or point in time of peak occurrence. The rate

$$\Theta(t) := \theta(t)\,\frac{S(t)}{N(t)} \tag{8}$$

is the standard incidence rate, which is similar to the mass action incidence $\beta\,S\,I$, but it is normalized by the total population size. The standard incidence is used for diseases for which the contact rate is assumed to not increase indefinitely most of the time [27, p.38].

2.2. The SVIHR Model

The SIR model was enhanced by a compartment of hospitalized individuals H and a compartment of fully, i.e. at least twice, vaccinated individuals V in [21]. We obtained our data sets from the Robert Koch-Institute (RKI) [23, 24] and the German COVID-19 Vaccination Dashboard [25], based on which parameter values and compartment sizes, referred to as *reported compartment sizes* or *reported data* in the sequel, were computed.

Infected individuals remain infected for T_I days until they recover, when a proportion ξ of all individuals transiting from the infected individuals are hospitalized. The exact daily or weekly number of infectious people among the infected ones is not known. The number of infections registered by the Robert Koch-Institute (RKI) is used to compute the reported size of the compartment I for all considered calendar weeks in this paper. This number is based on the number of infected individuals who are infectious enough so that the virus is usually verifiable via a rapid antigen test. Infectious and not infectious infected people are merged within the compartment I. The general degree of infectiousness of the individuals in I depends on the transmissibility of the virus and is included in the transmission rate.

According to the RKI, the concrete time period of contagiosity is not clearly defined, but infectiosity is highest right before and after the presence of first symptoms and drastically declines maximally 10 days after the very first symptom occurrence with a mild or moderate disease course [28]. We selected $T_I = 1.2$ weeks, i.e. 8.4 days, since we assumed a small time span of 1-2 days between the first showing of symptoms and getting tested. The parameter ω_1 is the rate at which persons per unit time (week) pass from compartment I to R. It is defined as

$$\omega_1 = \frac{1-\xi}{T_I}\,. \tag{9}$$

The rate η at which individuals reach the compartment H per unit of time is defined as

$$\eta = \frac{\xi}{T_I}. \tag{10}$$

As a proportion ξ of currently infected individuals is assumed to be hospitalized within T_I weeks, a proportion $1-\xi$ is assumed to recover within those T_I days, such that $\omega_1\, I(t)$ people recover and $\eta\, I(t)$ individuals are hospitalized within week t. It is assumed here that hospitalized individuals cannot infect susceptible individuals because of their isolated state. They remain infected T_H days from the time of their hospitalization. In a German academic survey with 1,426 COVID-19 patients with an acute respiratory disease, an average duration of

hospital stay of 10 days was observed [28]. We selected $T_H = 1.5$ weeks, i.e. 10.5 days. A proportion $\mathcal{M}$ of persons transiting from the hospitalized compartment die from disease-related causes rather than recover. Consequently, the rate ω_2 at which persons per unit time pass from compartment H to R is given by

$$\omega_2 = \frac{1 - \mathcal{M}}{T_H}. \tag{11}$$

The vaccinated compartment V contains all susceptible individuals who have received a COVID-19 vaccination. It is reached from the compartment S at a rate $\mathcal{V}$. If vaccination does not guarantee complete immunity to infection, we speak of a *leaky vaccination*. Due to the assumed leakiness, all vaccinated individuals have a lower probability of contracting the infection than susceptible individuals in compartment S. If an all-or-nothing vaccine was assumed, vaccinated people would be completely protected from the infection to a specific portion of the susceptible class per unit time t, whereas the other susceptibles did not gain any protection. Let κ denote the residual probability of infection after vaccination. The rate at which vaccinated individuals reach the infected compartment I is $\kappa\,\theta(t)$.

Furthermore, we incorporate a constant system inflow, the so-called recruitment rate Λ (e.g. birth of new individuals that can get infected), and the natural mortality rate μ. The recruitment and natural death rate are set to zero as they are regarded as equal in both [21] and this paper, but are still included in the system of ODEs for the purpose of properly deriving the denominator function in the NSFD scheme, see Section 3.1. The total population size is kept constant like this. The corresponding system of ODEs has the following form:

$$\begin{aligned}
\frac{\mathrm{d}S(t)}{\mathrm{d}t} &= \Lambda - \theta(t)\,\frac{S(t)}{N(t)} - (\mathcal{V} + \mu)\,S(t),\\
\frac{\mathrm{d}V(t)}{\mathrm{d}t} &= \mathcal{V}\,S(t) - \theta(t)\,\kappa\,\frac{S(t)}{N(t)} - \mu\,V(t),\\
\frac{\mathrm{d}I(t)}{\mathrm{d}t} &= \theta(t)\,\big(1 + \kappa\big)\frac{S(t)}{N(t)} - \big(\eta + \omega_1 + \mu\big)\,I(t),\\
\frac{\mathrm{d}H(t)}{\mathrm{d}t} &= \eta\,I(t) - (\omega_2 + \mu)\,H(t),\\
\frac{\mathrm{d}R(t)}{\mathrm{d}t} &= \omega_1\,I(t) + \omega_2\,H(t) - \mu\,R(t).
\end{aligned} \tag{12}$$

The system (12) extends the simple system (2) by the differential equations for $V(t)$ and $H(t)$, describing the inflow into and the outflow from the compartment V or H, respectively, as well as the recruitment rate Λ and the natural death rate μ. Figure 1 shows the dynamical system described by (12). Blue arrows from one compartment to another indicate a transition, where the compartment from which the red dashed arrow emanates can infect susceptibles. Table 1 lists

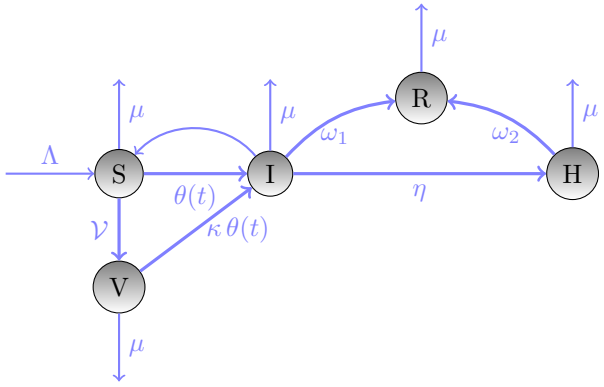


Figure 1: Compartment model for the SVIHR model

the model parameter definitions and used values. We note that the parameter values for the SVIHR model used in this paper stated in Table 1 differ from the ones used in [21].

3. Finding Optimized Weights in PINN Approaches

We implement a physics informed neural network (PINN) that is trained both w.r.t. German COVID-19 data and w.r.t. the SVHIR model introduced in Section 2.2 above. This technique is validated using error computations with regard to reported data. The method and structure of PINNs is explained in Section 3.2. Moreover, scenarios generated using the PINN are compared to those produced using the technique of nonstandard finite difference (NSFD) schemes. A short introduction to NSFD schemes and the application on the SVIHR model are outlined in Section 3.1.

Parameter	**Definition**	**Parameter Value used in 4.1**	**Parameter Value used in 4.2**
N	total population size	83,100,000 [29]	83,100,000 [29]
β	transmission risk parameter	*determined via PINN*	*determined via PINN*
κ	residual infection prob. after vaccination	*determined via PINN*	*determined via PINN*
T_I	average length of infection period	1.2 [28]	1.2 [28]
T_H	average length of hospitalization period	1.5 [28]	1.5 [28]
$\mathcal{V}$	vaccination coefficient	0.0159 [25]	0.0231 [25]
ξ	hospitalization coefficient	0.0862 [24]	0.0735 [24]
$\mathcal{M}$	mortality coefficient	0.0232 [23]	0.0142 [23]
ω_1	weekly recovery rate for infected people	0.7615	0.7721
ω_2	weekly recovery rate for hospitalized people	0.6512	0.6572
η	weekly hospitalization rate	0.0719	0.6125

Table 1: Parameters of the SVIHR model and their values in the implementation of the PINN.

Since the PINN is trained w.r.t. two loss terms, we take a biobjective approach to investigate the influence of weighting parameters on each loss term. We consider the two losses as independent objective functions and want to find weighting parameters to achieve an approximation of the Pareto front. We therefore first introduce certain aspects of bicriteria optimization in Section 3.3 and then introduce a dichotomic search scheme to efficiently approximate the Pareto front in Section 3.4.

3.1. Nonstandard Finite Difference Schemes

NSFD schemes trace back to a paper by Mickens published in 1989 [30]. Their structural properties originate from studies of special groups of differential equations, for which exact finite difference schemes are not available. For reasons of completeness, the NSFD scheme for the SVIHR model is derived at this point again. It has already been derived in a similar way in [21].

In NSFD schemes, derivatives have to be modelled by proper discrete analogues, i.e. nonstandard difference quotients of the form, cf. [31]

$$\frac{\mathrm{d}u(t)}{\mathrm{d}t} \to \frac{u_{n+1} - \psi(h)u_n}{\phi(h)}\,, \tag{13}$$

where $t_n = n\,h$, u_n is the approximation of $u(t_n)$, and $\psi(h) = 1 + \mathcal{O}(h)$. Using this rather general time discretization (13) in NSFD schemes our aim is to model the asymptotic long-time behaviour of the solution. A numerical scheme for a system of first-order differential equations is called NSFD scheme if at least one of the following conditions hold [31]:

- Discrete representations for derivatives must, in general, have nontrivial denominator functions. Here, the first-order derivatives in the system are approximated by the generalized forward difference method $\frac{\mathrm{d}u_n}{\mathrm{d}t} \approx \frac{u_{n+1}-u_n}{\phi(h)}$, where $u_n \approx u(t_n)$ and $\phi \equiv \phi(h) > 0$ is the so-called *denominator function* such that $\phi(h) = h + \mathcal{O}(h^2)$, with h the step size.

- The consistency orders of the finite difference quotients should be equal to the orders of the corresponding derivatives appearing in the differential equations.

- The nonlinear terms are approximated by non-local discrete representations, for instance by a suitable function of several points of a mesh, like $u^2(t_n) \approx u_n u_{n+1}$ or $u^3(t_n) \approx u_n^2 u_{n+1}$.
- Special conditions that hold for either the ODE and/or its solutions should also hold for the difference equation model and/or its solution, e.g. the equilibrium points of the underlying ODE system.

NSFD schemes satisfy the positivity condition and the conservation law for $\Lambda = \mu = 0$ yielding stability of the scheme. The equilibrium points of the ODE model appear in the proposed NSFD-scheme as well.

