Using Physics-Informed Machine Learning to Optimize 3D Printing Processes

Benjamin Uhrich[1(✉)], Martin Schäfer[2], Oliver Theile[2], and Erhard Rahm[1]

[1] Center for Scalable Data Analytics and Artificial Intelligence, Department of Mathematics and Computer Science, Leipzig University, Leipzig, Germany
{uhrich,rahm}@informatik.uni-leipzig.de

[2] Research in Coatings and Additive Manufacturing, SIEMENS AG Technology Berlin, Berlin, Germany
{martin.schaefer,oliver.theile}@siemens.com

Abstract. In this work, we present the design and evaluation of a physics-informed machine learning (ML) approach for 3D printing of metal components based on real experimental measurement data. For this purpose, different thermal processes during the manufacturing of different metal components are modelled and solved in space and time using physics-informed neural networks (PINNs), with special attention to the geometric domain of the heat source. In this way, a digital twin is created as a simulation model of the thermal process with fixed input parameters, which represents the physical behavior and allows interpretable conclusions to be drawn with regard to the temperature distribution. The presented approach includes discrete and continuous models that are compared with a numerical solution method (finite difference method, FDM) for partial differential equations (PDEs) and real measurement data from Siemens. In addition, learning is accelerated by designing the loss function with information about initial and boundary conditions (Neumann and Dirichlet) and the heat source. This approach does not require discretization and is robust to limited noisy data. The results have confirmed the advantages of PINNs for data-driven discovery of the heat equation and have shown that the temperature distribution during additive manufacturing (AM) processes can be estimated and predicted very well.

Keywords: data-driven modelling · physics-informed neural networks · machine learning · partial differential equation · digital twin

1 Introduction

Heating and cooling procedures during AM such as 3D printing of metal components have a significant impact on production quality. To optimize such production processes, it is essential to understand the complex physical interactions between manufacturing steps, material properties and the impact of different configurations on the temperature distribution. An exhaustive evaluation of different production configurations, resulting sensor data and the generated components is overly time consuming and costly. Instead a data-driven machine learning (ML) model of the production process can be developed

J. O. Correia Vasco et al. (Eds.): ProDPM 2021, STAM, pp. 206–221, 2023.
https://doi.org/10.1007/978-3-031-33890-8_18

for analysis and optimization. It is possible to incorporate mathematical formulations of the physical processes (PDEs) into such ML models to decipher underlying physical properties and to improve prediction accuracy with limited available data. The resulting hybrid model can thus combine physical interpretability and extrapolation capacity of PDEs with data-driven ML capabilities of deep neural networks. In order to efficiently predict interpretable print defects, machine learning methods are required that are trained on large sensor data streams. In particular, if interventions in the printing process are to be made in response to these printing errors, the cause of the error must be known. This requires sufficient training data containing these errors. With training on a small amount of sensor data, errors can possibly be predicted but the cause remains unknown, so that logical measures cannot be derived. It is precisely this problem that is to be tackled in the following with PINNs. Even though there is already work where PDEs are approximately solved with physics-informed neural networks, we first want to use PINNs to discover models that identify a digital heat twin as a simulation model for a fixed configuration in AM. In a further step, these discovered models (PDEs) will be solved with PINNs to extrapolate and thus predict temperature measurements. Second, this work compares different ways of modeling with physically informed deep learning in AM. In this paper advantages and disadvantages of different physics-informed deep learning approaches are discussed and illustrated by the results. These learned models provide efficient ways to optimize 3D printing processes while reducing costly experiments for data generation. In particular, the problem of overheating of metal components can be solved. To implement the project of modeling terms and parameters of the heat equation on a real-world use case, deep neural networks were used as approximate basis functions. This paper is divided as follows. In Sect. 2, additive manufacturing steps and their potential are described. Section 3 presents related work that serves as a basis for this work. Section 4 presents the various steps and approaches necessary to model the thermal processes in AM. This includes the formulation of the PDEs for the physical laws, design of the individual loss functions, and the formulation of the initial and boundary value problems to uniquely solve the models found. Section 5 shows the application of the previously described approaches to the real measurement data of SIEMENS. There, the sensor data generation and other physical peculiarities during the printing process are also discussed. Apart from that, Sect. 5 models the heat source and determines the exposure area using the temperature gradient. Finally, the results of the individual approaches are validated and discussed with the help of the measurement data and the FDM. Section 6 presents a summary of the work and shows possible perspectives for further work.

2 Additive Manufacturing

AM, such as 3D printing of metal components, is a process with great potential. Generally, the components are assembled by joining materials layer by layer in order to be able to produce complex geometries. Almost all types of non- standard structures and irregular shapes can be created directly from digital computer-aided design (CAD) models. AM reduces the need of many conventional machining steps. It is widely assumed that further development and exploration of 3D-printing will have a revolutionary impact in

automotive, aerospace and biomedical industry. The applications range from artificial implants in medicine to complex shapes such as turbine blades in aircraft manufacturing. These components require good mechanical properties, which depend on several influencing factors during the manufacturing process. [1–3]. Laser assisted selective laser melting (SLM) uses a high-intensity laser (energy source) to melt and fuse selective areas of the powder bed. Heating and cooling procedures result from these melting processes. Several variables such as laser power, scanning speed, layer thickness and material used influence these thermal processes and are decisive for the product quality. Trial and error is often used to determine the best configuration parameters for optimized structural product properties. It is usually expensive and time-consuming without any scientific evidence. [2, 4].

