Journal of Intelligent Manufacturing
https://doi.org/10.1007/s10845-022-02038-4

Development of a surrogate model for high-fidelity laser powder-bed fusion using tensor train and gaussian process regression

Umesh Kizhakkinan[1] · Pham Luu Trung Duong[2] · Robert Laskowski[3] · Guglielmo Vastola[3] · David W. Rosen[1,4] · Nagarajan Raghavan[1,2]

Received: 22 March 2022 / Accepted: 29 September 2022

Abstract
A multi-physics high-fidelity computational model is required to study the melting and grain growth phenomena in a laser powder-bed fusion (LPBF) additive manufacturing process. The major challenge with the high-fidelity model is long computational time, which makes it unsuited for any feasible process parameter optimization study in a high dimensional process design space. To address this challenge, surrogate models are a good option to replace the high-fidelity model, resulting in a significant shortening of the computational time at the expense of an acceptable drop in accuracy. In this study, a tensor train (TT) and Gaussian process regression (GPR) based methodology is proposed to develop a surrogate of the high-fidelity powder-scale model. An in-house developed powder-scale model is used to generate the training data by simulating a microscale model of the powder-bed for different values of laser power. The trained TT-GPR model can predict the thermal history of the powder-bed and melt pool geometry for a specified value of laser power, while the computation time required for prediction of any set of process conditions is less than one second. Here we can achieve an approximate computational speedup of 10^4 with the surrogate model. We provide evidence to claim that the proposed surrogate model provides high computational efficiency without compromising accuracy.

Keywords Laser powder-bed fusion · Surrogate model · Data-driven approach · Tensor train · Gaussian process regression

Introduction

Laser powder-bed fusion (LPBF) is an additive manufacturing process (AM) in which a three-dimensional component is manufactured by selectively melting a layer of metal powder using a laser heat source in a layer-by-layer fashion based on the CAD data of the component. The process parameters such as laser power, scan speed, hatch spacing, layer thickness, scan pattern, etc. should be optimized to manufacture a defect-free component with good mechanical properties. Understanding the physical phenomena at the powder-scale is a key requirement to process optimization.

A computational model is necessary to understand the physical phenomena of the powder bed fusion process,

✉ Umesh Kizhakkinan
umesh_kizhakkinan@sutd.edu.sg

Pham Luu Trung Duong
luu_pham@sutd.edu.sg

Robert Laskowski
rolask@ihpc.a-star.edu.sg

Guglielmo Vastola
vastolag@ihpc.a-star.edu.sg

David W. Rosen
david.rosen@me.gatech.edu

Nagarajan Raghavan
nagarajan@sutd.edu.sg

[1] Digital Manufacturing and Design Centre, Singapore University of Technology and Design, Singapore 487372, Singapore

[2] Engineering Product Development (EPD) Pillar, Singapore University of Technology and Design, Singapore 487372, Singapore

[3] Institute of High Performance Computing, Agency for Science, Technology and Research, Singapore 138632, Singapore

[4] G.W.W. School of Mechanical Engineering, Georgia Institute of Technology, Atlanta, GA 30318, USA

Published online: 27 October 2022

such as the melting of powder, the evolution of the melt pool, temperature gradients and cooling rates, grain growth, mechanical properties, etc. The computational model also helps to optimize the process parameters to achieve superior mechanical properties and dimensional accuracy. In literature, various computational models can be found, which were developed based on different governing equations and assumptions. A recent study by Yan et al. (2020) explained different models for powder bed fusion and these models were classified into three groups based on the level of physics involved in the model: (1) high-fidelity computational fluid dynamics (CFD) models (Khairallah et al., 2016; Yan et al., 2018), (2) continuum-based thermal models (Yan et al., 2015; Fu et al., 2018; Shahabad et al., 2020), and (3) semi-analytical thermal-conduction models (Devesse et al., 2014). The high-fidelity CFD model can incorporate the powder bed with individual powder particles and predict the melting of powder particles, thermal-fluid flow in the melt pool, and heat transfer. The accuracy of this multiphysics model is high; however, this results in high computational time. The continuum-based thermal models are simplified models that consider only heat transfer. The finite element method is used for solving the heat transfer governing differential equation by considering the powder-bed as a continuum. The computation cost of the continuum model is less expensive compared to the CFD model and this method can also be used for part-scale simulations. The semi-analytical thermal-conduction model is a highly simplified method that estimates the steady-state temperature field without considering the powder layer, heat source diameter, and heat loss. The authors conducted a single-track experiment to estimate the scan track width and compared it with the different model's predictions. The estimated relative errors are less than 10%, 20%, and 30% for the high-fidelity CFD, continuum-based thermal, and semi-analytical thermal-conduction models, respectively. A high-fidelity CFD model provides a high accuracy result, but the computation is expensive, and each simulation takes hours in a high-performance workstation or computing cluster. The high-fidelity model is not a feasible method to incorporate with a process parameter optimization algorithm. Here, a less computationally expensive approach that provides highly accurate results is needed. Thus, a surrogate of the high-fidelity model needs to be developed for use inside the optimization routine.

A surrogate model is a computationally accurate approximation of a high-fidelity model. In other words, a surrogate model can be trained using a limited set of experiment or simulation data or both, and the surrogate model can be used to predict the output for a given set of input without conducting further experiments or running a high-fidelity model. Moreover, the surrogate model should be computationally efficient. Review papers by Meng et al. (2020), Wang et al. (2020a), and Goh et al. (2020) discuss an extensive suite of different machine learning methods developed for the additive manufacturing process. The objective of this paper is to develop a specific machine learning method for LPBF in a high-dimensional space, and the focus is on predicting the physical phenomena at the powder-scale. Tapia et al. (2018) developed a surrogate model for LPBF based on Gaussian process regression (GPR). The input process parameters are laser power, scan speed, and laser beam size, and the output parameter is melt pool depth. The melt pool depth is an important parameter in process planning which decides the degree of bonding between the current layer and the previous layer. The training data were generated from experiments and high-fidelity simulations. Moges et al. (2021) developed a hybrid model using polynomial regression and Kriging methods for LPBF to estimate the melt pool width for a given input value of laser power and scan speed. The polynomial regression method is used to develop the surrogate model using the simulation data from a CFD model and the Kriging method is used for calibrating the surrogate model from the experimental data to reduce the predictive error. Yang et al. (2020) developed the neighboring effect modeling method to predict the melt pool area for a given value of laser power, scan speed, and scan pattern. The training data is generated from the image taken by the high-speed camera in the printer. The polynomial regression method is then used for training the model. Mondal et al. (2020) used the GPR to develop a surrogate model to predict the melt pool size: depth and width. They used the Eagar-Tsai (Eagar & Tsai, 1983) analytical model to predict the temperature field produced by a moving heat source, and this model considers only the thermal conduction phenomenon. The input parameters are laser power and scan speed, and the input data set is generated using the Latin hypercube sampling (LHS) method. The output, temperature field, is calculated by using the Eagar-Tsai model at different time instances. Since the Eagar-Tsai model considers only thermal conduction, the cross-section of the melt pool is generally semicircular. So, the melt pool is represented by two parameters: width and depth. Multiple independent GPR models were developed to estimate the melt pool width and depth as a function of time.