In the sequel, we will derive an appropriate *denominator function* $\phi(h)$ for the NSFD discretization of the system (12). This function is chosen such that the numerical solution exhibits the same asymptotic behaviour as the analytic solution. To do so, we consider the total population $N = S + V + I + H + R$ of the ODE system (12). Adding the equations of (12), a differential equation describing the dynamics of the total population N is obtained as

$$\frac{\mathrm{d}N(t)}{\mathrm{d}t} = \Lambda - \mu\, N(t)\,, \tag{14}$$

which is solved by

$$N(t) = \frac{\Lambda}{\mu} + \Big(N(0) - \frac{\Lambda}{\mu}\Big)\, e^{-\mu t} = N(0) + \Big(N(0) - \frac{\Lambda}{\mu}\Big)\, (e^{-\mu t} - 1), \tag{15}$$

where $N(0) = S(0) + V(0) + I(0) + H(0) + R(0)$. Now, adding the equations in the discrete NSFD model (21) yields

$$\frac{N^{n+1} - N^n}{\phi(h)} = \Lambda - \mu\, N^{n+1}, \tag{16}$$

i.e.

$$\begin{aligned} N^{n+1} = \frac{N^n + \phi(h)\Lambda}{1 + \phi(h)\,\mu} &= N^n - \Big(N^n - \frac{\Lambda}{\mu}\Big)\frac{\phi(h)\,\mu}{1 + \phi(h)\,\mu} \\ &= N^n + \Big(N^n - \frac{\Lambda}{\mu}\Big)\Big(\frac{1}{1 + \phi(h)\,\mu} - 1\Big). \end{aligned} \tag{17}$$

The denominator function can be derived by comparing Equation (16) with the discretized version of Equation (15), that is

$$N^{n+1} = N^n + \Big(N^n - \frac{\Lambda}{\mu}\Big)\, (e^{-\mu h} - 1), \quad h = \Delta t, \tag{18}$$

such that the (positive) denominator function is defined by

$$\frac{1}{1+\phi(h)\,\mu} = e^{-\mu t}, \tag{19}$$

i.e.

$$\phi(h) = \frac{e^{\mu h}-1}{\mu} = \frac{1+\mu h+\frac{1}{2}\mu^2 h^2+\ldots-1}{\mu} = h+\frac{\mu h^2}{2}+\ldots = h+\mathcal{O}(h^2). \tag{20}$$

Now, making use of the denominator function in (13), the NSFD discretization can be established. The implicit form of this discretization is provided in Equation (21). Here, $\phi(h)$ is given by (20)

$$\begin{aligned}
\frac{S^{n+1}-S^n}{\phi(h)} &= \Lambda - \beta\, I^n\, S^{n+1} - (\mathcal{V}+\mu)\, S^{n+1},\\
\frac{V^{n+1}-V^n}{\phi(h)} &= \mathcal{V}\, S^{n+1} - \beta\,\kappa\, I^n\, S^{n+1} - \mu\, V^{n+1},\\
\frac{I^{n+1}-I^n}{\phi(h)} &= \beta\,(1+\kappa)\, I^{n+1}\, S^{n+1} - (\eta+\omega_1+\mu)\, I^{n+1},\\
\frac{H^{n+1}-H^n}{\phi(h)} &= \eta\, I^{n+1} - (\omega_2+\mu)\, H^{n+1},\\
\frac{R^{n+1}-R^n}{\phi(h)} &= \omega_1\, In + 1 + \omega_2\, H^{n+1} - \mu\, R^{n+1}\,.
\end{aligned} \tag{21}$$

We can rewrite the scheme in order to obtain an explicit variant of it, as to be found in (22). From the explicit representation we can deduce that this scheme preserves the positivity.

$$\begin{aligned}
S^{n+1} &= \frac{S^n+\phi(h)\Lambda}{1+\phi(h)\,(\beta\, I^n+\mu)},\\
V^{n+1} &= \frac{V^n+\phi(h)\, S^{n+1}\,(\mathcal{V}-\beta\,\kappa\, I^n)}{1+\phi(h)\,\mu},\\
I^{n+1} &= \frac{I^n}{1+\phi\,(\eta+\omega_1+\mu-\beta\,(1+\kappa)\, S^{n+1})},\\
H^{n+1} &= \frac{\phi(h)\,\eta\, I^{n+1}+H^n}{1+\phi\,(\omega_2+\mu)},\\
R^{n+1} &= \frac{R^n+\phi(h)\,(\omega_1\, I^{n+1}+\omega_2\, H^{n+1})}{1+\phi(h)\,\mu}\,.
\end{aligned} \tag{22}$$

The calculation must be done in exactly this order. All parameters appearing in these type of epidemic models are always non-negative.

3.2. Physics-informed Neural Networks for Compartment Models

Physics-informed neural networks (PINN) are neural networks that include the laws of dynamical systems into a deep learning framework. Machine learning has emerged as an alternative to numerical discretization in high-dimensional problems governed by partial differential equations. Nonetheless, a sufficient amount of data as required for training deep neural networks is not necessarily available. In such cases, missing data can be substituted by incorporating additional information obtained from enforcing the physical laws of dynamical systems [32]. Such laws can be described by partial or ordinary differential equations. One example where dynamical systems can be used are populations undergoing transitions between different infected or uninfected states during an epidemic, as considered in this paper.

PINNs can approximate the solutions of differential equations by training a loss function incorporating the initial and boundary conditions and the residual at so-called collocation points [33]. Instead of approximating solutions of differential equations, PINNs can use a system of differential equations describing a certain real-world process along with time-series data sets describing the past course of such a process for the purpose of generating predicitions for future progressions.

The loss function of a corresponding neural network includes not solely the so-called *data error* related to the difference between the output of the network and the reported data used, but also the so-called *residual error* related to the ODEs or PDEs.

Olumoyin et al. [34] refer to a type of feedforward neural network including epidemiological dynamics such as lockdown into their loss function by using the term *Epidemiology-Informed Neural Network (EINN)*. EINNs extend PINNs for epidemiology models and are able to capture the dynamics of the spread of the disease and the influence of the mitigation measure. The loss function is enhanced to include time-varying rates using epidemiology facts about the infectious disease [34].

Shaier, Raissi and Seshaiyer [35] describe a type of PINN-based neural

network that can be applied to increasingly complex systems of differential equations describing various known infectious diseases with the term *Disease-Informed Neural Networks (DINN)*. DINNs can be systematically applied to increasingly complex governing systems of differential equations describing infectious diseases. They are able to effectively learn the dynamics of the disease and forecast its progression a month into the future from real-life data [35].

The neural network established and applied in this paper can be described as a special type of EINN or DINN based on a special kind of epidemic compartment model called SVIHR model regarding the susceptible, vaccinated, infected, hospitalized and recovered part of the population. We focus on the German population in this paper, using data provided by the Robert Koch-Institute [23, 24] and the German COVID-19 Vaccination Dashboard [25]. The model parameters included in (12) can be comprised in a vector ϑ:

$$\vartheta = [\beta, \zeta, q, \mathcal{V}, \kappa, \xi, T_I, T_H, \mathcal{M}, T_H]^\top . \tag{23}$$

This vector is partitioned into fixed parameters p_f and trainable parameters p:

$$\begin{aligned} p_f &:= [\zeta, q, \mathcal{V}, \xi, T_I, T_H, \mathcal{M}, T_H]^\top, \\ p &:= [\beta, \kappa]^\top . \end{aligned}$$

So we decide to optimize the transmission risk β and the factor κ during the training process of the neural network.

We fit a vector of t_l points in time $T = [t_1, \ldots, t_l]^\top$ as input vector to our network. Let

$$\hat{\mathcal{K}}(t) = [\hat{\mathcal{K}}^1(t), \ldots, \hat{\mathcal{K}}^n(t)]^\top$$

be the vector of the reported sizes of those n compartments at time t. If $\mathcal{K}$ denotes the n compartments, we express the terms on the right-hand side of (12) by

$$F_p(\mathcal{K}) = [F_p^1(\mathcal{K}), \ldots, F_p^n(\mathcal{K})]^\top .$$

Here, n is the number of compartments, i.e. $n = 5$ in case of the SVIHR model. The subscript p stands for the model parameters that the system of ODEs

depends on and thus the solution will depend on, too. It holds that $F_p^j = F_p(\mathcal{K}^j(t)) \in \mathcal{C}(\mathbb{R})$, for all $j \in \{1, \ldots, n\}$. The system of ODEs in (12) can then be discretized as

$$\frac{\mathrm{d}\mathcal{K}(t)}{\mathrm{d}t} - F_p(\mathcal{K}) = 0, \qquad t \in \{t_1, \ldots, t_l\}\,. \tag{24}$$

Our PINN

$$\mathcal{PINN}_p^W : \mathbb{R} \to \mathbb{R}^n$$

is used to approximate the solution

$$\mathcal{K}_p = [S, V, I, H, D, R]^\top : \mathbb{R} \to \mathbb{R}^n\,, \tag{25}$$

of the system of ODEs (12) by performing error minimization during training [36]. The superscript W represents the weights used during the forward and backward propagation in the neural network. At time instance t, $t \in \{t_1, \ldots, t_l\}$, the solution is expressed as

$$\mathcal{K}_p(t) = [\mathcal{K}_p^1(t), \ldots, \mathcal{K}_p^n(t)]^\top\,,$$

where $\mathcal{K}_p^j(t) \in \mathcal{C}^1(\mathbb{R})$ is the output of the PINN for the j^{th} compartment at time t. The parameters W and p are optimized during the backpropagation process of the neural network such that $\mathcal{PINN}_{\mathrm{p}}^{\mathrm{W}}$ fits the reported data $\hat{\mathcal{K}}$ in a least-squares sense [36]. In the k^{th} training iteration J_k, $k \in \{1, \ldots, M\}$, with PINN output $\mathcal{PINN}_{\mathrm{p}}^{\mathrm{W}}(J_k)$, we compute the usual *data error* defined as

$$\mathrm{MSE}_{\mathcal{U}} = \mathrm{MSE}_{\mathcal{U}}(W, p) := \frac{1}{M}\sum_{k=1}^{M} \| \mathcal{PINN}_{\mathrm{p}}^{\mathrm{W}}(J_k) - \hat{\mathcal{K}}_j \|^2\,. \tag{26}$$

Next, let us extend the loss function of the PINN by the additional term

$$\mathcal{F}_p(\mathcal{PINN}_{\mathrm{p}}^{\mathrm{W}}, J_k) := \frac{\mathrm{d}\,\mathcal{PINN}_{\mathrm{p}}^{\mathrm{W}}(J_k)}{\mathrm{d}t}\Big|_{t=k} - F_p\big(\mathcal{PINN}_{\mathrm{p}}^{\mathrm{W}}(J_k)\big)\,, \tag{27}$$

where

$$\mathcal{F}_p(\mathcal{PINN}_{\mathrm{p}}^{\mathrm{W}}, J_k) = 0 \quad \text{for all } t \in \{t_1, \ldots, t_l\} \tag{28}$$

means that the PINN solves the given system of ODEs exactly. The computation of the time derivative of the neural network output $\frac{\mathrm{d}\,\mathcal{PINN}_{\mathrm{p}}^{\mathrm{W}}(J_k)}{\mathrm{d}t}\Big|_{t=k}$ can be

performed using automatic differentiation [37]. Then the physics-informed part of the loss function, the *residual error*, is given by

$$\mathrm{MSE}_{\mathcal{F}} = \mathrm{MSE}_{\mathcal{F}}(W,p) := \frac{1}{M}\sum_{k=1}^{M} \left\| \mathcal{F}_p(\mathcal{PINN}^{\mathrm{W}}_{\mathrm{p}}, J_k) \right\|^2 . \tag{29}$$

We introduce a hyperparameter and weighting factor $\alpha \in [0,1]$ weighting the data loss and residual loss in the loss function. We define the loss function as

$$\mathcal{L}_\alpha = \mathcal{L}_\alpha(W,p) := \alpha\ \mathrm{MSE}_{\mathcal{U}} + (1-\alpha)\ \mathrm{MSE}_{\mathcal{F}} \tag{30}$$

and the minimization problem of the neural network as

$$\min_{W,p}\big(\mathcal{L}_\alpha\big)\,. \tag{31}$$

The loss function (30) can be enhanced by a third term to add another criterion to the respective optimization problem. The data loss error (26) can be further subdivided by introducing additionally a new so-called *NSFD-induced error* $\mathrm{MSE}_{\mathrm{NSFD}}$ that can be defined by computing the mean-squared error between the application of the system of ODEs to the network output of the current iteration and the current output of the proposed NSFD scheme designed especially for the SVIHR compartment model. The idea behind is that this loss function $\mathrm{MSE}_{\mathrm{NSFD}}$ takes into account which numerical solution method is used for solving the system of ODEs, e.g. compared to using MATLAB's standard ODE45 solver. Thus, the adequate numerical discretization method of NSFD can be used as another restrictive factor of the parameter optimization. Our results in Section 4 show that the NSFD loss term yields scenarios with smaller errors and more variable outputs. However, the impact on the biobjective network training turned out to be rather small.