3 Related Work

3.1 Numerical Simulation

To optimize production processes such as 3D printing, computational methods offer a powerful tool that has long been available. When simulating physical interactions, this tool has become an indispensable part of engineering development. Important components of numerical simulations include PDEs as a popular and well-researched formulation option of physical processes. The most well-known methods in numerical simulation are the finite difference method (FDM), the finite volume method (FVM), and the finite element method (FEM). We are grateful for the research and pioneering work in the field of numerics and simulation of PDEs, especially in the context of AM. The mathematical, physical and engineering community has already made it possible in recent years to better understand complex processes such as heat exchange and residual stresses in components. Relevant work that should definitely be mentioned here is T. Mukheerjee *et al.* with the development of a 3D heat and fluid flow model for powder bed melting [5, 6]. Wessels *et al.* demonstrate a stabilized optimal transport mesh-free method for analyzing metal particle fusion in AM [7]. Zhang *et al.* developed several numerical methods to simulate microstructural evolution in metals and alloys [8]. Cao *et al.* simulated how the influence of lay-up powder affects single-pass formation in selective laser melting [9]. Razavykia *et al.* gives a very good summary of numerical simulations of SLM [10]. Further work on the fundamental research of PDEs and numerical methods that made this work possible in the application of AM in the first place should also be mentioned here. Crank and Nicolson first published a method for numerical evaluation of solutions to partial differential equations of the heat conduction type in 1947 [11]. Narasimhan and Witherspoon of Lawrence Berkeley National Laboratory described as early as 1975 how the finite difference method can be used to analyze fluid flow in porous media [12].

3.2 Machine Learning

In order to optimize production processes in industry, prediction models based on deep learning methods are developed in addition to classical numerical simulation approaches. In recent years, the number of research milestones in ML and artificial intelligence (AI) has increased rapidly. Deep learning has become indispensable in many fields. In medicine, MRI or X-ray images are analyzed using machine learning methods [13, 14]. Another need for machine learning is object recognition in the development of autonomous driving systems [15]. Natural language processing and speech recognition have only become state of the art through the use of deep neural networks (DNNs) [16]. This development has led to machine learning methods entering various areas of industrial production. Sensor data and ML analysis methods can be used to optimize production processes and product quality. Optimizations with regard to the production process can be shorter machine times or material and energy savings. For a good overview of the possibilities of ML in production processes, see Weichert *et al.*"A review of machine learning for the optimization of production processes" [17]. Recent successes in the field of ML can be attributed to the rapid development of computer technology. ML models typically require parallelized high-performance computers and are therefore not suitable for all AM industries. Predictive models based on deep learning methods require the generation of large sensor data streams or synthetic data based on numerical simulations. It quickly becomes apparent that generating such large sensor data streams is as expensive as computationally intensive numerical simulations. In addition, sensor data must be labelled in order to efficiently train classifying prediction models. In order to develop good predictive models with as little sensor data as possible and to represent complex physical processes in an interpretable way, it is possible to combine classical ML models from computer science and-, formulations of physical processes by means of PDEs from mathematics with engineering applications such as AM in an interdisciplinary way. It is achievable to combine the advantages of these three pillars in a data-driven approach. Methods of this type are grouped under the term scientific machine learning (SciML) or PINNs. The basic idea is to incorporate PDEs into DNNs to improve prediction accuracy with limited amounts of available data sets. DNNs are used to find basis functions as solutions of PDEs. Similarly, it is also possible to model terms or variables from PDEs using deep learning. The foundation for these approaches and the pioneering work worth mentioning was laid by Raissi *et al*. with their work showing how a number of well-known benchmark problems of PDEs such as conservation laws, incompressible fluid flow and propagation of nonlinear shallow-water waves can be solved and also modelled in a data-driven way using DNNs [18–20]. Maziarissi also showed how to discover PDEs using deep hidden physics models without needing deep knowledge of mathematical physics. He transformed observed data into mathematical models of the physical world achieving human level [21]. Wessels *et al.* also developed a physics informed neural network in the application of computational fluid dynamics and called that"The neural particle method" [22].

4 Scientific Machine Learning Methodology

4.1 Deep Hidden Heat Model (DHHM)

In this section, we present an initial vision for data-driven modeling of PDE terms using physics-informed deep learning. Generated sensor data and the underlying physical laws are decoded in a way that was previously only possible through individual human expertise and deep knowledge of mathematical physics.

4.1.1 Discovery of Partial Differential Equations

Generally, most of the physical issues of the real world and also the thermal processes in AM are described by partial differential equations in the form:

$$f := \frac{\partial T(x,t)}{\partial t} - \Theta\left(t, x, T(x,t), q(x,t), \frac{\partial T(x,t)}{\partial x}, \frac{\partial^2 T(x,t)}{\partial x^2}\right) \tag{1}$$

where T is the solution of the temperature and Θ is a nonlinear function combining the solution T and its derivatives with space x, time t and the heat source q in a specific way. We present the solution of the PDE as an approximation by a fully connected deep neural network. The universal neural network approximation theorem described that DNNs can represent arbitrarily complex continuous functions with non polynomial activation function if there are no constraints regarding the number of hidden layers and neurons [23]. To model the individual differential equation terms, a second deep neural network is used. In order to efficiently train a deep neural network, an objective function is required. With the minimization of the objective function, an optimization problem is solved. To solve such an optimization problem, various optimization algorithms such as the annealed gradient descent method are used [24]. For this purpose, the gradients of the loss function with respect to the weights and biases are needed. It is possible to determine these derivatives by automatic differentiation with machine precision [25]. The learned functional description of the solution of the PDE can be derived with respect to the input variables t and x using the same procedure. Thus, the training data for the discovery of the PDE is provided by the DNN for the solution and resulting gradients. The goal is to train both deep neural networks such that the approximated solution can estimate the observed temperature sensor data and equally decipher the underlying physical dynamics. For the first and second DNN minimizing of the objective function is formulated in Eq. (2) and Eq. (3), respectively:

$$L_{data}(W_{data}, b_{data}) = \frac{1}{N_T} \sum_{i=0}^{N_T-1} \left(T_{data}\left(x^i, t^i\right) - T_{pred}\left(x^i, t^i, W_{data}, b_{data}\right)\right)^2 \tag{2}$$

$$L_{pde}\left(W_{pde}, b_{pde}\right) = \frac{1}{N_f} \sum_{i=0}^{N_f-1} \left(f_{pred}\left(x^i, t^i, W_{pde}, b_{pde}\right)\right)^2 \tag{3}$$

where N_T and N_f are the number of training data and collocation points, respectively. W and b are the optimized weights and biases of the DNN. For the second DNN, a penalty function is formulated. In favour of the successful minimization of Eq. (2), so-called collocation points are defined. These are taken from known methods for numerical solution of PDE and represent selected candidate points in the solution domain [26]. For learning the PDE, it is ensured that each collocation point at which the PDE is satisfied is assigned the value 0. Thus, learned PDE equation terms that do not satisfy the PDE are penalized. The two DNNs for discovering the PDE are shown in Fig. 1. In the literature, several functions are proposed as so-called activation functions in deep learning algorithms to make the network nonlinear with respect to the input parameters [27]. Popular activation functions are sigmoid, relu, tanh and heaviside functions. Here, tanh is used as the activation function σ:

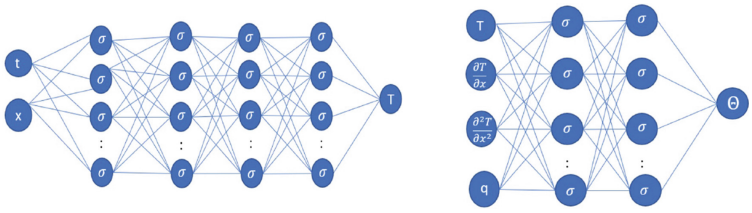

Fig. 1. Potential deep neural network for modeling of PDE in (1)

$$\sigma^l\left(W^l z^{l-1} + b^l\right) = \tanh\left(W^l z^{l-1} + b^l\right) = \frac{e^{\left(W^l z^{l-1}+b^l\right)} - e^{\left(W^l z^{l-1}+b^l\right)}}{e^{\left(W^l z^{l-1}+b^l\right)} + e^{\left(W^l z^{l-1}+b^l\right)}},\ l = 1, \ldots, L \quad (4)$$

where W^l, b^l are weights and biases of each layer l. z describes the input parameters x and t, respectively. We applied the adams method to optimize the searched weights and biases[28]. The discovered PDE is not physical interpretable and represents a kind of empirical black box model. This PDE cannot be solved by classical numerical methods.

4.1.2 Solution of Partial Differential Equation

To solve the discovered PDE and estimate and predict temperature values, a PINN is used as a special kind of black box solver. For the solution of the hidden heat model, a fully connected neural network is used, which consists of two parts. The first part represents an uninformed neural network in which the solution is approximated as closely as possible to the observed data. The second part represents the physically informed part, which ensures that the laws of physics are respected. To obtain a unique solution, an initial boundary value problem must be solved. Thus, the loss function contains only information about the initial condition and Neumann and Dirichlet boundary conditions, as described in Eq. (5):

$$
\begin{aligned}
L_{sol}(W_{sol}, b_{sol}) = \tfrac{1-\lambda_1-\lambda_2}{N_0} \sum_{i=0}^{N_0-1} \left(T_{data_0}\left(x^i, t^i\right) - T_{pred_0}\left(x^i, t^i, W_{sol}, b_{sol}\right)\right)^2 \\
+ \tfrac{\lambda_1}{N_b} \sum_{i=0}^{N_b-1} \Bigg[\left(T_{lb_{data}}\left(x^i, t^i\right) - T_{lb_{pred}}\left(x^i, t^i, W_{sol}, b_{sol}\right)\right)^2 \\
+ \left(T_{ub_{data}}\left(x^i, t^i\right) - T_{ub_{pred}}\left(x^i, t^i, W_{sol}, b_{sol}\right)\right)^2 \Bigg] \\
+ \tfrac{\lambda_2}{N_b} \sum_{i=0}^{N_b-1} \Bigg[\tfrac{\partial}{\partial x}\left(T_{lb_{data}}\left(x^i, t^i\right) - T_{lb_{pred}}\left(x^i, t^i, W_{sol}, b_{sol}\right)\right)^2 \\
+ \tfrac{\partial}{\partial x}\left(T_{ub_{data}}\left(x^i, t^i\right) - T_{ub_{pred}}\left(x^i, t^i, W_{sol}, b_{sol}\right)\right)^2 \Bigg]
\end{aligned} \tag{5}
$$

where N_0 and N_b are the number of initial data and boundary data, respectively. This is a multivariate optimization. The individual terms of the objective function are weighted differently. $\lambda_{1,2}$ usually receive values between 0 and 1, so that the individual terms add up to 1. The obvious disadvantage of the DHHM is the lack of interpretability of the functional description. It is a purely empirical black box model and does not contain any physically interpretable parameters.