In the case of a high-fidelity CFD model, the outputs are high-dimensional variables. For example, the temperature ($T(x, y, z, t)$) is a four-dimensional spatiotemporal variable and that is a function of three spatial coordinates and time. Developing multiple surrogate models to predict a high-dimensional variable is not practical. To develop a surrogate model to predict a high-dimensional variable, a *dimension reduction* method needs to be integrated with the machine learning methods. High-dimensional data can be decomposed into low-dimensional approximations and surrogate modeling can be carried for the low-dimensional data. This reduces the training time for the surrogate model. The surrogate model predicts the low-dimensional output data for

a given input data, and the low-dimensional output data can in turn be used to reconstruct the high-dimensional space. Wang et al. (2019) developed a Singular value decomposition (SVD) based Kriging surrogate model to predict the thermal field in the LPBF process. The training data were generated using a finite element heat transfer model with a moving heat source. Vohra et al. (2020) proposed a method combining Principal component analysis (PCA) with active subspaces to build a surrogate model to predict the residual stress developed in the electron beam melting process. The principal component analysis is used for dimension reduction of the output variable. In this case, the output variable is residual stress, which is a two-dimensional variable. Twelve input variables are considered, namely, process parameters, mechanical and thermal properties, and training points are generated using the LHS method. The output, 2-dimensional stress field, is generated using a finite element-based thermo-mechanical model. The standard approach for dimension reduction is to vectorize the multi-dimensional array (tensor) before applying the methods such as SVD and PCA. This vectorization neglects the correlation between the data points in a multi-dimensional space thereby leading to inaccuracy.

There are three popular approaches for space efficient representation of multi-dimensional array/tensor: Canonical Polyadic/Parallel Factor (CP/PARAFAC), High order decomposition (HOSVD), and Tensor Train (TT) approach (Eldén, 2007; Zhang, 2017). Tensor train decomposition (TTD), which is also known as matrix-product states, was first introduced by Fannes et al. (1992) and rediscovered later by Oseledets (2011). The CP/PARAFAC has the storage complexity (the number of degree of freedom) $\Theta(dnr)$, where d is the dimension of problem (for the case study d=5), r=20 is the rank of the decomposition, n=100=max(100,80,80,39) is the number of discretize point per dimension. Unfortunately, the stability of numerical method for CP/PARAFACE expansion is not guaranteed. The HOSVD has the storage complexity of $\Theta(dnr + r^d)$. The storage complexity of TT is $\Theta(dnr^2)$. The TT approach has the advantage of a superior ability to reduce the degrees of freedom in comparison to the other two methods (Zhang, 2017).

Using the TTD method, a high-dimensional tensor can be approximated as a product of lower-order tensors. TTD has become a popular approach to deal with high-dimensional data in machine learning. In literature, several researchers have used TTD to compress the high-dimensional input data. For example, a video is a collection of image frames at different time. TTD can compress the video data without a significant loss of accuracy and the compressed data can be used as the input for Recurrent neural networks (RNN) (Yang et al., 2017; Tjandra et al., 2017, 2018) and three-dimensional convolutional neural networks (3DCNN) (Wang et al., 2020b) for classification. TTD helps in reducing the computational burden for training and prediction of RNN and 3DCNN. In the work of Dolgov et al. (2021), the TT approach was used to solve a high dimensional Hamilton Jacobi Bellman equation appearing in optimal feedback control. In Ref. (Karagoz & Batselier, 2020), TT and B-Splines was proposed for identification of nonlinear dynamic systems.

It will be hard to obtain an analytical solution to the PDE system that describes the melting and grain growth phenomena in an LPBF additive manufacturing process. Thus, multi-physics computational modeling is a standard approach to study the spatiotemporal process dynamics. The large computational burden required by the multi-physics model makes the optimization of a high dimensional process parameter design space a very challenging task. A common approach for reducing the computational time is to utilize different surrogate modeling techniques. Despite many efficient surrogate model techniques being available (Rasmussen et al., 2005; Kocijan, 2016; Luu Trung Duong et al., 2018), only a few of them consider the case with high dimensionality of the output (Duong et al., 2020b; Kalogeris & Papadopoulos, 2020).

In the article by Duong et al. (2020b) from our research group, the Tensor train (TT) in conjunction with Gaussian process regression (GPR) was proposed to develop a surrogate model for dynamic crack propagation (mode I) in a composite plate, which is a spatiotemporal problem in 3-dimension (x, y, t). The proposed TT-GPR framework consists of two key components: TT, which helps reduce the high dimensional spatiotemporal response to reduced dimension outputs, and GPR to learn the relationship between the reduced dimension output and the process parameters. A two-dimensional rectangular plate, modelled using a 41×82 grid, with an initial crack was considered for the study. Dynamic crack propagation was solved using the phase-field model with the finite difference method which is a computationally expensive approach. The phase-field model was solved for the spatial grid points and for 128 points in the time domain, giving a data set size of 430,336 points. After training the TT-GPR model, the L_2 prediction error was less than 5%. The prediction time for the TT-GPR surrogate model was 9 s. However, the phase-field model took 778 s to complete one simulation. This shows the computational efficiency of the TT-GPR approach over a full-order model.

In the LPBF process, a high-fidelity powder-scale model is required to accurately predict the melting of powder particles, generation of the melt pool, solidification, and grain growth during the printing process. The high-fidelity model is very computationally expensive. Temperature is the primary output and is a function of 3D position and time. Specifically the spatiotemporal domain is discretized into $100 \times 80 \times 80 \times 39$ points for a given value of laser power, which yields a data set that is about 50 times larger than the 2D crack propagation model from Duong et al. (2020b). Twenty-one different

laser power values were chosen as inputs to the high fidelity simulation to generate data sets.

The contributions of the paper are twofold. First, we demonstrate that the TT-GPR method is more accurate than other dimension reduction surrogate modeling methods for 4-dimensional problems with very small training data sets. We compared our TT-GPR results with those from a combination of PCA and GPR and showed that our method was significantly more accurate. Second, we demonstrated the application of the TT-GPR method to AM process modeling problems, which is a new application area for TT-GPR methods. The complex interacting physical phenomena in PBF processes leads to highly nonlinear behaviors that are difficult to capture in surrogate models. In this work, we demonstrated the capability of estimating thermal histories and melt pool shapes in laser powder bed fusion processes with acceptable accuracy.

The paper is organized as follows. The formulation of our high-fidelity powder-scale model is discussed in Section "High-fidelity (HF) powder-scale model". Section "Tensor train—Gaussian process regression (TT-GPR) surrogate model" presents the details of the TT-GPR surrogate model development. Training of the TT-GPR is explained in Section "Training the TT-GPR model". Section "Results and discussion" discusses the prediction using the TT-GPR surrogate model, the prediction accuracy, and the computational advantage of the surrogate model. The concluding remarks of the proposed TT-GPR method are given in Section "Conclusions of the study".