Let the output of the NSFD scheme for a compartment K^j , $j \in \{1,\dots,n\}$, after M training steps be given by ${K^j}_{\mathrm{NSFD}} \in \mathbb{R}^m$, where $m \geq l$ is the number of weeks predicted by the respective NSFD scheme. As before, l stands for the number of weeks that the PINN makes predictions for. This NSFD scheme needs parameter values ϑ and initial compartment sizes $\hat{\mathcal{K}}^0$ as inputs. Then,

the new numerical loss term can be defined as

$$\mathrm{MSE}_{\mathrm{NSFD}} = \mathrm{MSE}_{\mathrm{NSFD}}(W,p) := \frac{1}{M}\sum_{k=1}^{M} \| {K^j}_{\mathrm{NSFD}} - F_p\big(\mathcal{PIMN}^{\mathrm{W}}_{\mathrm{p}}(J_k)\big)\|^2 \,. \tag{32}$$

It is weighted with a parameter γ here, such that the new loss function becomes

$$\mathcal{L}_{\alpha,\gamma} = \gamma\ \mathrm{MSE}_{\mathrm{NSFD}} + (1-\gamma)\,\mathcal{L}_{\alpha}\,. \tag{33}$$

The NSFD-induced loss (32) can be viewed as a type of data loss calculated not based on the reported data but the output of the NSFD scheme with initially reported compartment sizes. Since the numerical NSFD scheme is based on the underlying compartment model and system of ODEs, the loss term can also be described as partial physical loss.

In the computation of this loss in the k^{th} training step, the available values for β and κ of this k^{th} training iteration are used. For all implementations, the programming language *Python* and the deep-learning framework of *PyTorch* are used. We establish two input and two output layers per PINN predicting the size of one compartment for l following weeks, as e.g. infection numbers. The inputs are the t_l points in time as well as the initial sizes of the five compartments at the time at which a respective prediction starts. The outputs are an optimized parameter vector $p = [\beta, \kappa]^\top$ as well as inferred sizes of the respective compartment $\mathcal{K}_p$ at the points in time $t_{l+1}, \ldots, t_{2\,l}$.

During the validation process, the network structure was adapted step by step. Our final version of the PINN has six linear layers with 300 neurons each and a linear output layer of size $300 \times t_l$ for the output of infection or hospitalization numbers. Moreover, it has three linear layers with 200 neurons each and a rectified linear unit output layer of size 200×2 for the output p. The ADAM optimizer, rectified linear unit activation functions and the learning rate $lr = 0.003$ are applied. A good overview of current descent methods in machine learning is provided by [38].

The data sets obtained from the RKI [23, 24] and Germany Vaccination Dashboard [25] were pre-processed and then fed into the PINN. They refer to

the calendar weeks 10 in 2020 to 14 in 2022. Weekly case-hospitalization, case-fatality and vaccination rates were computed based on the given data sets. The RKI registers deceased individuals, in whom the SARS-CoV-2 pathogen was detected, as people who died from COVID-19. With respect to the underlying compartment size data set $\hat{\mathcal{K}} \in \mathbb{R}^{n \times t_l}$, we firstly select $t_l = 76$, covering the calendar weeks 10 in 2020 to 32 in 2021, and compute prediction errors starting from the calendar week 33 in 2021.

The described PINN training does not use the labelled training data, where compartment sizes $\hat{\mathcal{K}}$ are assigned to specific weeks $t_i, i \in \{1, \ldots, l\}$, as its direct input but for the computation of the data loss in every single training step. Since we do not use completely unlabelled data as the training basis here, we cannot describe our PINN approach as an unsupervised neural network. Nonetheless, we do not have a typical kind of supervised classification network. Consequently, the following validation process in not a typical sensitivity or specificity analysis. We focus on the investigation of the predicted trends with local maxima and minima during the course of the pandemic in specific time periods. Errors between the reported data and predictions are computed to more specifically compare the results obtained with different weighting parameter values.

The weighted loss function consists of the data loss and the residual loss term. As pointed out in [32], the training using the data loss (i.e. measurements, physics-uninformed) is regarded as supervised learning while the training w.r.t. the residual loss using the governing differential equation (physics-informed) is regarded a unsupervised learning.

3.3. Conflicting Training Goals: Pareto Front and Trade-Off Analysis

In this paper, we take a biobjective perspective on the optimization problem described in (30). This is the basis for a thorough trade-off analysis regarding the two loss terms $\mathrm{MSE}_{\mathcal{U}}$ and $\mathrm{MSE}_{\mathcal{F}}$. Rather than considering a weighted sum of these two training goals with a fixed weighting parameter, we consider both optimization goals independently and comprise them in a vector-valued objective function $\mathcal{L}$ that maps every feasible solution vector (W, p) to a two-

dimensional outcome vector $y = \mathcal{L}(W,p) \in \mathbb{R}^2$:

$$\min_{W,p} \mathcal{L}(W,p) = \min_{W,p} \big(\mathrm{MSE}_{\mathcal{U}}(W,p), \mathrm{MSE}_{\mathcal{F}}(W,p)\big). \tag{34}$$

As before, W denotes the neural network weights and p the trainable parameters of the compartment model. Let us emphasize here that the objective values also depend on the training data x^d, which are usually assumed to be given and fixed. All variations of the input data are thus highlighted to avoid misinterpretations of the results. To analyze the biobjective optimization problem (34), we first review some basic concepts from the field of multiobjective optimization. For a more detailed introduction into this field see, e.g., [39].

Towards this end, we denote by Y the *outcome set of problem* (34) that includes all possible outcome vectors $y = \mathcal{L}(W,p) \in \mathbb{R}^2$. A solution $(\hat{W},\hat{p})$ is called *efficient* or *Pareto optimal* if there exists no other solution (W,p) such that $\mathcal{L}(W,p) \leq \mathcal{L}(\hat{W},\hat{p})$, i.e., if there is no other solution (W,p) such that $\mathrm{MSE}_{\mathcal{U}}(W,p) \leq \mathrm{MSE}_{\mathcal{U}}(\hat{W},\hat{p})$ and $\mathrm{MSE}_{\mathcal{F}}(W,p) \leq \mathrm{MSE}_{\mathcal{F}}(\hat{W},\hat{p})$ where at least on of these two inequalities is strict. If $(\hat{W},\hat{p})$ is Pareto optimal, then the corresponding outcome vector $\mathcal{L}(\hat{W},\hat{p})$ is called *nondominated point*. Hence, Pareto optimal solutions are those solutions that can not be improved in one loss function without deterioration in the other loss function. The set of all Pareto optimal solutions (nondominated points, respectively) is denoted by X_E (Y_N, respectively). Figure 2 shows an example of a set of nondominated points Y_N within a set of outcome vectors Y in $\mathbb{R}^2$. Note that Y_N is also often referred to as *Pareto front*.

In order to better analyze the trade-off between the data loss and the residual loss, we want to approximate the Pareto front of the problem (34). This can be implemented by solving a series of parametric single-objective subproblems, so-called *scalarizations* (see again, e.g., [39]). To keep these subproblems simple, we use a *weighted sum* approach that leads to the single-objective optimization problem (31) with the objective function (30), i.e.,

$$\min_{W,p}\big(\mathcal{L}_\alpha(W,p)\big) = \min_{W,p}\big\{\alpha\ \mathrm{MSE}_{\mathcal{U}}(W,p) + (1-\alpha)\ \mathrm{MSE}_{\mathcal{F}}(W,p)\big\} \tag{31}$$

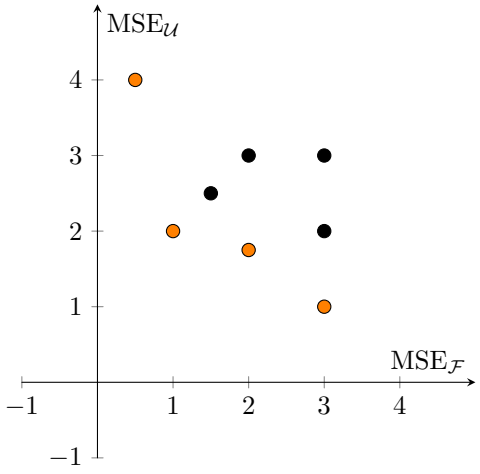


Figure 2: Points belonging to Y_N are depicted in orange.

It is a well-known fact [39] that for all weighting parameters $\alpha \in (0,1)$, an optimal solution of (31) is always part of Y_N. However, the converse statement is only true under certain convexity assumptions. Indeed, the complete set Y_N can be generated by varying the weighting parameter $\alpha \in (0,1)$ whenever the set Y is $\mathbb{R}^2_{\leq}$-convex, a property that can generally not be guaranteed in neural network training. We recall that Pareto optimal solutions $(\hat{W}, \hat{p}) \in X_E$ are called *supported* if there is some $\alpha \in (0,1)$ such that $(\hat{W}, \hat{p}) \in X_E$ is an optimal solution of (31). The sets of all supported efficient solutions and supported nondominated points are denoted X_{sE} and Y_{sN}, respectively. Note that all supported nondominated points are located on the boundary of the convex hull of Y (see, e.g., [40]). Figure 3 illustrates supported as well as unsupported nondominated outcome vectors for an exemplary outcome set Y.

In order to obtain a first approximation of the Pareto front, we used a predefined set of weighting parameters α in problem (31) ranging from 0.05 to 0.95 in steps of 0.05. The results of this search can be seen in Figure 4. This scan of the Pareto front required 19 training runs in total, which is generally too costly. Moreover, the distribution of points on the Pareto front can not be expected to always be as uniform as seen in Figure 4. Indeed, even for biobjective and

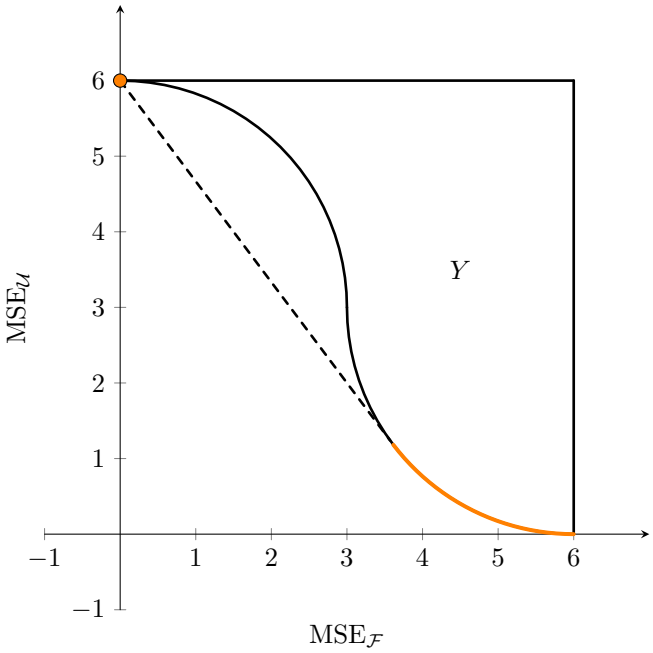


Figure 3: The set of supported nondominated points Y_{sN} of an exemplary problem with $Y \subset \mathbb{R}^2$, shown here in orange. The dashed line depicts the part of the boundary of the convex hull of Y that is not part of the set Y.

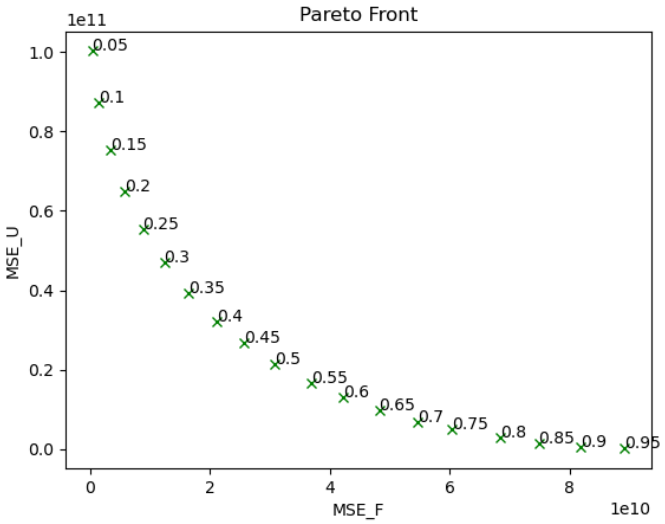


Figure 4: Scan of the Pareto front with pre-defined weighting parameters α. The value of α is indicated for each generated outcome vector.

convex problems, pre-defined weighting parameters may lead to very un-evenly distributed points on the Pareto front, see, e.g., [41]. Moreover, this approach does not scale well to higher-dimensional problems that include more than two training objectives.