4.2 Continuous Heat Model (CHM)

There is another way to model PDEs with DNNs while addressing the interpretability problem. The temperature distribution over a defined time period within a manufacturing process with fixed input parameters can be described with a heat/diffusion equation:

$$
\frac{\partial T(x,t)}{\partial t} = \frac{\lambda}{\rho c} \frac{\partial^2 T(x,t)}{\partial x^2} + \mu q_1(x,t) - q_2(x,t) \; x \in \Omega, t \in [0,T] \tag{6}
$$

where T is the solution of temperature, ρ is the density, λ is the thermal conductivity of the metal components, c is the specific heat capacity, μ is the exposure intensity, $q_1(x,t)$ is the heat source and $q_2(x,t)$ is the heat loss. Most of the energy disappears downward in the z-direction of the component and must necessarily be accounted for in a simplified 2-dimensional model. $q_2(x,t)$ can be given analytically or represented by a DNN,as in our case. The assumption is made that heat transfer in the component is exclusively by conduction. The heat fluid flow in the molten pool is not considered here. The PINNs are used to identify the significant parameters c, λ, and μ that further describe the heating and cooling of the components. In addition, we applied PINNs for the data-driven solution of the discovered PDE to predict the temperatures of the components. The loss function to be optimized to identify the parameters results from the sum of the two residuals (2) and (3) from the DHHM. For the solution of the discovered model, Eq. (5) is minimized.

4.3 Discrete Heat Model (DHM)

In this section a third class of algorithms using physics informed deep learning is presented. The discovery and solution of the physical temperature model here is based only

on two available temporal data snapshots of temperature measurements. First we introduce it by applying Runge-Kutta methods with p stages to the heat Eq. (5) for timestep t_0 and t_1:

$$T_0(x) = T(x) - \Delta t \sum_{j=1}^{p} a_{ij}\left(-\tfrac{\lambda}{\rho c}\tfrac{\partial^2 T(x)}{\partial x^2} - \mu q_1(x) + q_2(x)\right),\ i = 1, \ldots, p \tag{7}$$

$$T_1(x) = T(x) - \Delta t \sum_{j=1}^{p} \left(b_j - a_{ij}\right)\left(-\tfrac{\lambda}{\rho c}\tfrac{\partial^2 T(x)}{\partial x^2} - \mu q_1(x) + q_2(x)\right),\ i = 1, \ldots, p \tag{8}$$

where T_0 and T_1 are the temperature distributions at two different time points, Δt is the timestep and a_{ij} and b_j are the characteristics coefficients known as weights of the explicit time stepping sheme. To decode the PDE, a neural network is applied that provides multiple output temperature values for a time step. This neural network results in two physics-informed neural networks, which are trained based on observed temperature data of two time points of the temperature distribution during manufacturing processes. Accordingly, the loss function is designed as follows:

$$\begin{aligned} L_{pdeD}\left(W_{pde}, b_{pde}\right) = & \frac{1-\lambda}{N_0} \sum_{j=0}^{p-1} \sum_{i=0}^{N_0-1} \left(T_{0_{data}}\left(x^i\right) - T^j_{0_{pred}}\left(x^i, W_{pde}, b_{pde}\right)\right)^2 \\ & + \frac{\lambda}{N_1} \sum_{j=0}^{p-1} \sum_{i=0}^{N_1-1} \left(T_{1_{data}}\left(x^i\right) - T^j_{1_{pred}}\left(x^i, W_{pde}, b_{pde}\right)\right)^2 \end{aligned} \tag{9}$$

where N_0 and N_1 are the number of training data points for both distinct temporal snapshots. For the solution of the discovered heat model and to predict temperature distribution for a future time the Runge-Kutta methods are applied:

$$T_0(x) = T(x) - \Delta t \sum_{j=1}^{p} b_j\left(-\frac{\lambda}{\rho c}\frac{\partial^2 T(x)}{\partial x^2} - q_1(x) + q_2(x)\right) \tag{10}$$

$T(x)$ here is the searched temperature prediction for a future time step. The solution of $T(x)$ is approximated by an uninformed neural network, which results in a physics informed neural network and is trained by minimizing the loss function:

$$\begin{aligned} L_s(W_s, b_s) = & \frac{1-\lambda_1-\lambda_2}{N_0} \sum_{j=0}^{p} \sum_{i=0}^{N_0-1} \left(T_{0_{data}}\left(x^i\right) - T^j_{0_{pred}}\left(x^i, W_s, b_s\right)\right)^2 \\ & + \frac{\lambda_1}{N_b} \sum_{j=0}^{p-1} \left(T^j_{pred}\left(x^{lb}, W_s, b_s\right) - T^j_{pred}\left(x^{ub}, W_s, b_s\right)\right)^2 \\ & + \frac{\lambda_2}{N_b} \sum_{j=0}^{p-1} \left(\frac{\partial}{\partial x} T^j_{pred}\left(x^{lb}, W_s, b_s\right) - \frac{\partial}{\partial x} T_{pred}\left(x^{ub}, W_s, b_s\right)\right)^2 \end{aligned} \tag{11}$$

Here, N_0 are the number of points at previous snapshot T_0 and N_1 are the boundary points of the future snapshot T_1. This is once again a multiple optimization problem with three differently weighted terms. The first term guaranteed the solution of the previous time step and the second and third one restrict the prediction to the boundary conditions of the PDE. With the discrete heat model it is possible to predict the full solution $T(x,t)$ of the temperature distribution when learning is repeated and the predicted temperature distribution is used as the previous step $(T_0,T_1,T_2,T_3...,T_N)$. This gives the solution at discrete time points depending on the time step size. In order to ensure convergence and stability of the solution, the step size in conventional numerical methods is chosen to be very small.