High-fidelity (HF) powder-scale model

The powder scale model couples thermal and phase solvers with a fluid dynamics solver. The heat equation (Eq. 1) is solved to provide the temperature field considering the heat source parameters (position and power), boundary conditions and phase information:

$$\rho c_p \left(\frac{\partial T}{\partial t} + \mathbf{u}\nabla T\right) = \nabla k \nabla T - h\,(T - T_0) - \epsilon\sigma\left(T^4 - T_0^4\right) + q''_{ij,laser} - q''_{evap} - \frac{\mathrm{d}\eta_s}{\mathrm{d}t}\Delta H_{ls} \tag{1}$$

Here, ρ is the total density, c_p is the heat capacity, $\boldsymbol{u}$ is flow velocity, k is heat conductivity, h is convective heat transfer coefficient, ϵ is emissivity, σ is Stefan-Boltzmann constant, T_0 is ambient temperature, q''_{laser} is laser energy, q''_{evap} is the total evaporation latent heat, $\mathrm{d}\eta_s/\mathrm{d}t$ represents the change of amount of solid phase and ΔH_{ls} is the latent heat of fusion. Instead of explicitly rendering the laser by the ray tracing method, we use a simple cylindrical heat source model commonly used by the welding community (Vastola et al., 2018).

The phase solver tracks the solid-liquid transition under a given temperature field. It operates in the framework of the phase-field method by minimizing the free energy defined in Eq. 2 following the work of Lu et al. (2018)

$$F = \int_V \rho^2 \left[(\eta^s - 1)^2 \cdot \Phi(\tau) + (\eta^l - 1)^2 \cdot (1 - \Phi(\tau))\right] + (\rho - 1)^2(\eta^v - 1)^2 + w\sum_{i\neq j}(\eta^i\eta^j)^2 - \sum_{i<j} k_{ij}\nabla\eta^i\nabla\eta^j \tag{2}$$

In the present work, we are not interested in resolving the grain microstructure, therefore the model considers only solid (η^s), liquid (η^l) and vapor (η^v) order parameters, while density (ρ) is an auxiliary quantity serving as mass-stabilizing agent. The function, $\Phi(\tau)$, is a switching function driving transition between the solid and liquid state, defined such that $\Phi(\tau) \longrightarrow 1$ when $\tau < 1$ and $\Phi(\tau) \longrightarrow 0$ when $\tau > 1$, where $\tau = T/T_m$, k is gradient energy coefficient and w is the potential coefficient enforcing single-phase solution at a given spatial point. The solid η^s and liquid η^l fields are evolved as non-conservative phase order parameters following Eq. 3, while variation of density, ρ, follows the conservative Eq. 4. Here, M is the temperature-independent mobility.

$$\frac{\partial\eta}{\partial t} = -M\frac{\partial F}{\partial\eta} - \mathbf{u}\nabla\eta \tag{3}$$

$$\frac{\partial\rho}{\partial t} = \nabla M\nabla\frac{\partial F}{\partial\rho} - \mathbf{u}\nabla\rho \tag{4}$$

The fluid dynamics solver computes fluid velocity and supplies advective terms to the thermal and phase solvers. It is implemented using the Lattice Boltzmann method and is called the Lattice Boltzmann Solver in this work. Our implementation of the HF powder-scale model follows the approach proposed by Fakhari et al. (2017). The model was parametrized to IN718 superalloy material (Agazhanov et al., 2019) and a flow chart briefly summarizing the integration of individual solvers is shown in Fig. 1. Compared to similar capabilities demonstrated by other groups, such as Ref. Yang et al. (2021), our model had the advantage of being fully developed in-house and was customized to enable batch data generation and postprocessing, useful for training the surrogate model. Importantly, the model allows us to explore the influence of process parameters (laser power, scan speed, hatch spacing, layer thickness) onto the quality of the build in terms of surface roughness and defect density and was extensively validated in terms of porosity and microstructure predictions in Mikula et al. (2021); Wei et al. (2022). Therefore, it can be used to reduce the cost of process parameter exploration by reducing the need of experiments.

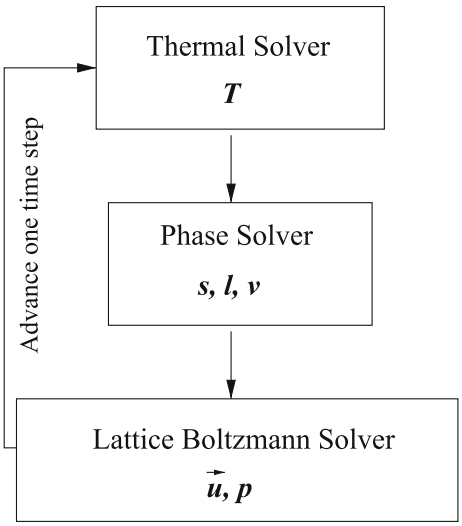


Fig. 1 Schematic workflow of the implementation in the high-fidelity powder-scale model, comprising a thermal solver, a phase solver, and a fluid dynamics solver. The physical fields computed by each solver are listed in italic. T stands for temperature, s, l, v stands for the local amount of solid, liquid and vapor phase, $\boldsymbol{u}$ is the fluid velocity and p is the fluid normalized pressure

Tensor train—Gaussian process regression (TT-GPR) surrogate model

The outputs of a multi-physics computational model normally are of high dimensionality. Hence, they are best represented by a tensor or a multi-dimensional array. However, such an approach neglects the intrinsic correlation between the data points in a multi-dimensional space (Kolda & Bader, 2009; Anandkumar et al., 2014; Lock, 2018). This negligence can make the model less accurate as some strongly correlated components are treated as independent. Moreover, the vectorization leads to the data with large dimension, which can require a very complex model for prediction. Gaussian process regression (GPR) is a popular non-parametric surrogate modeling method. While most GPR works focus on learning scalar output functions, there existing a few multi-output GPR models that are limited to low dimensional output (Rasmussen et al., 2005; Kocijan, 2016; Richardson et al., 2017; Duong et al., 2018; Liu et al., 2018). To address these issues, the Tensor Train based Gaussian Process Regression (TT-GPR) was proposed recently by our group in Ref. (Duong et al., 2020a) for learning spatiotemporal response from data. The TT-GPR method is briefly discussed below.

Tensor Train decomposition (TTD) is a space efficient way for multi-dimension array/tensor representation in which the multi-dimension array is factorized as a product of tensors of lower order (Oseledets, 2011). Thus, TTD can be used to reduce the number of outputs from the spatiotemporal response from high-fidelity simulations and the GPR model can be constructed on the reduced dimensionality output (Duong et al., 2020a).