In the next section we present an alternative approach that supports an adaptive selection of weighting parameters and thus allows for a problem-dependent choice of the accuracy of the approximation and the overall computational time (i.e., the number of training runs).

3.4. Dichotomic Search for Adaptive Pareto Front Approximations

In the previous section, we scanned the Pareto front to find solutions with meaningful trade-offs. But it would be desirable to find such solutions without scanning the whole Pareto front to save computational time and to perform fewer training runs. This can be realized with a scalarization-based dichotomic search algorithm as described in [40]. It aims at an automatic adaptation to the curvature and scaling of the problem in order to quickly find a diverse set of

solutions. Moreover, dichotomic search can be easily integrated in an interactive procedure that allows to zoom in into specific parts of the Pareto front that are most interesting to the decision maker, see, e.g., [42].

The following ideas can be found in [40]. In the biobjective case, the dichotomic search comes down to solving a sequence of weighted sum scalarizations (31) with $\alpha \in (0,1)$ and makes use of the fact that in the two-dimensional case, for two nondominated points y^r and y^s it holds that $y_1^r < y_1^s$ implies $y_2^r > y_2^s$. A weighted sum scalarization (31) with $\alpha = (y_2^r - y_2^s)/c > 0$ and $1-\alpha = (y_1^s - y_1^r)/c > 0$, where $c = y_2^r - y_2^s + y_1^s - y_1^r$, is then solved to find new supported points between y^r and y^s. The weighting parameter α hence defines a normal vector to the line segment connecting y^r and y^s since

$$\begin{aligned}\left(y^r - y^s\right)^\top \begin{pmatrix} \alpha \\ 1-\alpha \end{pmatrix} &= \begin{pmatrix} y_1^r - y_1^s, & y_2^r - y_2^s \end{pmatrix} \begin{pmatrix} \alpha \\ 1-\alpha \end{pmatrix} \\ &= c \cdot \begin{pmatrix} -(1-\alpha), & \alpha \end{pmatrix} \begin{pmatrix} \alpha \\ 1-\alpha \end{pmatrix} = 0.\end{aligned}$$

This is illustrated in the left of Figure 5. Solving the weighted sum problem with the new weighting parameter α leads to a nondominated point y^t (if the problem is solved to global optimality) for which two cases can occur:

1. If $(\alpha, 1-\alpha)^\top y^t < (\alpha, 1-\alpha)^\top y^r$, then y^t is a new supported nondominated point. Two new subproblems are generated, one of which is defined by y^r and y^t while the other one is defined by y^t and y^s. This case is illustrated in the right of Figure 5.
2. If $(\alpha, 1-\alpha)^\top y^t = (\alpha, 1-\alpha)^\top y^r = (\alpha, 1-\alpha)^\top y^s$, then y^t lies on the line segment connecting y^r and y^s and the search can stop in this interval.

The dichotomic search is progresses in *levels*, where level 1 contains the outcome vectors of the weighted sum scalarization with the two initial weights α_1 and α_2, $\alpha_1 < \alpha_2$, and one dichotomy step (see the left part of Figure 5 for an illustration of the associated weighting parameter). In level 2, weighted sum scalarizations are solved for all weights defined by the line segments comprising the convex hull

of the current approximation of the Pareto front (see the right part of Figure 5 for an illustration). The search is repeated until a prespecified number of levels has been evaluated, or until no new subproblems have been generated.

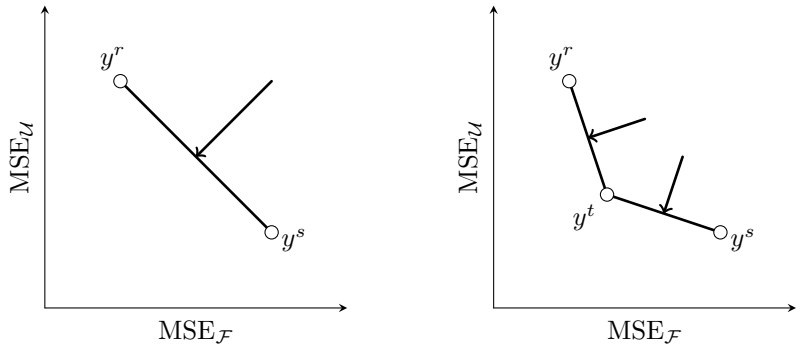


Figure 5: Mode of action of the dichotomic search. On the left: The connecting line segment between y^r and y^s is perpendicular to the vector $(\alpha, 1-\alpha)^\top$ that is here multiplied with a negative scalar to show the minimization direction. On the right: Since $(\alpha, 1-\alpha)^\top y^t < (\alpha, 1-\alpha)^\top y^r$, the new point y^t is nondominated and two new subproblems are generated. [40, p. 7] (own illustration).

Algorithm 1 summarizes the implementation of the dichotomic search that was used in this work. It is based on [20].

4. Results

This section is subdivided into three parts. In Sections 4.1 and 4.2, our PINN is validated. We use two different sets of training data for this. At first, data that do not include infections predominantly tracing back to the omicron variant, but to the alpha, beta, gamma and delta variant, are used and errors are computed. At next, exclusively data covering the autumn/winter 2021/2022 are used to train the network. So, we firstly make predictions based on a 76-week-long training data set and later on a 16-week-long training data set. Differences between the outcomes using the two training data sets are discussed. Consequently, we distinguish between training data sets covering the time since the outbreak of the pandemic in Germany, which we call *long-term training*

Algorithm 1: Dichotomic Search

Data: Training data S^d, hyperparameter settings for single-objective stochastic gradient descent (SGD) solver, depth of the search (levels), initial weighting parameters $\alpha_1, \alpha_2 \in (0,1)$, $\alpha_1 < \alpha_2$, to approximate extremal solutions focusing on $\mathrm{MSE}_{\mathcal{U}}$ and $\mathrm{MSE}_{\mathcal{F}}$, respectively (see (30))

Result: Approximation of the Pareto front and corresponding PINN parameters

$\Lambda \leftarrow \{\alpha_1, \alpha_2\}$;
`cand` $\leftarrow \emptyset$;
for $l = 1, \ldots,$ `levels` **do**
  for $\alpha \in \Lambda$ **do**
    train with weighted sum objective $\mathcal{L}_\alpha$;
    add objective vector to `cand`;
  delete all dominated points in `cand`;
  sort `cand` by second objective function (in increasing order);
  if $l <$ `levels` **then**
    for $i \in \{2, \ldots, |\texttt{cand}|\}$ **do**
      $\mathrm{diff} \leftarrow \texttt{cand}(i) - \texttt{cand}(i-1)$;
      $\Lambda \leftarrow \Lambda \cup \left\{ \frac{\text{-diff}_1}{\text{diff}_2 - \text{diff}_1} \right\}$
    resort Λ increasingly
Return nondominated points from `cand` to illustrate trade-offs;

data, and training data sets that exclusively contain data of a specific wave or even peak reached during the pandemic, which we call *short-term training data*.

For better comparability, we let the PINN identify a fixed transmission risk β in both validation subsections. We use this as a point of reference and apply a fixed transmission risk in the implementation of the NSFD scheme as well. The NSFD scheme substantially serves as a numerical method for comparison with the results of the PINN and the effect of a manual alteration of the transmission

risk is observed. It is also applied to obtain the *NSFD-induced* loss for a part of our validation runs.

In Section 4.3, we present the results of the dichotomic search to examine the influence of the weighting parameter α in the training objective (30).

4.1. Validation of Scenarios Generated with Long-Term Training Data

In this first validation part, a major target was the observance of differences resulting from distinct choices of the weighting parameters α and γ. We firstly use data reported for the calendar weeks 10 in 2020 to 32 in 2021, i.e. long-term data, in order to perform our first validation runs. We made use of the range of 76 calendar weeks because our first aim was to adapt the learning procedure of the PINN to multiple mutated variants, all seasons and different experienced scenarios of intervention. We regard the prediction error between reported and PINN-predicted infection numbers starting in the calendar week 33 in 2021.

The following validation covers the time until February 2022, but NSFD output is visualized for 32 further weeks here, since we want to include the local peaks in the plots in order to notice when they are reached and what their height is. Thus, Figure 6 shows the size of the infected compartment over 24 points in time (weeks) resulting from the PINN and given by the reported data set and for 56 points in time for the NSFD scheme with different values of the transmission risk parameter β. The prediction with the NSFD scheme in this figure corresponds to a prediction of the PINN with a *NSFD-induced* loss with parameter choice $\gamma = 1$. We use the 10^{th} calendar week in 2020 as the 1^{st} week considered, since this is the first week for which data are available. The forecast then begins in the 76^{th} week considered (calendar week 33 in 2021). The same NSFD predictions and reported data are shown in all four diagrams of Figure 6, but a distinct PINN prediction is depicted in each diagram.

We can see from Figure 6 that the actual infection rate increased with a growing slope between the 76^{th} and 90^{th} as well as 95^{th} and 100^{th} considered week. In fact, a local maximum of 693 infections per 1 million people were registered in Germany on November 28^{th} 2021, which already was the highest

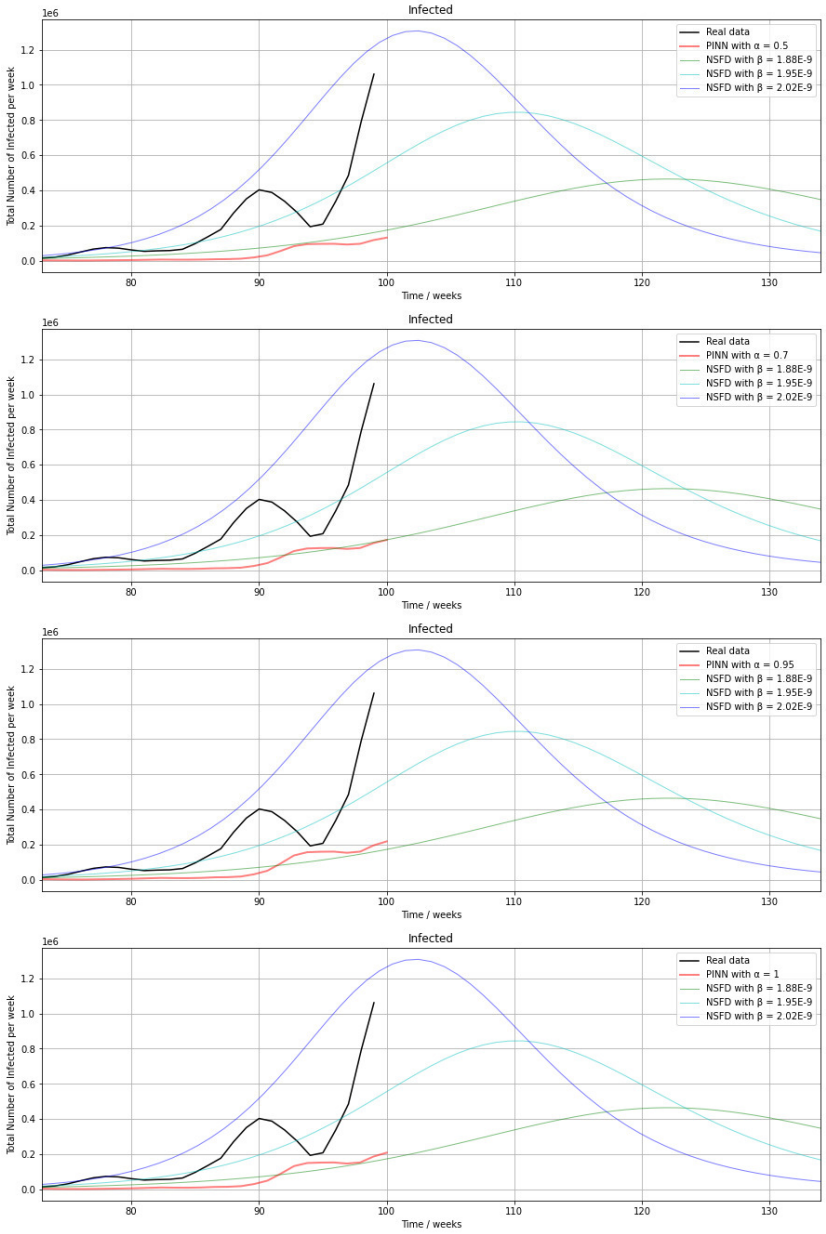


Figure 6: Per diagram: Infection numbers obtained from the reported data (black), the NSFD scheme, i.e. $\gamma = 1$, and with three different transmission risk parameter values (blue, cyan, green), and the training of the PINN with the loss terms $\text{MSE}_{\mathcal{U}}$ and $\text{MSE}_{\mathcal{F}}$ (red) using a distinct choice of the weighting parameter α ($\alpha \in [0.5, 0.9, 0.95, 1]$). The prediction starts in the 76^{th} considered week starting at August 9^{th} 2021. The used training data cover the calendar weeks 10 in 2020 to 32 in 2021 (1^{st} to 76^{th} considered week).

infection rate since March 2020. A more than 3 times as large peak with 2,434 infections per 1 million people was attained on February 14^{th} 2022 [5].