5 Application to Real Datasets from SIEMENS

In this section, the previously described models are presented on the use case AM of metal components on real experiments from SIEMENS.

5.1 Sensor Data Generation

The laser melting machines used have build platform spaces of 250 × 250 × 325 mm^3 and 280 × 280 × 365 mm^3 respectively. Both machines are equipped with thermal imaging cameras. These have a resolution of 382 × 288 pixels and maximum frame rates of 200 Hz. To generate a manageable amount of data, the frame rate was limited to 1 Hz. The thermal imager allows the measurement of thermal emission between 7 μm and 14 μm wavelength. In this wavelength range, the measurements are affected by the emissivity of the object under investigation. The materials used are AISI 316L steel material and IN718 nickelbased alloy with a focus on heat-resistant materials. Both powder materials are frequently used in the laser melting process in industry. The advantages of the material properties, their possible applications and recyclability in the process lead to this widespread industrial use. AISI 316L describes stainless, austenitic chromium-nickel-molybdenum steels that have good resistance in non-oxidizing acids and chlorine-containing media. Due to their chemical composition, 316L materials are inherently corrosion-resistant metal alloys. The"L" stands for"low carbon" and means that it differs from AISI 316 by having a lower carbon content. Due to its good heat-resistant properties (corrosion resistance; high tensile, fatigue, creep and fracture strength up to 700 C), nickel alloy is increasingly used in energy technology (e.g. exhaust gas components in the gas turbine business), the oil and gas industry, aerospace and racing.

5.2 Plancks Law

A thermal imaging camera does not directly measure the temperature of a surface. Instead, it collects the thermal radiation from that surface, which is correlated with its temperature. Any body above zero Kelvin emits radiation.

The radiation per area and wavelength for a black body is given by Plank's law:

$$M_{\lambda,S}(T,\lambda) = \frac{2\pi c^2 h}{\lambda^5\left(e^{\frac{ch}{k\lambda T}}-1\right)} \tag{12}$$

where c is the speed of light, h is the planck constant, λ is the wavelength and k is the boltzmann constant. The emitted amount of radiation is the integral over Planks law. For this camera model, this is between 7 m and 14 m.

5.3 Emissivity

However, this range applies only to perfect black bodies. Metal surfaces are not black and therefore the emissivity of the surfaces must be considered. The emissivity may be close to 1 for a heavily oxidized metal surface. In other cases, a brightly polished surface may have an emissivity close to 0.02. For this reason, we conducted experiments with metal powder and printed surfaces. The experimental setup consisted of a thermocouple as a reference and thermal camera images. It turned out that the emissivity of a good (glossy and not rough) printed surface is 0.2 and that of the powder is 0.6. This can vary depending on the printing parameters chosen and the age of the powder. Another parameter must be taken into account. Depending on the emissivity and the selected temperature range, the camera selects the values of its internal amplifiers and reference voltages. For a measurement with the extreme value metal and powder, there is a realistic chance that the camera is no longer able to measure and an overload of the camera is the result. For this reason, we choose a value of 0.4 for the emissivity. This allows the metal surfaces to be monitored without the risk of overexposing certain pixels. This has the advantage that the entire job can be monitored with one setup and the data remains unchanged even when recalibrated. The calculated temperatures do not correspond exactly to the numerical temperatures of the surfaces or the powder. Due to the unchanged calibration of the camera over the entire print job, a measured temperature increase is a real temperature increase of this component. For training and predictions using the physics-informed neural networks, the exact numerical temperature value of the surfaces is not crucial. What is important is to estimate the change in temperatures and predict which temperature changes of the component are acceptable and which are not.

5.4 Modeling of Heat Source

In order to describe the heating process during the manufacturing steps, it is necessary to mathematically formulate the energy source (laser) and simultaneously define the exposure area for all models (except heat hidden heat model). Here, a method is proposed to approximate the edges of the components and identify the area of the heat source. Assuming that the temperature distribution on a component section is constant at time t_n, the temperature gradient can be used to determine the domain of the heat source. The powder bed area is identified with the help of a threshold. The result is shown in Fig. 2. Due to the hidden exposure times in the data set, the heat source is modulated by a sequence of randomly distributed rectangular laser pulses. After training, the heat source is then adjusted so that the PDE at the collocation points is close to 0, except for the DHHM. There, the sequence of randomly distributed rectangular laser pulses is used. In contrast to the DHHM, a continuous function is required as the heat source in the other models.

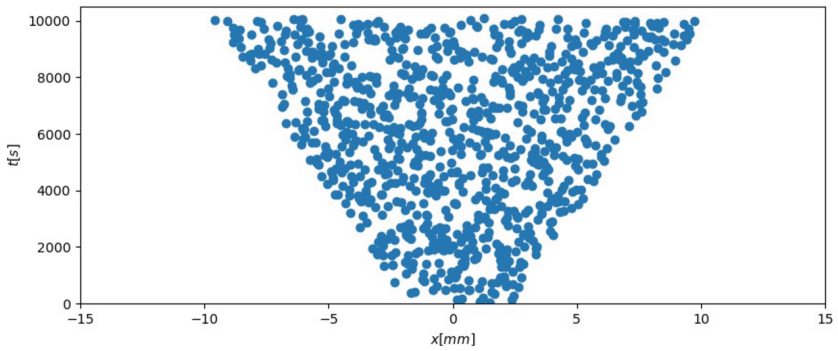

Fig. 2. Domain of heat source - Area of laser exposure, shape of the pyramid component can be seen over the time

5.5 Results

Based on the results, we would like to emphasize how well the modeling and especially the learning of differential equation terms and different parameters works using PINNs. For this, the heating of metallic components, within a manufacturing process, over several layers was modeled using the heat equation and different approaches based on physics informed ML. Later, these discovered models were used to predict the temperature process for higher component layers and later time points.