For the purpose of training data generation, the high-fidelity model will be solved n_1 times with the process parameter $\theta^{(i)}, i = 1, \ldots, n_1$; and the output results are collected into a 5D tensor/array $T(\theta, x, y, z, t) = U(i_1, i_2, i_3, i_4, i_5), i_k = 1, ..., n_k$. The size of the tensor data will be $\prod_{i=1}^{5} n_i$, where n_1 is number of training samples of parameter, n_2, n_3, n_4 are the grid size in spatial dimension x, y, z; and n_5 is the number of time steps in the simulation. One will need to construct $\prod_{i=2}^{5} n_i$ GPR models if the standard GPR approach is used. By approximating the data tensor with the TTD as in Eq. 5, the number of outputs can be significantly reduced:

$$U(i_1, i_2, i_3, i_4, i_5) = \sum_{\boldsymbol{\beta}}^{\mathcal{R}} C_1(i_1, \beta_1) C_2(\beta_1, i_2, \beta_2) C_3(\beta_2, i_3, \beta_3) C_4(\beta_3, i_4, \beta_4) C_5(\beta_4, i_5) \tag{5}$$

where $\boldsymbol{\beta} = [\beta_1, ..., \beta_4]$ is a vector of indices and $\mathcal{R} = [r_1, ..., r_4]$ is a vector of TTD-rank. The terms, C_i, are referred to as the core of the TTD. According to Eq. 5, each element in the collected data tensor is approximated as a multiplication of 5 matrices. The TTD is depicted visually in Fig. 2.

Following the TTD, the data tensor is decoupled, and the first core is corresponding to the parameter space. Thus, the number of outputs is reduced from $\prod_{i=2}^{5} n_i$ to r_1. The procedure for TT-GPR is summarized in Algorithm 1:

Algorithm 1:

- Input: Training parameter set $\left\{\theta^{(i_1)}\right\}, i_1 = 1, ..., n_1$, training data tensor $U(i_1, i_2, i_3, i_4, i_5)$, ε: relative error with respect to data tensor $U(i_1, i_2, i_3, i_4, i_5)$.
- Begin TTD
  - Compute truncation parameter $\delta = \frac{\varepsilon}{2} \|U\|_F$, where $\|.\|_F$ is the Frobenius norm of a matrix.
  - Define a temporary tensor $C = U, r_0 = 1$
  - **For** k=1 to 4 do:

$$W = reshape\left(\mathbb{C}, \left[r_{k-1} n_k, \frac{numel(\mathbb{C})}{r_{k-1} n_k}\right]\right) \tag{6}$$

where *numel()* is the number element of a matrix, *reshape()* is the reshape operator which changes the shape of the array while preserving its column wise order .

- Compute the truncated SVD of W with accuracy δ:

$$U_k S_k V_k + E = W, \|E\|_F \leq \delta, r_k = rank_\delta(W) \tag{7}$$

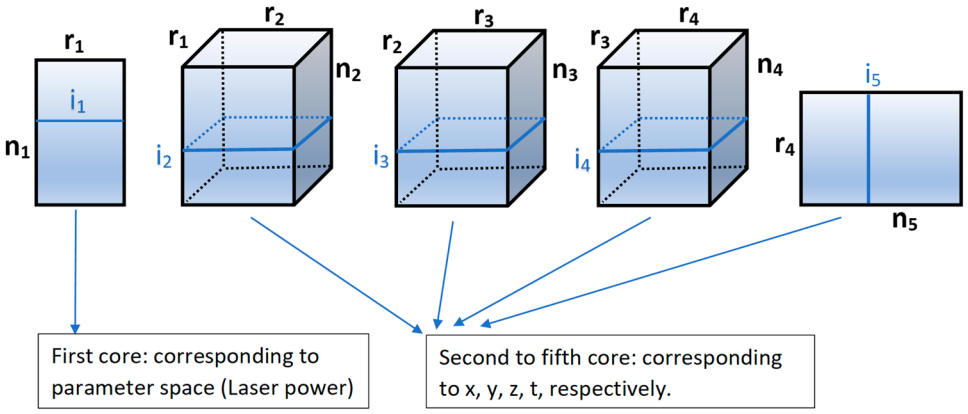


Fig. 2 Graphical visualization of the TTD framework in Eq. 5

- Compute the kth core:

$$C_k = reshape(U_k, [r_{k-1}n_k, r_k]) \tag{8}$$

$$\mathbb{C} = S_k V_k \tag{9}$$

– **end for**

$$C_5 = \mathbb{C} \tag{10}$$

- Train the GPR on the reduced output (columns of the first core): The first core, C_1, corresponds to the parametric space of the solution of the high-fidelity model. It is a matrix of size $n_1 \times r_1$. Fit r_1 GPR surrogate models with r_1 columns of the core C_1.[1]
- To get approximated solutions of the HF at the new set of process parameters represented by a set of $\left\{\theta^{(*j)}\right\}, j = 1, \ldots, n_{test}$:
  - Compute a matrix M, whose columns are the predicted output of r_1 GPR models at the new set of parameter values:

$$[M]_{ji} = \mu_i(\theta^{(*j)}) \tag{11}$$

where $*$ is used to denote the test points or quantities that are computed at the test points, $[\ ]_{ij}$ denote the element at the ith row – jth column of a matrix, $\mu_i(\theta^{(*j)})$ is the predicted output of i-th GPR model at test point $\theta^{(*j)}$.
  - Replace the first core C_1 with the matrix M in Eq. 11.

A flowchart of the proposed methodology is shown in Fig. 3.

[1] Detail about training the GPR model is included in the Appendix. In this work, a scalar process parameter was considered, i.e., the laser power. For vector of process parameters, the algorithm is still the same (Duong et al., 2020a).

Training the TT-GPR model

The training data for TT-GPR are generated using the high-fidelity (HF) powder-scale model. A single-track simulation is performed using the HF powder-scale model for different values of laser power. The domain size of the powder-scale model is 1000 μm × 500 μm × 600 μm with a grid spacing of 5 μm. The simulation is performed at a time increment of 5×10^{-8} s. The base plate thickness is 300 μm. The material considered for the base plate and powder is Inconel 718. The melting temperature of Inconel 718 is 1537.65 K and other thermophysical properties are available in Ref. (Agazhanov et al., 2019). A cylindrical laser heat source with a spot diameter of 200 μm is considered in the powder-scale model, and the ambient temperature is 350 K. The scan speed is kept constant at 1000 mm/s for all the simulations.