It is obvious that the strong increase of infection rates up to $1,571,595$ weekly new infections (last week of March 2022) is not predicted by the PINN if we use training data covering the time until August 2021. The omicron variant with its higher transmissibility than previous variants is not a part of the training data in this case. A maximum of 174,701 weekly new infections was registered by the RKI in calendar week 51 in 2020, i.e. during the weeks used for the training. We also remain on a level below 200,000 weekly infections in our prediction. Nevertheless, a steady increase in infection numbers from the last week of November 2021 is predicted by our PINN. Figure 6 serves to analyze the effect of different choices of the hyperparameter α on the prediction outcome. Among the four given choices of α, the value of $\alpha = 0.95$ yields the highest infection numbers and the resulting curve comes closest to the reported data here, although a good approximation of high omicron infection numbers is not obtained as explained before. Differences between the computed errors for different values for the weighting parameters α and γ can be found in Table 2.

The NSFD scheme with a constant transmission risk obviously predicts one maximum during the course of one wave. In fact, a local maximum of 1,307,475 weekly new infections was reached in the calendar week 6 in 2022 (102^{nd} considered week) and a global maximum was attained in the 108^{th} considered week (1,571,595 weekly new infections). We notice variations in the NSFD prediction if the transmission parameter β is modified. To create Figure 6, three slightly different values of β were selected such that great variations in the prediction are obtained. If we select $\beta = 1.88 \cdot 10^{-8}$, the NSFD scheme approximates the curves predicted by the PINN based on training data covering times of present alpha, beta, gamma and beginning delta variants rather than the sharp increase drawing back to the omicron variant. In contrast to that, the choice $\beta = 2.02 \cdot 10^{-9}$ allows the prediction to approach the omicron-effected sharp increase of infection rates from January 2022. Maxima with a height of $1.2 \cdot 10^6$ to $1.5 \cdot 10^6$ weekly infections between the 102^{nd} and 108^{th} considered week can

be attained with the NSFD scheme if $\beta \in [1.99 \cdot 10^{-9}, 2.03 \cdot 10^{-9}]$. We found that assigning $\beta = 1.98 \cdot 10^{-9}$ is most suited for the approximation of the peak in the 90^{th} considered week at the same time, cf. Figure 8.

We note that the inclusion of a time-varying transmission rate in the NSFD approach would facilitate the prediction of multiple peaks within a wave, as shown in [21]. For simplicity we did not include a time-dependent rate in the PINN or NSFD scheme used in this paper.

The third loss term, based on the NSFD scheme, was added to the PINN to analyze whether a reduction in the error between the reported data and the PINN outputs could be achieved by including the NSFD-induced loss term. This term is a numerically calculated part of the network and results from the system of ODEs in a different way than the residual loss. We have created a Table 2 that gives the prediction errors for different values of α and γ. Using the third loss term, we are able to compare the PINN results with the $\gamma \in [0, 1]$ parameter, i.e., we can consider a PINN without any influence of the NSFD scheme on the results or a PINN whose loss function is calculated based solely on the NSFD method.

It is striking that the computed error is distributed differently on the 27 considered calendar weeks for distinct values of γ. It lies between 0 and 200,000 during the first 23 weeks, for which validation forecasts were made. We observe that the error in the first 15 considered weeks (beginning of August until end of November 2021) and in the 20^{th} to 23^{rd} considered week (January 2022) consistently increases the larger $\gamma > 0.1$ is selected. We can see from Figure 7 that the error is nevertheless larger for $\gamma = 0.05$ than for $\gamma = 0.1$ in the first 10 considered weeks, and remains clearly below 50,000. The error is significantly smaller between the 15^{th} to 19^{th} considered week (December 2021) for a larger value of γ. Whereas a local error peak of 300,000 is reached in the 17^{th} considered week (end of September 2021) for $\gamma = 0.05$, this peak is only 100,000 for $\gamma = 0.5$. The figure implies that the inclusion of the *NSFD-induced* loss term to a larger proportion enables a better approximation of the real data at the end of the first omicron wave (late November to late December 2021). However, Fig-

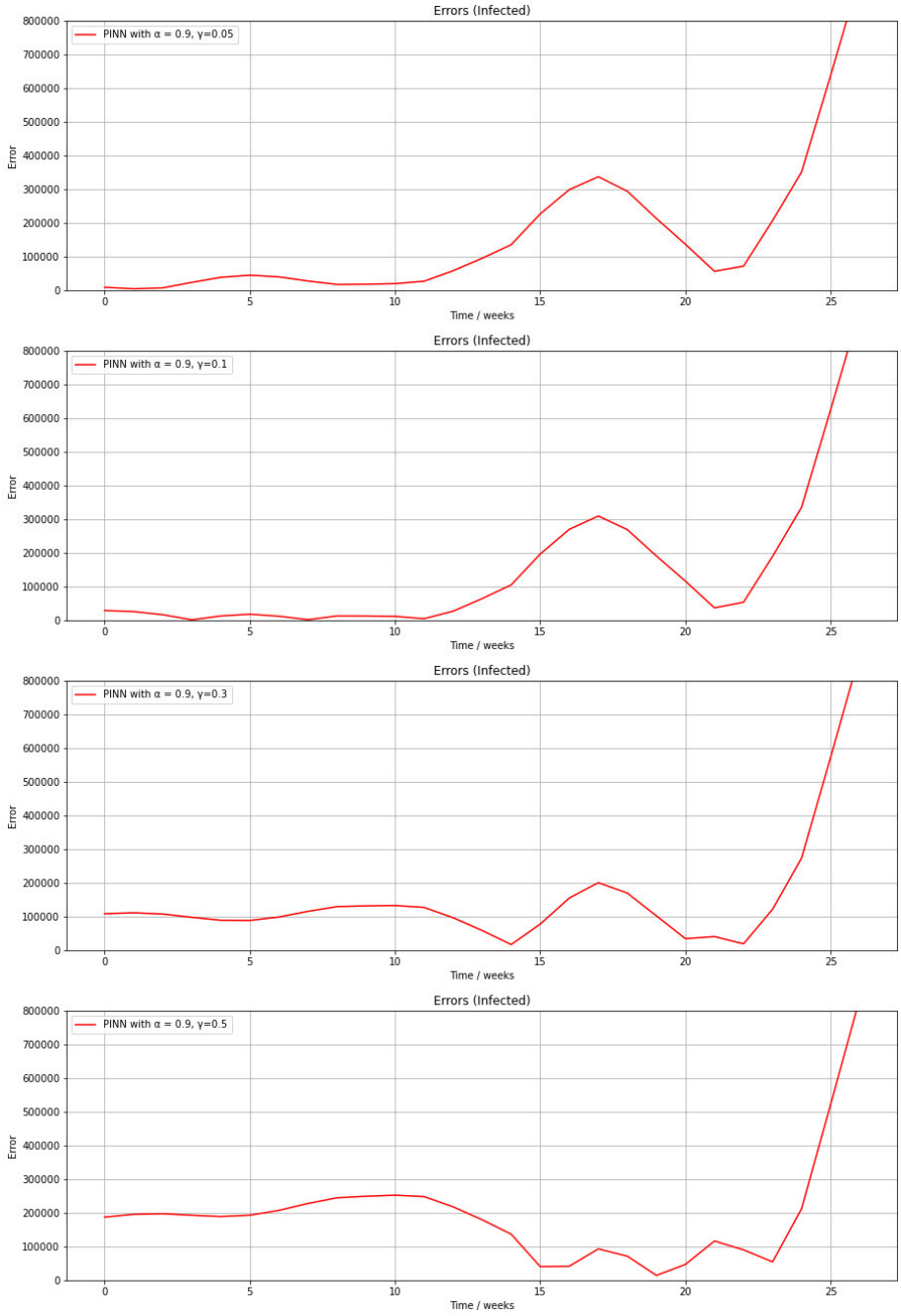


Figure 7: Per diagram: Errors between the reported data and the infection numbers obtained with the PINN with $\alpha = 0.9$ and an additional NSFD-based loss with $\gamma = \in [0.05, 0.1, 0.3, 0.5]$ and for 27 calendar weeks from August 2021.

ure 7 indicates that the hyperparameter γ should be chosen only as $\gamma = 0.1$ to obtain a very small error for August to November 2021. It is relevant to always consider the strong dependence of the performance of the NSFD scheme on the transmission parameter β. We also found that the NSFD scheme was suited for the prediction of specific peaks heights like the peak height of the first omicron wave during the course of the pandemic, as well as smooth increases of infection numbers. In contrast to that, the PINN based on the data and residual loss is able to capture smaller local peaks and oscillations.

In all cases of choices of the parameter γ, the error most sharply increases from the 22^{nd} observed week (late January 2022), which is reasoned by the second maximum during the fourth wave responsible of the highest infection numbers since the outbreak of SARS-CoV-2 in early 2020. Table 2 provides a comparison between the four choices of the parameter γ depicted in Figure 7.

To create this table, we computed the absolute difference between the reported data and compartment size output of the PINN as

$$\text{diff} = \sum_{j=1}^{k} \text{diff}_j \,, \tag{35}$$

with

$$\text{diff}_j = ||\hat{\mathcal{K}}_j - \mathcal{PINN}^{\text{W}}_{\text{p}\ j}|| \quad \text{for all } j = 1, \ldots, k. \tag{36}$$

Moreover, we computed the mean squared error as

$$\text{MSE}_{\text{diff}} = \frac{1}{k} \sum_{j=1}^{k} \text{diff}_j{}^2 \,, \tag{37}$$

where k is the number of weeks for which the output of the PINN and the reported data shall be compared.

Absolute differences and mean squared errors for chosen parameter assignments are shown in Table 2 for different values of the weighting parameters α and γ. We selected $k = 25$. The used learning rate is $lr = 0.01$.