Table 1 shows the significant parameters for a fixed process configuration. We compare the relative $L^2{}_{exact}$ error of the different models with each other:

$$L^2_{exact} = \frac{1}{N_{nc}} \sum_{i=0}^{N_{nc}-1} \sqrt{(T_{exact}(x^i, t^i) - T_{sol}(x^i, t^i))^2} \tag{13}$$

where N_{nc} is the number of measurement points, T_{exact} are the temperature measurements and T_{sol} are the temperature solution of the different heat models. In the case of the DHM $t = 9602$ s for heating up and $t = 354$ s after exposure time during cooling, respectively. So here we are only dealing with an error between the temperature distribution at fixed points in time, which is why the error is the lowest. For a calculation of the temperature curve over the entire manufacturing time on discrete points similar to the FDM process, the computational effort is very high, so this was deliberately omitted. To obtain the smallest deviation, we refer to the continuous heat model with a relative error $L^2{}_{exact}$ of 30.16×10^{-3}. In addition, all models are able to estimate the temperature distribution over several hours and predict based on the learned models for future layers. In Fig. 3 the heating process based on real measured data from SIEMENS can be seen, as well as learning the different heat models and temperature prediction together with the numerical solution. The numerical solution was determined using the finite difference method, a numerical approximation method for the solution of partial differential equations. In order to compare the results, a parameter optimization of the heat equation was performed using the least square method.

Table 1. Comparison of different computational approaches for the heat up process, showing identified parameters, deviation from experimental data and the results of the converging loss functions for training the deep neural networks.

	DHHM	CHM	DHM	FDM
Material	IN718 $\rho = 8.19$ g/cm^{-3}			
λ (Conductivity)	–	0.3	–	0.29
c (Specific heat)	–	1.89	–	1.67
μ (Laser intensity)	–	294.1	–	265.5
Error L^2_{data}	3.650	–	–	–
Error $L^2_{pde}/$ $L^2_{data+pde}$ $/L^2_{pdeD}$	79.72×10^{-2}	16.22×10^{-5}	–	–
Error $L^2_{sol} \Big/ L^2_s$	2.72×10^{-2}	1.4×10^{-3}	8.9×10^{-3}	–
Error L^2_{exact}	33.07×10^{-3}	30.16×10^{-3}	90.26×10^{-4}	19.09×10^{-2}

Table 2. Comparison of different computational approaches for the cooling down process, showing identified parameters, deviation from experimental data and the results of the converging loss functions for training the deep neural networks.

	DHHM	CHM	DHM	FDM
Material	316L $\rho = 8.0 g/cm^{-3}$			
λ (Conductivity)	–	3.86	–	3.19
c (Specific heat)	–	20.75×10^{-3}		28.63×10^{-3}
Error L^2_{data}	4.561	–	–	–
Error L^2_{pde}	29.65×10^{-2}	19.31×10^{-3}	–	–
Error $L^2_{sol} \Big/ L^2_s$	33.61×10^{-4}	31.5×10^{-5}	94.56×10^{-4}	–
Error L^2_{exact}	20.9×10^{-3}	53.05×10^{-3}	10.68×10^{-3}	34.9×10^{-3}

In this way, parameters could be found which reproduce the measured data. For the discrete heat model, the temperature prediction for time $t = 9602$ s in Fig. 4 was compared with the heating up predictions of the other heat models. Given the thermal training data of 117 min from the manufacturing process, the next 58 min of the heating process on the component can be predicted very well. Another 3D print job with the same parameter configuration can also be simulated from the beginning with variable

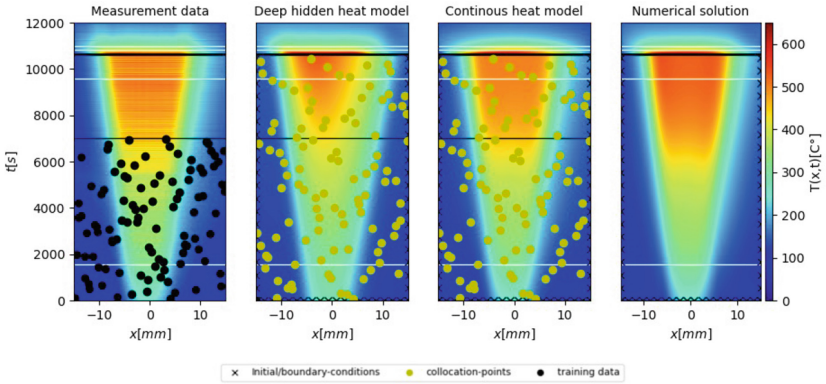

Fig. 3. Comparison of temperature prediciton 1 - heating up layer by layer and cooling down INC718. Real measurement data of experiments from SIEMENS. Temperature prediction based on collocation points and initial and boundary conditions for the different heat models. Discovery of PDE is demonstrated by the training data, which require only 117 min of the measured data to learn the heating up process so that the remaining time steps can be extrapolated. The white lines show which points in time will be compared later in Figs. 4 and 5. The large black line shows the end of heating and the beginning of cooling. The cooling process must be learned in an extra model without heat source in the same way.