Figure 4 shows the single-track simulation result for a laser power of 225 W. To generate the training data, the single-track simulation is performed for different values of laser power ranging from 100 W to 300 W with an increment of 10 W, resulting in a training data of 21 sets. The output variables of the powder-scale model are temperature, solid phase, liquid phase, vapor phase, etc. In this paper, our focus is to develop a surrogate model to predict the temperature distribution and melt pool geometry. Therefore, the spatiotemporal data of the temperature, T, are extracted for a subdomain shown in Fig. 5 for training the TT-GPR model. The size of the subdomain is 500 μm × 400 μm × 400 μm. The subdomain consists of 100 × 80 × 80 grid points. For the time domain data, the simulation results are saved for every 500-time increment. The temperature, $T(P, x, y, z, t)$, is a five-dimensional variable, where P represents the laser power. This five-dimensional variable is first decomposed into multiple low-dimensional components using the Tensor train and the GPR is used to train the component representing the laser power, as discussed earlier in Section "Tensor train—Gaussian process regression (TT-GPR) surrogate model". This resulting size of data tensor for training the surrogate model is $n_1 \times 100 \times 80 \times 80 \times$

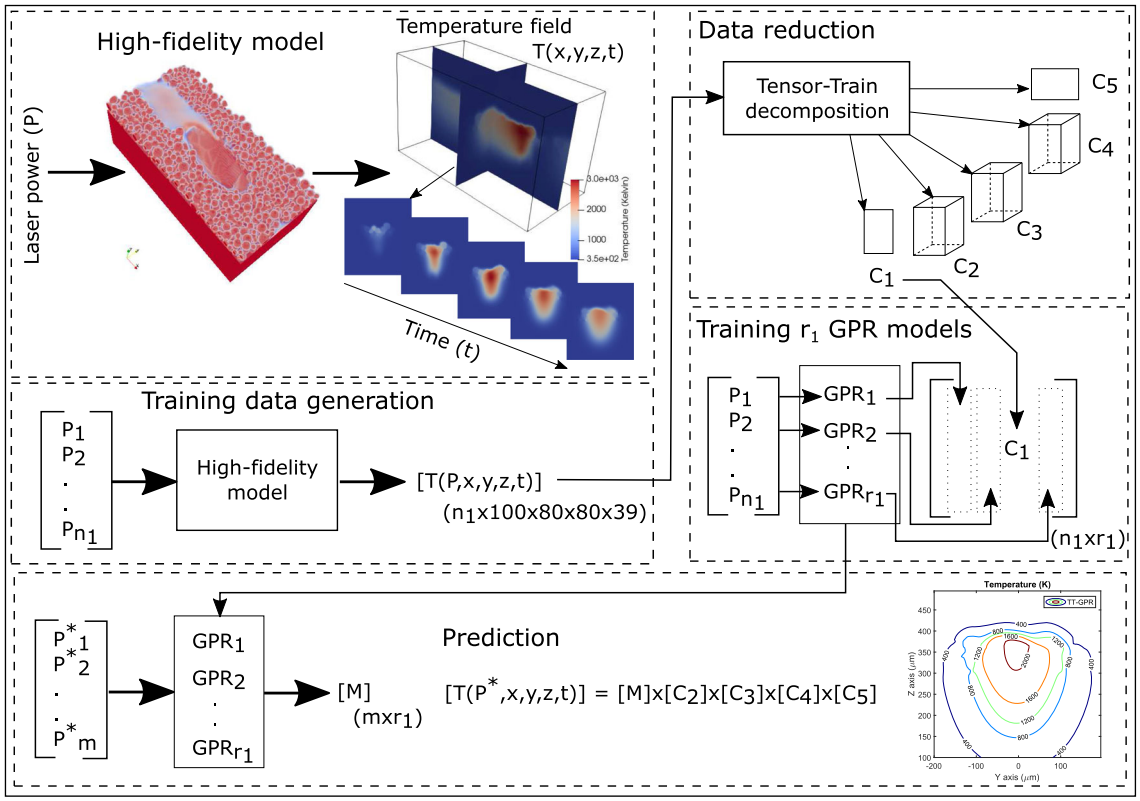


Fig. 3 Flow chart of the TT-GPR surrogate modeling method

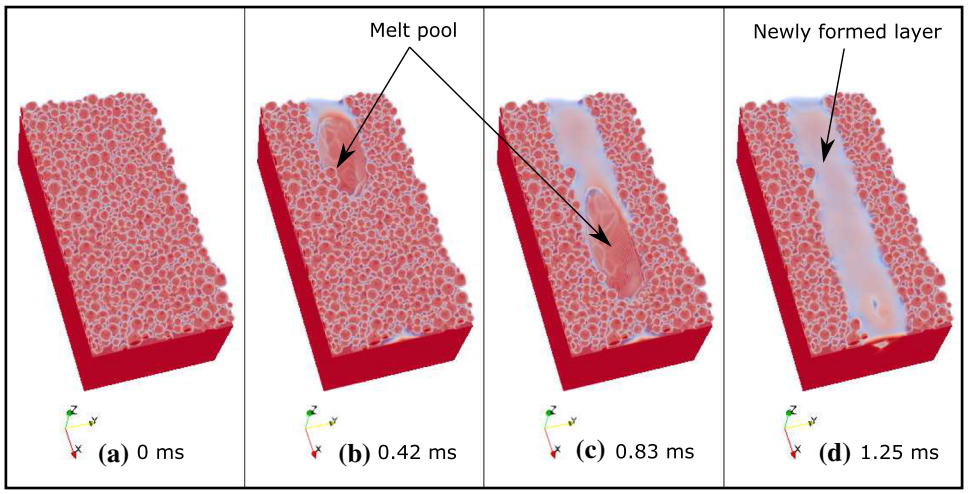


Fig. 4 Different stages of a single-track simulation at different time instances for a laser power of 225 W

39, where n_1 is the number of training samples. Using the TTD algorithm, the number of outputs is reduced from $\prod_{i=2}^{5} n_i = 100 \times 80 \times 80 \times 39$ to 21 (r_1).

Results and discussion

The TT-GPR surrogate model is developed using the training set consisting of 21 values of laser power. Four laser power values viz. 125W, 175W, 225W and 275 W are considered as testing sets. Two independent surrogate models are generated to predict the temperature field and melt pool geometry. The accuracy of the surrogate model is shown in three different scales: (1) comparison through contour plots at a plane in the subdomain for a time instance, (2) cross-validation scatter plots of the subdomain for a time instance, and (3) estimating accuracy metric such as root mean square error and L_2 error over a specific period.