In this case, we manually modified the values of α and γ in the implementation, such that results for selected values are shown in Table 2. Furthermore, it is important to note that we analyze errors emerging from a special scenario

value of α	**value of γ**	diff (regularized by 10^{-6})	MSE_{diff} (regularized by 10^{-11})
1	0	47.97354	41.134955
0.999	0	47.96365	41.095809
0.995	0	44.81831	33.610758
0.99	0	44.74700	33.556702
0.96	0	44.68934	33.309817
0.95	0	44.66775	33.261432
0.94	0	44.98273	33.997154
0.9	0	45.22915	034.567847
0.8	0	46.72179	37.996877
0.5	0	51.15098	49.218819
0.9	0.05	43.20369	40.192425
0.9	0.1	38.63568	34.506249
0.9	0.3	41.49101	24.350765
0.9	0.5	52.51707	33.797033
0.95	0.1	38.28079	33.624826

Table 2: Absolute differences and mean squared errors between the reported data and outputs of the PINN with training data covering the calendar weeks 10 in 2020 to 32 in 2021 if distinct weighting parameter values are used.

with a training basis of 76 weeks between March 2020 and August 2021, and a prediction covering 25 calendar weeks. For other scenarios, different ranges of errors would be obtained.

We can observe that the smallest errors, i.e. diff < 45, are reached with $\alpha \in [0.94, 0.995]$. This leads to the impression that the predictive value of the data loss in the loss function of the PINN is highly relevant, and much more significant than the residual loss. The data-drivenness of the PINN enables to capture dynamics of the infectious disease, which numerical methods cannot anticipate. Table 2 implies that the smallest error among the chosen parameter values is achieved with $\alpha = 0.95$, where diff $= 44.66775$.

Despite the remarkable significance of the data loss, we have to take into account that the parameter value $\alpha = 0.8$ leads to a smaller error than $\alpha = 1$, i.e. weighting the residual loss 20 % reduces our prediction error compared to weighting it 0 %. By including the residual term, we use systematic knowledge about the disease propagation and make our neural network a more mathematical approach. Certain parameter values can be derived from elaborate studies, knowledge about a possible next mutation, intervention or vaccination strategy can be incorporated, and SIR-type models provide clues to transition dynamics.

We can see that incorporating the *NSFD-induced* loss into the loss function can further reduce our error. The system of ODEs, which is the physical part of our PINN, is used in a different way in the establishment of the NSFD scheme than in the computation of the residual error. In our manually generated scenarios, the choice $\gamma = 0.1$ among $\gamma \in [0, 0.05, 0.1, 0.3, 0.5]$ results in the smallest errors for different selections of α. For instance, we obtain diff $= 38.28079$ with the combination $\alpha = 0.95, \gamma = 0.1$. It has to be stressed that little modifications of the transmission risk can cause big changes in NSFD predictions in terms of the slope and attained height of the reached maximum.

In order to obtain a better approximation of the strong increase before the second peak of the fourth and omicron-dominated COVID-19 wave in Germany, we extended the training data set. The extended set includes the calendar weeks 10 in 2020 to 50 in 2021, i.e. 94 weeks now. Thus, it also contains data of the

first peak during the fourth wave. We still deal with long-term training data here. Figure 8 depicts the result of the PINN prediction with $\alpha = 0.95$ and $\gamma = 0$ as well as NSFD predictions, which correspond to PINN predictions with $\gamma = 1$. We can see in Figure 8 that the height and point in time of the local

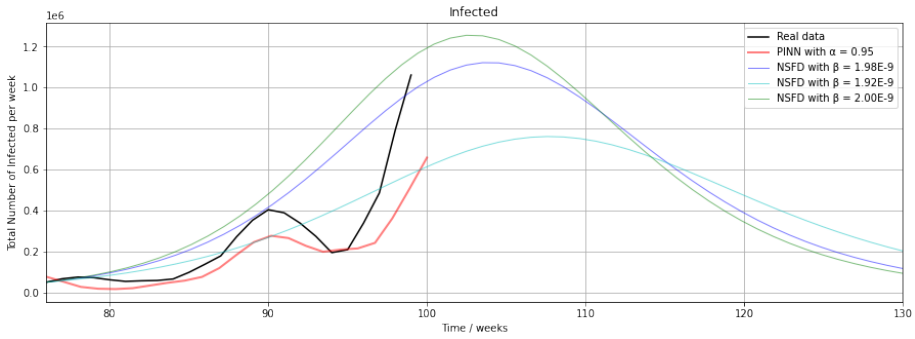


Figure 8: Infection numbers obtained from the reported data (black), the training of the PINN with the loss terms $\text{MSE}_{\mathcal{U}}$ and $\text{MSE}_{\mathcal{F}}$ (red) with $\alpha = 0.95$ and $\gamma = 0$, or the NSFD scheme, i.e. $\gamma = 1$, with distinct transmission risk parameter values (blue, cyan, green) from the 77^{th} considered week starting at August 16^{th} 2021. The used training data covers the calendar weeks 10 in 2020 to 50 in 2021 (1^{st} to 94^{th} considered week).

maximum in the 90^{th} considered week is approximated much better than in the diagrams of Figure 6. The local minimum of 193,939 weekly infections is captured by the PINN. The following strong increase before the second peak during the fourth wave is also predicted by the PINN, although the predicted increase is less sharp than it is in the reported data. Taking into account the 77^{th} to 100^{th} considered week and $\alpha = 0.95$, the computed mean squared error is $\text{MSE}_{\text{diff}} = 17.36 \cdot 10^9$. The implementation with the larger training data set substantiates our assumption that the inclusion of specific virus variant-influenced compartment size data has a great effect on the resulting predicted scenario. It has to be noted that the training data has a substantial impact on the inferred transmission rate, which again influences the system of ODEs used for residual loss computation in every single training step.

4.2. Validation of Scenarios Generated with Short-Term Training Data

Aside from the above validation results, which serve to analyze the influence of the two weighting parameters α and γ, we want to analyze the potential improvement of our predictions if data of infection rates at times of a dominant omicron variant are used as a part of the data basis. In fact, we want to make a short-term prediction. The question is whether the high infection rates of February to April 2022 tracing back to the omicron variant are foreseeable with our PINN approach. Since in this case, we want to focus on the trade-off between data loss and residual loss, we set $\gamma = 0$, i.e., we do not include the NSFD scheme. The underlying data set is reduced in the way that only data of the calendar weeks 40 in 2021 to 2 in 2022 are given as an input to the neural network. This means that the calendar weeks 10 in 2020 to 39 in 2021 do not serve as a training basis here, and hence the training data set is reduced by 80 % in size. It is important to note that data covering the increase in infection rates due to the omicron spread must be included in the training data to be able to predict the further increase. The PINN adapts its trained parameters to the underlying data set.

Figure 9 illustrates the results of this additional analysis. It shows three prediction curves emerging from a short-term prediction until the calendar week 8 in 2022 based on training data of the weeks 40 in 2021 to 2 in 2022 and using different values for the weighting parameter α.

It should be noted that the calendar weeks 40 to 43 are not visualized here. The compartment size data of the calendar week 40 are given as an initial condition $\hat{\mathcal{K}}^0$ and direct input to the PINN. This explains the small gap between the red and black curves in the very beginning of the predictions in Figure 9.

We observe that the first actual peak in 2022 (1,307,475 weekly infections in calendar week 6) as well as the local minimum of 193,939 weekly infections in calendar week 51 in 2021 are both recognized if $\alpha = 0.95$ is selected. In the case $\alpha = 0.9$, the local minimum of the 95^{th} considered week is captured as well, and the gap between the prediction and reported data is $1.2 \cdot 10^5$ in the 101^{st} considered week. The error between the reported and predicted data between

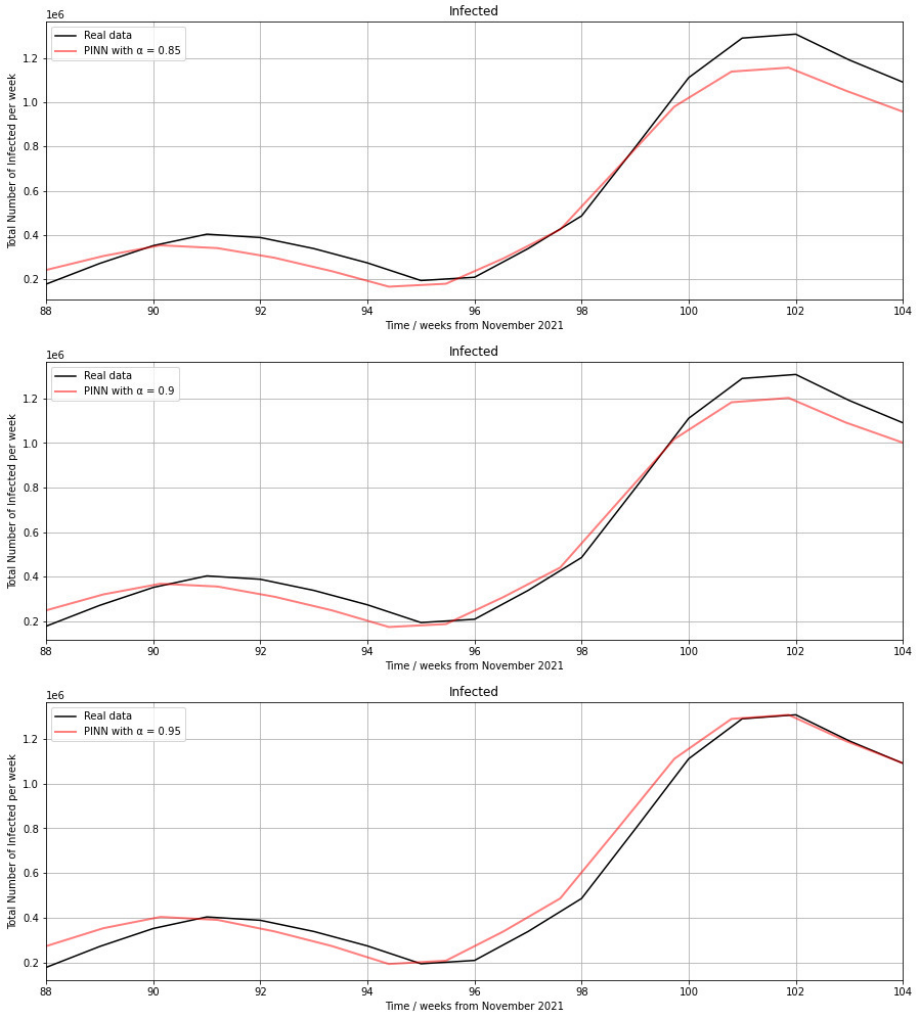


Figure 9: Infection numbers obtained from the reported data (black) or the training of the PINN with the loss terms $\text{MSE}_{\mathcal{U}}$ and $\text{MSE}_{\mathcal{F}}$ (red) from the 88^{th} considered week in November 2021 for three different assignments of the weighting parameter α ($\alpha = 0.85, 0.9, 0.95$) and $\gamma = 0$. The training data set covers the calendar weeks 40 in 2021 to 2 in 2022 (84^{th} to 98^{th} considered week).

the 96^{th} and 101^{st} considered week is smaller for $\alpha = 0.9$ than for $\alpha = 0.95$. The specific errors are shown in Table 3. The performance of the prediction with $\alpha = 0.85$ (upper diagram) is less satisfying than the one with $\alpha = 0.9$ in terms of the gaps of $1.5 \cdot 10^5$ in the 101^{st} and $1.7 \cdot 10^5$ in the 102^{nd} considered week. However, the increase in the reported data between the calendar weeks 52 and

5 is accounted for best among the three choices of the weighting parameter.

In comparison with the results shown in Figure 6, the sharp increase of, on average, 75,305 weekly registered infections between the calendar week 44 in 2021 to 6 in 2022 is captured in the middle and lower diagram of Figure 9. This stresses the strong dependence of the performance of our PINN on the training data set. The beginning of the mentioned increase is included in the training data set of the prediction visualized in Figure 9. Our result here is that short-term predictions should be favoured if we are interested in precise predictions.

The respective computational errors for different values of the weighting parameter α and for $k = 16$ considered points in time are shown in Table 3. To create Table 3, we manually modified the weighting parameter $\alpha \in (0, 1)$. Compared to Table 2, we have much smaller error values, which draws attention on the better performance of the PINN for short-term predictions.