laser power using the DHM, CHM and FDM. It can be seen that the optimization of the numerical solution using FDM gives worse results than the physics-informed ML approaches. Furthermore, the FDM has the disadvantage that the solution space has to be discretized in a first step in order to determine the solution on these points iterative. All methods have in common that the unique solution is based on the initial conditions as well as Neumann and Dirichlet boundary conditions. After the components have been exposed, the cooling process begins. Figure 5 shows the results of all thermal models for such a cooling process for a alloy component. In contrast to the previously

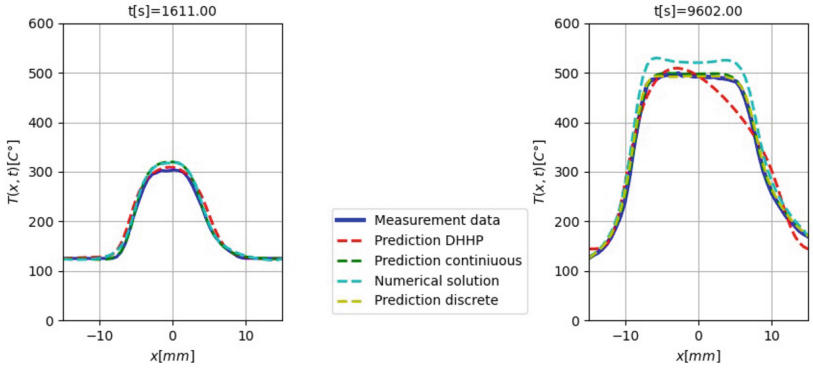

Fig. 4. Comparison of temperature prediction 2 - heating up layer by layer INC718 Left: Prediction of different models at time $t = 1611$ s Right: Prediction of different models at time $t = 9602$ s

presented results, the DHHM has the smallest relative error 20.9×10^{-3} as shown in Table 2. The aim of the modelling is to estimate and predict temperature rise of the components over several layers, that the temporary cooling processes are negligible. If a component becomes too hot, deformations occur and the component quality deteriorates significantly. Therefore, if the maximum temperature of the components can be predicted in the process, it is possible to detect and prevent overheating later in the process.

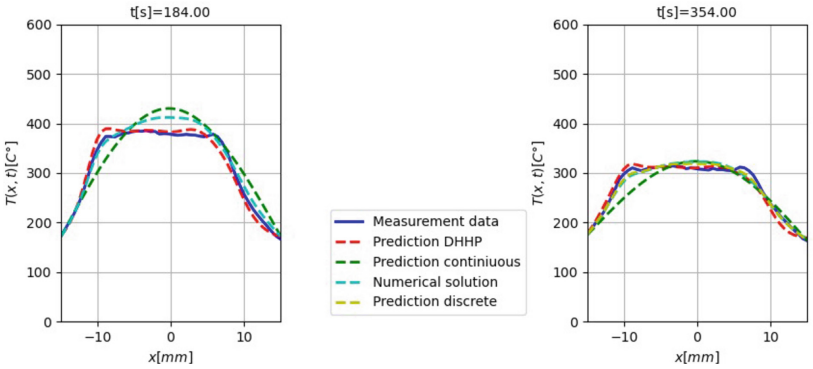

Fig. 5, Comparison of temperature prediction 3 - cooling down after exposure INC718 Left: Prediction of different models at time $t = 184$ s Right: Prediction of different models at time $t = 354$ s

6 Summary and Discussion

In this paper, we present a way to model heating and cooling processes within the AM using physics-informed machine learning. For this purpose, data-driven models are discovered that make it possible to predict temperatures in the manufacturing process. Different SciML approaches were applied to two different print jobs. On the one hand, it was presented how the global heating of the components over several layers can be predicted. Secondly, it was shown how temporary cooling can be modeled using PINNs. Both were demonstrated on real data sets from SIEMENS. The results show that both discrete and continuous PINN models can reproduce and predict temperature values well and significantly reduce the amount of sensor data. We arrived at these findings by comparing the results with the real measured data and the FDM. To achieve the best possible results, the heat source (laser) was modeled by a sequence of rectangular Heaviside functions, but then adjusted to a continuous function later (except DHHM) in the learning process, and the exposure area and thus the edges of the pyramids were determined using the temperature gradient. It should be noted that global heating of the components can be learned, although the components cool temporarily after exposure and coating. In the manufacturing process, the heating process is repeatedly affected by temporary cooling processes, and although it is thus not a classcial continuous heating process, the PINNs perform very well. For this reason, the discovered parameters for thermal conductivity and heat capacity differ from the actual material parameters. In addition, there is noise in the measured data. All approaches presented here have advantages and disadvantages.

With the DHHM, a mathematical description of the temperature distribution can be learned almost blindly, without deeper physical and mathematical understanding. The artificial modeling can keep up with that of a human, as shown. Unfortunately, however, it is not possible to interpret the model and it represents a black box. In the case of the continuous heat model, the interpretability problem has been addressed, but deeper mathematical knowledge is required for modeling. Interpretable parameters, however, can be learned as a byproduct of the optimized weights of the deep neural network. However, no parameters could be found with the DHM to describe the temperature distribution. For the solution at discrete points of the found PDEs of the other models, this approach is nevertheless excellently suitable. However, this model requires more computation time since the solutions are computed at several different future snapshots depending on the time step size. PINNs will probably not replace classical numerical methods in the near future. Especially in low-dimensional spaces, numerical methods require significantly less computation time and are thus faster. The advantages of PINNs only become apparent in higher-dimensional spaces, which can be presented in a future paper. However, as shown earlier in this paper with the 3D printing application, these methods offer a good viable alternative and can prevent components from overheating or cooling too quickly. In addition, the work shown here is a highly simplified thermal model resulting from the low frame rate of 1 frame per second. For future work, this model can be significantly optimized based on more detailed documentation of the manufacturing process using sensor data. With a higher frame rate of the thermal imaging camera, exposure times can be more accurately determined so that the heat source is no longer a black box. In a further work we will try to consider other physical factors like fluid flow in a higher dimensional problem depending on the sensor data.