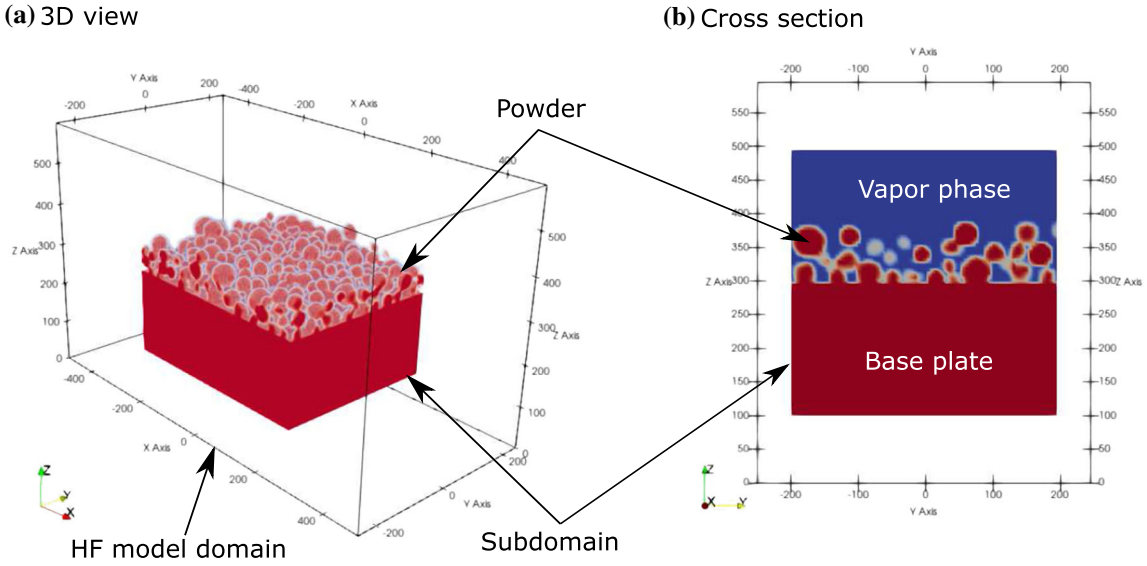


Fig. 5 The subdomain (500 μm × 400 μm × 400 μm) considered for training the TT-GPR model. The size of the HF simulation model is 1000 μm × 500 μm × 600 μm

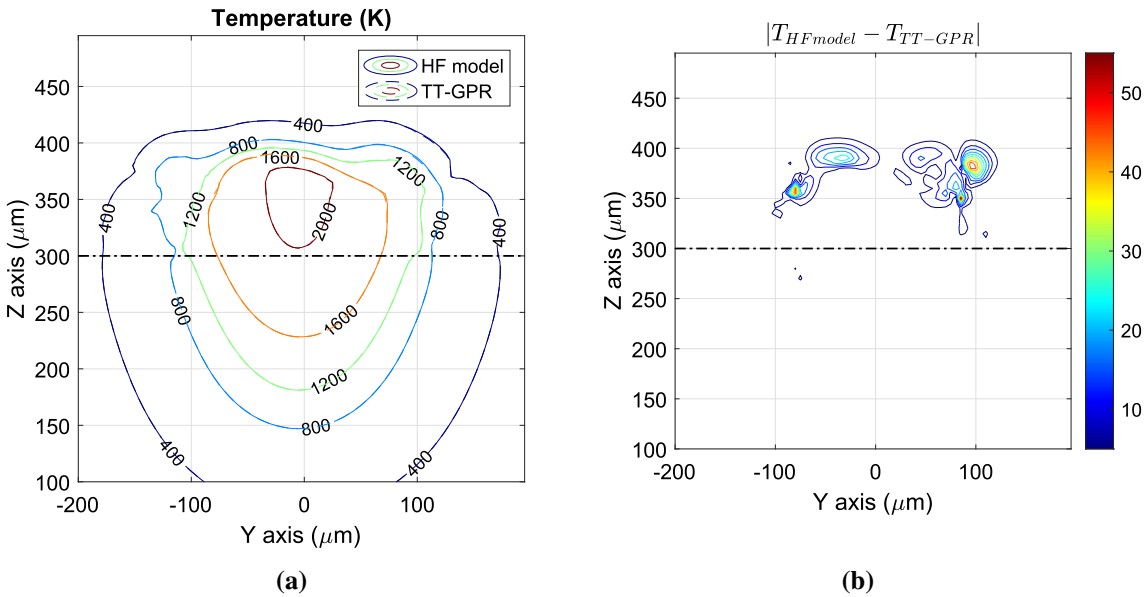


Fig. 6 Comparison of temperature distribution of TT-GPR prediction with the high-fidelity calculation for laser power 225 W at a time of 0.7 ms. **a** Temperature contours on the x-plane at the center of the subdomain. **b** Absolute error between actual and predicted values. The dash-dot line represents the baseplate

Prediction of temperature field

The temperature distribution is estimated using the TT-GPR model at 125, 175, 225, and 275 W laser power and compared with the HF model simulation result. Figure 6 shows the comparison between actual and predicted contour plots of thermal field for the laser power of 225 W. Figure 6a shows the temperature distribution on the x-plane at the center of the subdomain at a time snapshot of 0.7 ms and Fig. 6b represents the absolute error between the actual and predicted values. It can be seen from Fig. 6 that the TT-GPR prediction shows good agreement with the HF model simulation values. The maximum value of the error is around 55 K. The maximum error lies between 350 and 400 μm along the z-direction, where the temperature gradient is high at the powder and vapor interphase. Figure 7 shows the temperature contours and absolute error on the y-plane at the center of the subdomain. In this case, also, the error in the prediction lies above 350 μm in the z-direction. In an LPBF process, the focus is on the temperature distribution in the base plate, which is up

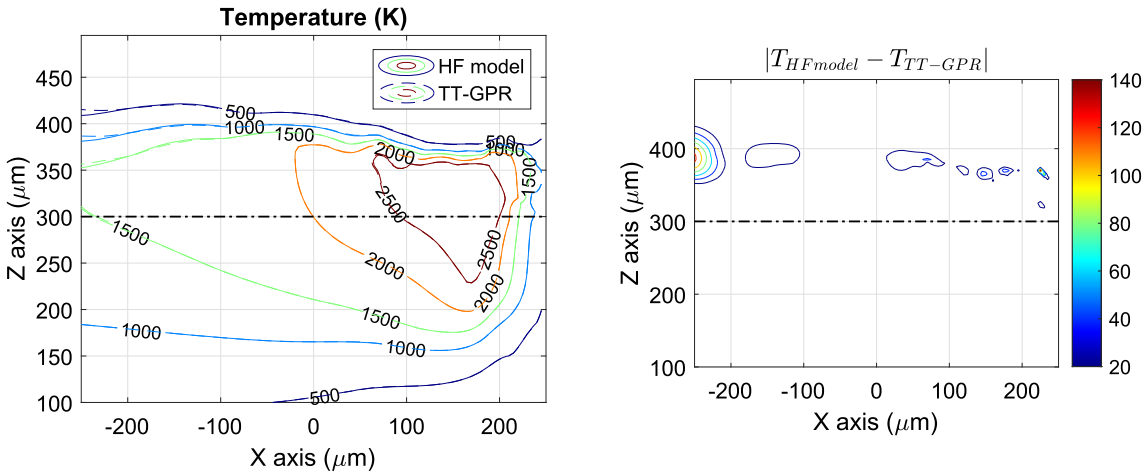


Fig. 7 Comparison of temperature distribution of TT-GPR prediction with the high-fidelity result for a laser power of 225W at a time of 0.7 ms. **a** Temperature contours on the y-plane at the center of the subdomain. **b** Absolute error between actual and predicted values. The dash-dot line represents the baseplate

to 300 μm in the z-direction (represented as dash-dot line), to estimate the grain growth and residual stresses due to rapid cooling.