It is remarkable that the smallest error is achieved with $\alpha = 0.94$ (among the considered parameter values.) With $\alpha = 0.95$, the error diff becomes 1.82 times as large and with $\alpha = 0.93$ it becomes 2.23 times as large. Note, however, that the individual training runs do not necessarily terminate with a (globally) optimal solution; so the reported error values can only approximate the best possible error for the respective choices of weighting parameters. Nevertheless, we can observe that diff is approximately twice and MSE_{diff} more than four times as large for $\alpha = 0.9$ as for $\alpha = 0.99$ in the considered scenario. Clear reductions in the performance of the network are observable when setting $\alpha < 0.9$. For instance, we obtain an approximate duplication of diff if the weighting parameter α is decreased from $\alpha = 0.91$ to $\alpha = 0.85$, and almost a tripling if it is decreased from $\alpha = 0.91$ to $\alpha = 0.65$ or from $\alpha = 0.65$ to $\alpha = 0.1$, and a 2.2-fold increase if the parameter is decreased from $\alpha = 0.65$ to $\alpha = 0.3$. Those two aspects stress the significance of the data loss even more than in the previously considered prediction with a larger training data set.

The errors for $\alpha = 0.92$ are nevertheless smaller than for $\alpha = 0.99$. Completely neglecting the residual loss still does not yield the best performance of our

value of α	diff (regularized by 10^{-3})	MSE_{diff} (regularized by 10^{-9})
0.1	8,782.944	427.6832
0.3	6,469.858	230.0317
0.5	4,396.296	105.3495
0.65	2,969.012	91.7439
0.85	1,217.442	79.6116
0.9	804.601	34.6124
0.91	554.790	16.4997
0.92	391.053	9.8716
0.93	175.823	5.2877
0.94	78.409	0.3301
0.95	142.195	1.8924
0.96	237.102	3.0019
0.97	344.892	6.3855
0.99	401.915	7.8172

Table 3: Absolute and mean squared errors between the reported data and outputs of the PINN with training data covering the calendar weeks 40 in 2021 to 3 in 2022, depending on the weighting parameter α.

PINN. Although no mathematical model can optimally describe an infectious disease as not every epidemiological detail relevant to transmission is known, we can incorporate systematic knowledge about the spread and transmission dynamics of the disease into our neural network with the aid of the residual loss. Since $\alpha = 1$ or $\alpha = 0.99$ does not yield the smallest errors in our validation runs, the inclusion of the residual error remains warrantable and reasonable. It enables to trade between the traditional mathematical approach and newer data-driven method. In general, predictions of future pandemic courses are connected to several uncertainties owing to unknown mutant variants, changing intervention measures or population compliance and new vaccination strategies.

Apart from this, we note much larger differences between the errors for choices of $\alpha \geq 0.9$ and $\alpha < 0.9$ than in Table 2. While we have only an increase of 13 % in diff and 48 % in MSE_{diff} if α is decreased from 0.9 to 0.5 in Table 2, we can see a 5.47-fold increase in diff and a tripling of MSE_{diff} for the same alteration of the weighting parameter in the implemented short-term prediction. This emphasizes the larger sensitivity to weighting parameter modification of the prediction with a training data set covering only one wave over 16 weeks compared to the forecast with a training data set covering 76 weeks. Our results imply that the short-term prediction yields better results in terms of the approximation of omicron-effected high infection rates. This is recognizable in the extremely large differences in diff and MSE_{diff}.

The above discussion shows that the choice of the weighting parameter α plays an important role in the prediction quality. Note that the prediction quality here is measured by comparisons with the real data. It is therefore not surprising that the prediction quality is higher when the data loss $\text{MSE}_{\mathcal{U}}$ is highly weighted in the weighted sum training objective (30). However, the results also show that the residual loss $\text{MSE}_{\mathcal{F}}$ should not be ignored. This motivates a more detailed analysis of the trade-off between data loss and residual loss using dichotomous search in the following section.

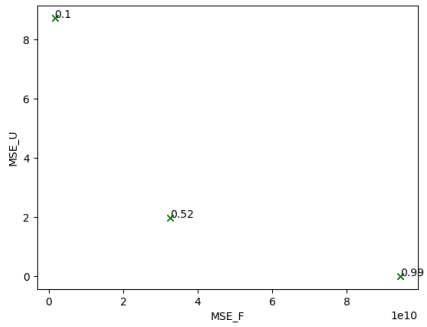
4.3. Results of the Dichotomic Search

In this section, we present the results of dichotomic search using Algorithm 1. Based on the discussion in the previous sections, we focus on short-term training data from calendar week 40 in 2021 to 2 in 2022 (84^{th} to 98^{th} considered week). See Section 4.2 and Figure 9 for comparison. Four levels of Pareto front approximations (based on three, five, nine and 17 training runs, respectively) are shown in in Figure 10. Each training run was performed with 5000 epochs and the ADAM optimizer with a fixed learning rate of 0.003. The search was initiated with $\alpha_1 = 0.1$ and $\alpha_2 = 0.99$. Other search windows can be used depending on preferences or if additional information is available. In addition, the search window can be used to zoom into a specific part of the Pareto front.

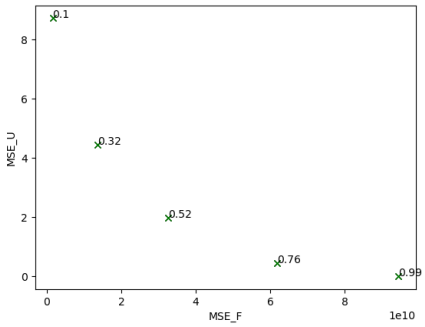
The results confirm that a pronounced Pareto front with a diverse set of outcome vectors was approximated after only a few training runs. Indeed, level 2 based on five training runs (see Figure 10(b)) already provides a rough approximation of the Pareto front. At level 3 (Figure 10(c)), after nine training runs, we already have a very good approximation of the Pareto front. The final level 4 is based on 17 training runs (Figure 10(d)). On a machine with an AMD Ryzen 7 3700X 8-Core Processor and NVIDIA GeForce RTX 3060 graphic card, each training run took about 3 minutes, so the final level needed a computation time of about 51 minutes. The computation time of each training run is of course dependent on the specific problem and network, however, the goal here was to show that one can gain insight into the complete Pareto front of the problem with rather few training runs.

This shows that the dichotomic search scheme was successful in obtaining a diverse set of solutions with only very few training runs necessary, allowing the decision maker insight into the effect of changing the weighting parameter α that reflects the importance of each loss function.

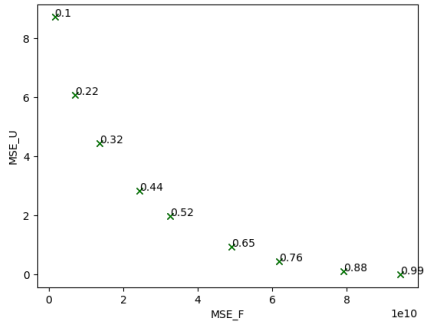
Figure 11 shows a dichotomic search with starting weights $\alpha_1 = 0.8$ and $\alpha_2 = 0.99$ in order to zoom into the part of the Pareto front that emphasizes the data loss $\text{MSE}_{\mathcal{U}}$. This is motivated by the results presented in Section 4.2 for the same short-term data that indicated that the best prediction results



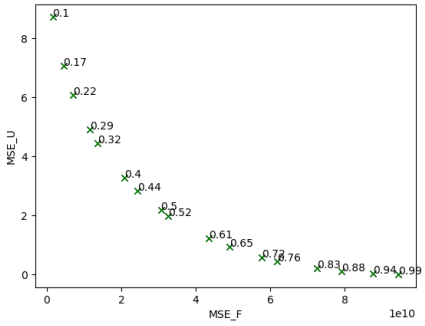
(a) Level 1: Lexicographic optimization and one weighted sum (three training runs)



(b) Level 2: Five training runs



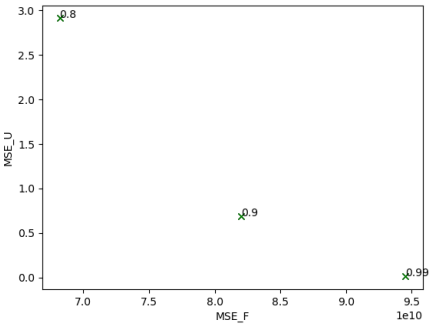
(c) Level 3: A pronounced Pareto Front is forming



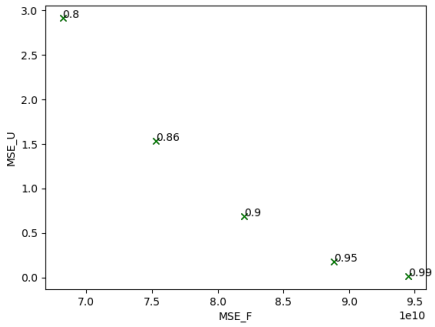
(d) Level 4: Final Level with 17 training runs

Figure 10: Pareto front approximation using dichotomic search with a learning rate of 0.003 with starting weights $\alpha_1 = 0.1$ and $\alpha_2 = 0.99$. Favorable trade-offs can be identified starting from Level 3.

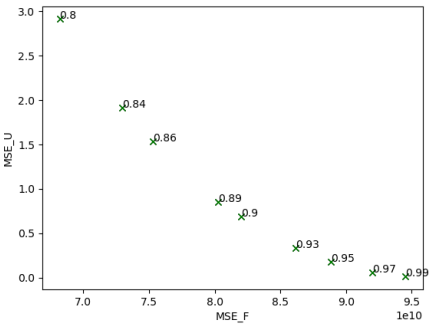
are obtained for weighting parameters $\alpha \approx 0.94$. Starting from level 3 (see Figure 11(c)) the additional gain wrt. the data loss $\text{MSE}_{\mathcal{U}}$ is only marginal when we increase the weighting parameter α beyond 0.93. This is in accordance with the fact that the prediction accuracy, that relates not only to the prior data but also to the SVIHR model, does not improve beyond $\alpha \approx 0.94$.



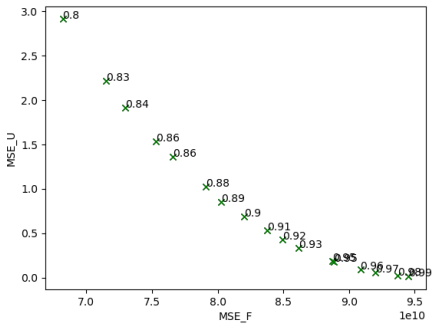
(a) Level 1: Lexicographic optimization and one weighted sum (three training runs)



(b) Level 2: Five training runs



(c) Level 3: $\text{MSE}_{\mathcal{U}}$ only improves marginally beyond $\alpha = 0.93$



(d) Level 4: Final Level with 17 training runs

Figure 11: Pareto front approximation using dichotomic search with a learning rate of 0.003 with starting weights $\alpha_1 = 0.8$ and $\alpha_2 = 0.99$.

Overall, it can be seen that the variation of α has a large impact on the relative importance of data loss and residual loss. The results of the previous sections show that a high prediction accuracy for weighting parameters α in the

range of 0.95 is achieved. This is due to the fact that the prediction accuracy was assessed by comparison with the real data and not with the model predictions. In other applications, different preferences may lead to a different preferred choices of α.