References

1. Frazier, W.E.: Metal additive manufacturing: a review. J. Mater. Eng. Perform. **23**(6), 1917–1928 (2014)
2. DebRoy, T., et al.: Additive manufacturing of metallic components – process, structure and properties. Prog. Mater Sci. **92**, 112–224 (2018)
3. Knapp, G.L., et al.: Building blocks for a digital twin of additive manufacturing. Acta Mater. **135**, 390–399 (2017)
4. Yap, C.Y., et al.: Review of selective laser melting: materials and applications. Appl. Phys. Rev. **2**(4), 041101 (2015)
5. Mukherjee, T., Wei, H.L., De, A., DebRoy, T.: Heat and fluid flow in additive manufacturing—part i: modeling of powder bed fusion. Comput. Mater. Sci. **150**, 304–313 (2018)
6. Mukherjee, T., Wei, H.L., De, A., DebRoy, T.: Heat and fluid flow in additive manufacturing – part ii: powder bed fusion of stainless steel, and titanium, nickel and aluminum base alloys. Comput. Mater. Sci. **150**, 369–380 (2018)
7. Wessels, H., Weißenfels, C., Wriggers, P.: Metal particle fusion analysis for additive manufacturing using the stabilized optimal transportation meshfree method. Comput. Methods Appl. Mech. Eng. **339**, 91–114 (2018)
8. Zhang, Z., et al.: Numerical methods for microstructural evolutions in laser additive manufacturing. Comput. Math. Appl. **78**(7), 2296–2307 (2019)
9. Cao, L.: Numerical simulation of the impact of laying powder on selective laser melting single-pass formation. Int. J. Heat Mass Transf. **141**, 1036–1048 (2019)

10. Razavykia, A., Brusa, E., Delprete, C., Yavari, R.: An overview of additive manufacturing technologies-a review to technical synthesis in numerical study of selective laser melting. Materials (Basel, Switzerland) **13**(17), 3895 (2020)
11. Crank, J., Nicolson, P.: A practical method for numerical evaluation of solutions of partial differential equations of the heat-conduction type. Adv. Comput. Math. **6**(1), 207–226 (1996)
12. Narasimhan, T.N., Witherspoon, P.A.: An integrated finite difference method for analyzing fluid flow in porous media. Water Resour. Res. **12**(1), 57–64 (1976)
13. Razzak, M.I., Naz, S., Zaib, A.: Deep learning for medical image processing: overview, challenges and the future. In: Dey, N., Ashour, A.S., Borra, S. (eds.) Classification in BioApps. LNCVB, vol. 26, pp. 323–350. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-65981-7_12
14. Jiao, L., Zhao, J.: A survey on the new generation of deep learning in image processing. IEEE Access **7**, 172231–172263 (2019)
15. 2017 IEEE Intelligent Vehicles Symposium (IV). IEEE (2017)
16. Deng, L., Liu, Y. (eds.): Deep Learning in Natural Language Processing. Springer, Singapore (2018)
17. Weichert, Dorina, Link, Patrick, Stoll, Anke, Rüping, Stefan, Ihlenfeldt, Steffen, Wrobel, Stefan: A review of machine learning for the optimization of production processes. The Int. J. Adv. Manuf. Technol. **104**(5–8), 1889–1902 (2019). https://doi.org/10.1007/s00170-019-03988-5
18. Raissi, M., Perdikaris, P., Karniadakis, G.E.: Physics informed deep learning (part i): Data-driven solutions of nonlinear partial differential equations (2017)
19. Raissi, M., Perdikaris, P., Karniadakis, G.E.: Physics informed deep learning (part ii): Data-driven discovery of nonlinear partial differential equations (2018)
20. Raissi, M., Perdikaris, P., Karniadakis, G.E.: Physics-informed neural networks: a deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. J. Comput. Phys. **378**, 686–707 (2019)
21. Raissi, M.: Deep hidden physics models: deep learning of nonlinear partial differential equations. J. Mach. Learn. Res. **19**(1), 932–955 (2018)
22. Wessels, H., Weißenfels, C., Wriggers, P.: The neural particle method – an updated lagrangian physics informed neural network for computational fluid dynamics. Comput. Methods Appl. Mech. Eng. **368**, 113127 (2020)
23. Hornik, K., Stinchcombe, M., White, H.: Multilayer feedforward networks are universal approximators. Neural Netw. **2**(5), 359–366 (1989)
24. Pan, H., Niu, X., Li, R., Dou, Y., Jiang, H.: Annealed gradient descent for deep learning. Neurocomputing **380**, 201–211 (2020)
25. Bartholomew-Biggs, M., Brown, S., Christianson, B., Dixon, L.: Automatic differentiation of algorithms. J. Comput. Appl. Math. **124**(1–2), 171–190 (2000)
26. Katsurada, M., Okamoto, H.: The collocation points of the fundamental solution method for the potential problem. Comput. Math. Appl. **31**(1), 123–137 (1996)
27. Nwankpa, C., Ijomah, W., Gachagan, A., Marshall, S.: Activation functions: Comparison of trends in practice and research for deep learning (2018)
28. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization (2014)