Figure 8 shows the actual versus predicted scatter plots of the temperature field at a time snapshot of 0.7 ms for the test data sets. The scatter plot represents the deviation in the predicted values from the actual values. For an accurate prediction, the points should be close to the diagonal line. The vertical distance between a point and the diagonal line represents the error in the prediction. It can be seen from Fig. 8 that a large majority of the points are distributed close to the diagonal line which shows that the TT-GPR predictions have good accuracy across the entire spatial domain.

The accuracy of the TT-GPR method is assessed by two metrics: root mean square error (RMSE) and L_2 error. Figure 9a shows the RMSE values at different time instances for the test data set. RMSE provides the standard deviation of the prediction error. The maximum value of RMSE is 10.19 K for the laser power of 225 W. The L_2 error is estimated by $\|Y_{actual} - Y_{predicted}\|_2/\|Y_{actual}\|_2$, where $\|.\|_2$ denotes the standard Euclidean norm and Y represents the output variable. Figure 9b shows the L_2 error plots at different time instances. The maximum value of $L_2(\%)$ error is 1.23 for the laser power of 225W.

Comparison of TT-GPR and PCA-GPR methods

To demonstrate the advantage of the TT decomposition method over other dimension reduction approaches such as SVD and PCA, a surrogate model is developed using the PCA-GPR method. Here PCA is used to reduce the dimensions of the training data, and then the GPR model is trained. The temperature field is predicted for different values of laser powers: 125, 175, 225, and 275 W. The accuracy of this method is estimated and shown in Fig. 10 for the testing data sets. It can be seen from Fig. 10a that the maximum value of RMSE is 19.83 K, which is substantially higher than the maximum RMSE value of TT-GPR (10.19 K) shown in Fig. 9a. Similarly, for the L_2 error, the maximum value of the L_2 (%) error is 2.29 (Fig. 10b), and this value is also substantially higher than the TT-GPR case (1.23) shown in Fig. 9b. From Figs. 9 and 10 , it can be concluded that the TT decomposition approach provides better accuracy than the PCA method.

Effect of number of training data

To study the effect of the number of training data on the prediction accuracy, the TT-GPR model is also trained using 11 and 8 data sets. Figure 11 shows the comparison of the prediction accuracy of three models trained using 8, 11, and 21 data sets for the 225 W case. The maximum values of RMSE are 21.9, 17.74, and 10.19 for the model trained using n_1 = 8, 11, and 21 data sets, respectively. Note that for different numbers of data sets, the size of reduced output r_1 can vary differently and be subjected to $r_1 \leq n_1$. Similarly, the maximum values of the L_2 (%) error are 2.49, 2.11, and 1.23. It shows that the prediction accuracy of the TT-GPR method reduces with a smaller number of training data sets, but the accuracy is in the acceptable range in the case of an interpolation.

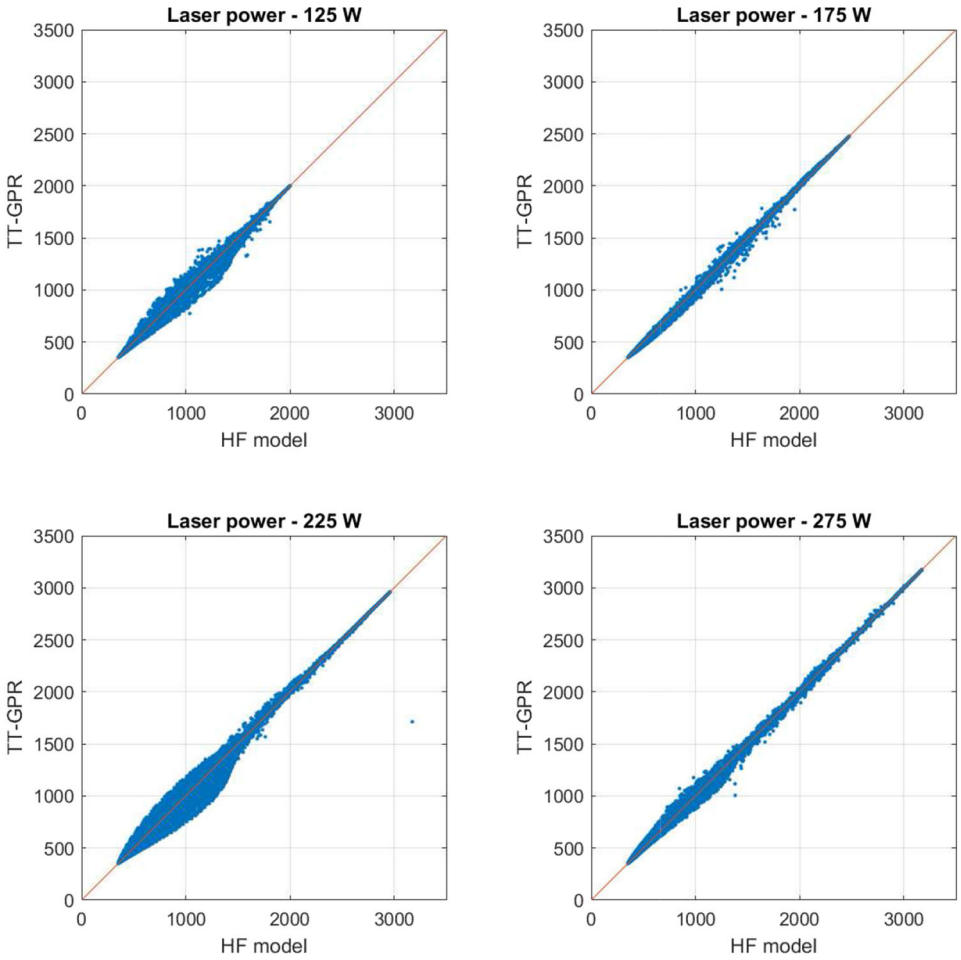
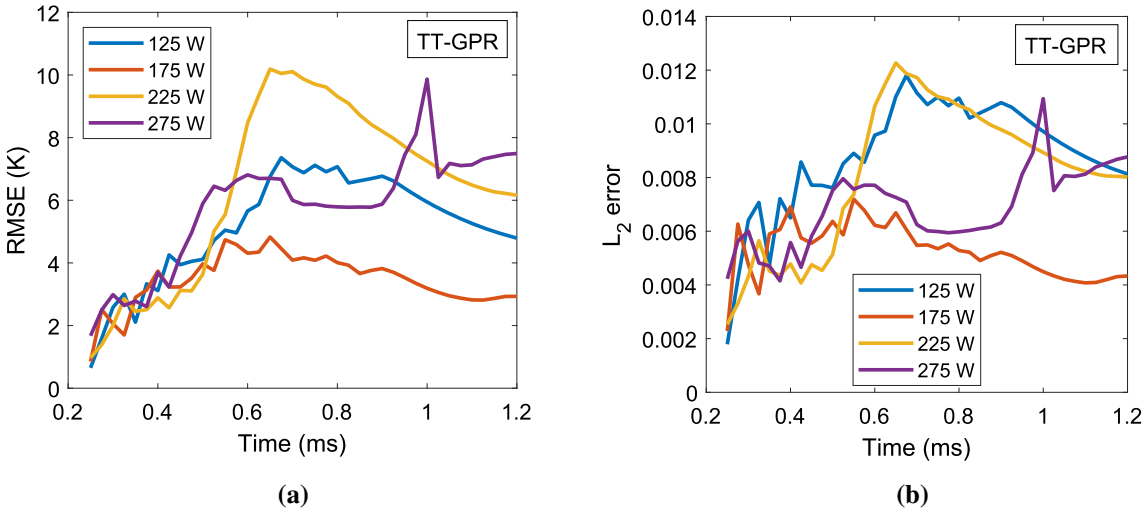