5. Conclusion and Outlook

We considered the *physics-informed neural networks* (PINNs) method, a Deep Learning technique in which the weighted loss function consists of a data loss and a residual loss. In our approach, the data loss results from the sizes of the compartments of an established *susceptible-vaccinated-infected-hospitalized-recovered* (SVIHR) model derived from a training set of reported COVID-19 data. The residual loss is derived from a system of ODEs based on the proposed SVIHR model, which mathematically describes the dynamics of transitions between different compartments and infectious disease transmission in a population affected by the COVID-19 pandemic. We proposed a NSFD scheme especially designed for the solution of the SVIHR model. Its solution leads to a straight-forward extension of the used loss function, also to measure the s

Our results for predicting the COVID-19 pandemic show large differences when using short-term data covering 16 weeks compared to long-term data covering a 76-week interval. Our 76-week training data set does not include the fourth wave, i.e., autumn 2021, winter 2021/2022, or spring 2022, when the Omicron variant spread and three spikes were observed. Our numerical results indicate that accurate predictions would require the consideration of adaptive and time-dependent transmission rates. With fixed transmission rates, however, the PINN was not able to anticipate the strong increases in November/December 2021 and early 2022. When the beginning of the first local maximum of the fourth COVID-19 wave, that was reached in November 2021, is included in the training data we obtain a better prediction of the steep increase of infection numbers in early 2022. Alternatively, we considered shorter data sets to predict individual waves. In this case, the prediction based on the 16-week training

dataset covering only the autumn/winter of 2021/2022 performs very well in approximating the second peak during the fourth wave.

Finally, we considered PINN training from a biobjective perspective to better analyze the trade-off between data loss and residual loss. We performed a dichotomic search that successfully approximated the Pareto front and thus found a large number of solutions (i.e., trained networks) with comparatively few training runs. We found that the preferred values of the weighting parameter α in the total loss function were greater than 0.9 in most cases, thus giving greater weight to data loss than to residual loss. This could indicate that one should further improve the model. We find that the biobjective approach to PINN training can be applied to PINNs regardless of the number of loss terms or application field.

In future work, time-varying functions for vaccination and transmission rates will be included to account for seasonal and variant-dependent fluctuations, cf. [43]. Time-variability in the transmission rate should especially be taken into consideration with respect to long-term predictions, in which the training data set covers multiple months and thus includes different mutations, lockdown politics and seasons. An adaptive learning rate scheme will be applied and compared with the fixed learning rate in terms of the prediction accuracy achieved, cf. [44]. It is worth noting that we only dealt with two loss terms here and therefore used a biobjective optimization model. The concept of multiobjective PINN training can easily be extended to consider more than two objective functions. This occurs in more advanced applications such as solving PDEs by PINNs, e.g., using gradient-enhanced PINNs (gPINNs) [45], in which case the number of loss terms is greater than two and a multiobjective approach is required. For example, in [45, Section 2.2], the number of loss terms is $3 + d$, where d is the dimension of the spatial domain. Here, one has to identify similar and conflicting training objectives in order to avoid an overly large parameter set. This extension to multiobjective optimization approaches will be the content of a follow-up article.

References

[1] M. Raissi, P. Perdikaris, G. Karniadakis, Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations, J. Comput. Phys. 378 (2019) 686–707. doi:10.1016/j.jcp.2018.10.045.

[2] J. Hoffer, A. Ofner, F. Rohrhofer, et al., Theory-inspired machine learning – towards a synergy between knowledge and data, Weld World (2022). doi:10.1007/s40194-022-01270-z.

[3] Association of researching pharmaceutical companies, last access: March 29, 2022 (2022). [link].
URL https://www.vfa.de/de/arzneimittel-forschung/coronavirus/impfstoffe-gegen-sars-cov-2-varianten

[4] Robert Koch-Institute, last access: May 6, 2022 (2022). [link].
URL https://www.rki.de/DE/Content/InfAZ/N/Neuartiges_Coronavirus/Virologische_Basisdaten.html;jsessionid=960297181B52EB351E833DD09CD96CAB.internet081?nn=13490888#doc14716546bodyText6

[5] Ourworldindata.de, last access: May 5, 2022 (2022). [link].
URL https://ourworldindata.org/coronavirus

[6] J. Malinzi, S. Gwebu, S. Motsa, Determining COVID-19 dynamics using physics informed neural networks, Axioms 11 (3) (2022) 121. doi:10.3390/axioms11030121.

[7] E. Kharazmi, M. Cai, X. Zheng, G. Lin, G. Karniadakis, Identifiability and predictability of integer- and fractional-order epidemiological models using physics-informed neural networks, Nature Comput. Sci. 1 (11) (2021) 744–753. doi:10.1101/2021.04.05.21254919.

[8] G. Pang, L. Lu, G. E. Karniadakis, fPINNs: fractional physics-informed neural networks, SIAM J. Sci. Comput. 41 (4) (2019) A2603–A2626. doi:10.1137/18M1229845.

[9] M. Cai, G. E. Karniadakis, C. Li, Fractional SEIR model and data-driven predictions of COVID-19 dynamics of omicron variant, arXiv preprint arXiv:2205.11379 (May 2022).

[10] F. Rohrhofer, S. Posch, B. Geiger, On the Pareto front of physics-informed neural networks, arXiv preprint 2105.00862 (2021).

[11] X. Jin, S. Cai, H. Li, G. E. Karniadakis, NSFnets (Navier-Stokes flow nets): Physics-informed neural networks for the incompressible Navier-Stokes equations, J. Comput. Phys. 426 (2021) 109951. doi:10.1016/j.jcp.2020.109951.

[12] S. Wang, Y. Teng, P. Perdikaris, Understanding and mitigating gradient flow pathologies in physics-informed neural networks, SIAM J. Sci. Comput. 43 (5) (2021) A3055–A3081. doi:10.1137/20M1318043.

[13] S. Maddu, D. Sturm, C. Müller, I. Sbalzarini, Inverse Dirichlet weighting enables reliable training of physics informed neural networks, Machine Learning: Sci. Techn. 3 (2021) 015026. doi:10.1088/2632-2153/ac3712.

[14] J.-A. Désidéri, Multiple-gradient descent algorithm (MGDA) for multiobjective optimization, Compt. Rend. Math. 350 (2012) 313–318. doi:10.1016/j.crma.2012.03.014.

[15] J. Fliege, B. Svaiter, Steepest descent methods for multicriteria optimization, Math. Meth. Oper. Res. 51 (3) (2000) 479–494. doi:10.1007/s001860000043.

[16] O. Sener, V. Koltun, Multi-task learning as multi-objective optimization, CoRR (2018). arXiv:1810.04650v2.

[17] S. Liu, L. Vicente, The stochastic multi-gradient algorithm for multi-objective optimization and its application to supervised machine learning, Annals Oper. Res. (2021). doi:10.1007/s10479-021-04033-z.

[18] L. McClenny, U. Braga-Neto, Self-adaptive physics-informed neural networks using a soft attention mechanism, arXiv preprint 2009.04544 (2020).

[19] A. F. Psaros, K. Kawaguchi, G. E. Karniadakis, Meta-learning PINN loss functions, J. Comput. Phys. 458 (2022) 111121. doi:10.1016/j.jcp.2022.111121.

[20] M. Reiners, K. Klamroth, F. Heldmann, M. Stiglmayr, Efficient and sparse neural networks by pruning weights in a multiobjective learning approach, Comput. Oper. Res. 141 (2022) 105676. doi:10.1016/j.cor.2021.105676.

[21] S. Treibert, M. Ehrhardt, A physics-informed neural network to model COVID-19 infection and hospitalization scenarios, IMACM preprint 22/11, University of Wuppertal, Germany (May 2022).

[22] W. Kermack, A. McKendrick, Contributions to the mathematical theory of epidemics, Bull. Math. Bio. 53 (1) (1991) 700–721. doi:10.1016/S0092-8240(05)80040-0.

[23] Robert Koch-Institute, last access: March 29, 2022 (2022). [link].
URL https://www.rki.de/DE/Content/InfAZ/N/Neuartiges_Coronavirus/Daten/Fallzahlen_Kum_Tab.html

[24] Robert Koch-Institute, last access: March 29, 2022 (2022). [link].
URL https://www.rki.de/DE/Content/InfAZ/N/Neuartiges_Coronavirus/Daten/Klinische_Aspekte.html

[25] COVID-19 Vaccination Dashboard, last access: May 6, 2022 (2022). [link].
URL https://impfdashboard.de/daten

[26] S. Treibert, H. Brunner, M. Ehrhardt, A nonstandard finite difference scheme for the SVICDR model to predict COVID-19 dynamics, Math. Biosci. Engrg. 19 (2022) 1213–1238. doi:10.3934/mbe.2022056.

[27] M. Martcheva, An Introduction to Mathematical Epidemiology, 1st Edition, Springer, 2015. doi:10.1007/978-1-4899-7612-3.

[28] Robert Koch-Institute, last access: March 29, 2022 (2022). [link].
URL https://www.rki.de/DE/Content/InfAZ/N/Neuartiges_Coronavirus/Steckbrief.html

[29] Federal Statistical Office, Germany, last access:January 10, 2022 (2022). [link].
URL https://www.destatis.de/DE/Themen/Gesellschaft-Umwelt/Bevoelkerung/Bevoelkerungsstand/_inhalt.html

[30] R. E. Mickens, Exact solutions to a finite-difference model of a nonlinear reaction-advection equation: Implications for numerical analysis, J. Diff. Eqs. Appl. 9 (11) (2003) 313–325. doi:10.1080/1023619031000146959.

[31] R. E. Mickens, Applications of nonstandard finite difference schemes, World Scientific (2000).

[32] G. E. Karniadakis, Y. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, L. Yang, Physics-informed machine learning, Nat. Rev. Phys. 3 (2021) 422–440. doi:10.1038/s42254-021-00314-5.

[33] S. Cuomo, V. S. di Cola, F. Giampaolo, G. Rozza, M. Raissi, F. Piccialli, Scientific machine learning through physics-informed neural networks: Where we are and what's next (2022). doi:10.48550/ARXIV.2201.05624.

[34] K. D. Olumoyin, A. Q. M. Khaliq, K. M. Furati, Data-driven deep-learning algorithm for asymptomatic COVID-19 model with varying mitigation measures and transmission rate, Epidemiologia 2 (4) (2021). doi:10.3390/epidemiologia2040033.

[35] S. Shaier, M. Raissi, P. Seshaiyer, Data-driven approaches for predicting spread of infectious diseases through dinns: Disease informed neural networks (2021). doi:10.48550/ARXIV.2110.05445.

[36] V. Grimm, A. Heinlein, A. Klawonn, M. Lanser, J. Weber, Estimating the time-dependent contact rate of SIR and SEIR models in mathematical epidemiology using physics-informed neural networks, Electr. Trans. Numer. Anal. 56 (2022) 1–27.

[37] A. G. Baydin, B. A. Pearlmutter, A. A. Radul, J. M. Siskind, Automatic differentiation in machine learning: a survey, J. Mach. Learn. Res. 18 (2017) 5595–5637.

[38] S. Ruder, An overview of gradient descent optimization algorithms (2016). doi:10.48550/ARXIV.1609.04747.

[39] M. Ehrgott, Multicriteria Optimization, 2nd Edition, Springer, 2005.

[40] A. Przybylski, K. Klamroth, R. Lacour, A simple and efficient dichotomic search algorithm for multi-objective mixed integer linear programs, arXiv preprint 1911.08937 (2019).

[41] I. Das, J. Dennis, A closer look at drawbacks of minimizing weighted sums of objectives for Pareto set generation in multicriteria optimization problems, Struct. Optim. 14 (1997) 63–69. doi:10.1007/BF01197559.

[42] K. Klamroth, K. Miettinen, Integrating approximation and interactive decision making in multicriteria optimization, Operations Research 56 (2008) 222–234.

[43] M. Jagan, M. deJonge, O. Krylova, D. Earn, Fast estimation of time-varying infectious disease transmission rates, PLoS Comput. Biol. 16 (9) (2020) e1008124. doi:10.1371/journal.pcbi.1008124.

[44] A. D. Jagtap, K. Kawaguchi, G. E. Karniadakis, Adaptive activation functions accelerate convergence in deep and physics-informed neural networks, J. Comput. Phys. 404 (2020) 109136. doi:10.1016/j.jcp.2019.109136.

[45] J. Yu, L. Lu, X. Meng, G. Karniadakis, Gradient-enhanced physics-informed neural networks for forward and inverse PDE problems, Comput. Meth. Appl. Mech. Engrg. 393 (2022) 114823. doi:10.1016/j.cma.2022.114823.