Fig. 8 Actual versus predicted scatter plots of the temperature at a time snapshot of 0.7 ms



(a) **(b)**

Fig. 9 Root mean square error (RMSE) and L_2 error plots of the TT-GPR surrogate model for the testing data sets

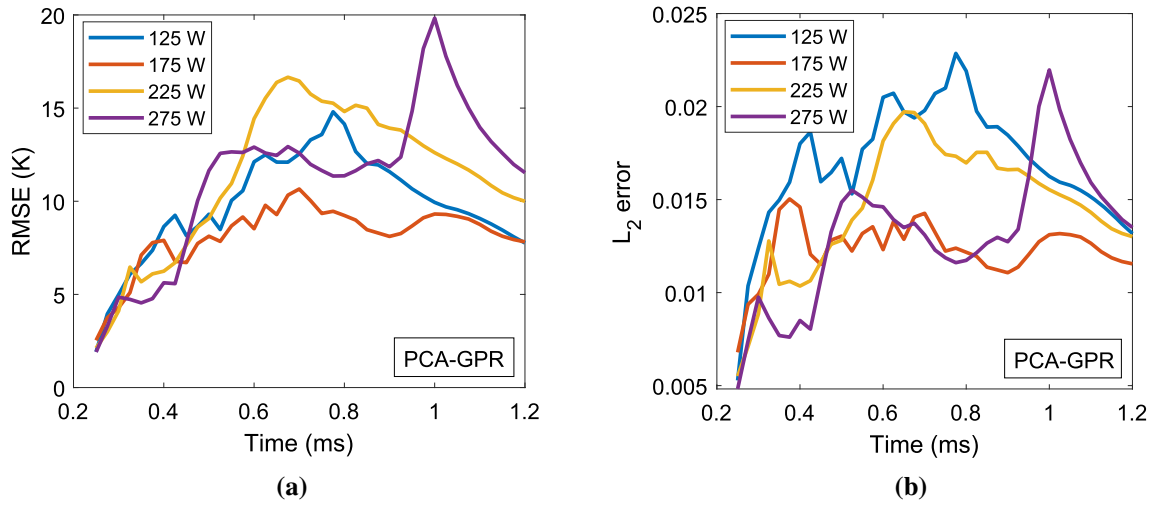


Fig. 10 Root mean square error (RMSE) and L_2 error plots for the testing data sets from the surrogate model developed using the PCA-GPR method

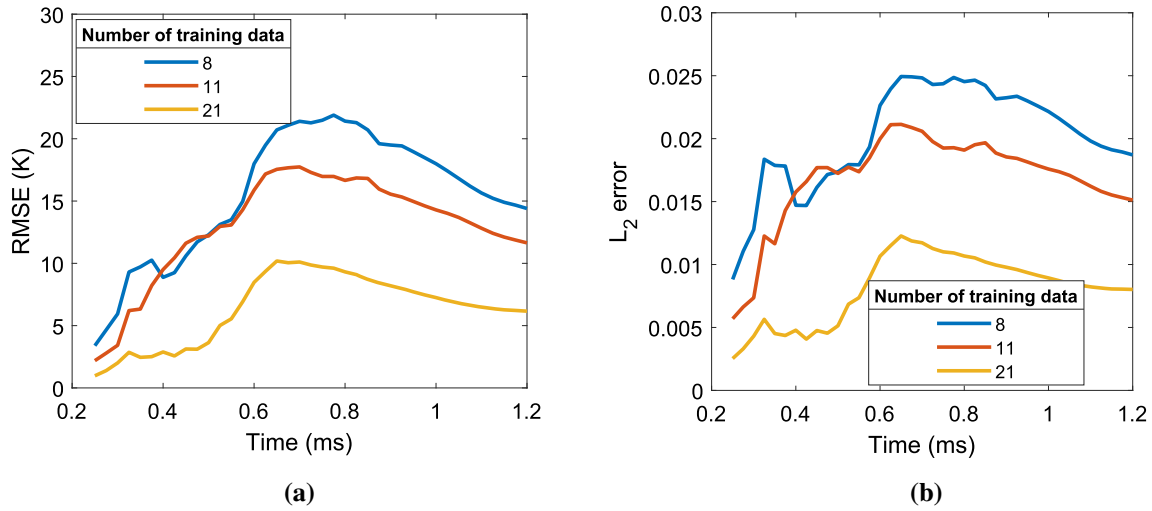


Fig. 11 Comparison of the prediction accuracy of three models trained using different number of training data {8, 11, 21} for the 225 W case

Extrapolation

To study the robustness of the TT-GPR method for an extrapolation scenario, the model is re-trained using the training data sets from 100 W to 260 W with an increment of 10 W. This training data set consists of 17 laser power values. The laser powers of 270, 275, and 280 W are considered as the testing data set. Figure 12 shows the prediction accuracy plots for the different laser powers. The maximum RMSE values for the laser power of 270, 275, and 280 W shown in Fig. 12a are 94.13, 191.06, and 328.81 K, respectively. However, the maximum RMSE value for the interpolation case shown in Fig. 9a is 10.19 K. Similarly the maximum values of L_2 (%) error for the extrapolation case shown in Fig. 12b are 10.51, 21.14, and 36.09 for laser power of 270, 275, and 280 W, respectively. The maximum value of the L_2 (%) error for the interpolation case shown in Fig. 9b is 1.23. It is clear from Figs. 9 and 12 that the RMSE and L_2 error values are substantially high for the extrapolation cases, which indicates that the TT-GPR method is less accurate for extrapolation of temperatures for this problem. The only way possible for improvement is to embed some physics knowledge into the machine learning algorithm.

There are several possible methods such as physics-informed neural networks (PINN) (Raissi et al., 2019), physics-informed Gaussian process (PIGP) (Pang & Karniadakis, 2020) and they consider the full output of the system, which in turn, will require a sufficiently large amount of data. Moreover, it would be very complex to extend and combine this method with model reduction techniques such as tensor train (TT). For the PIGP, one should derive the covariance functions that link the inputs and outputs of the PDE. The

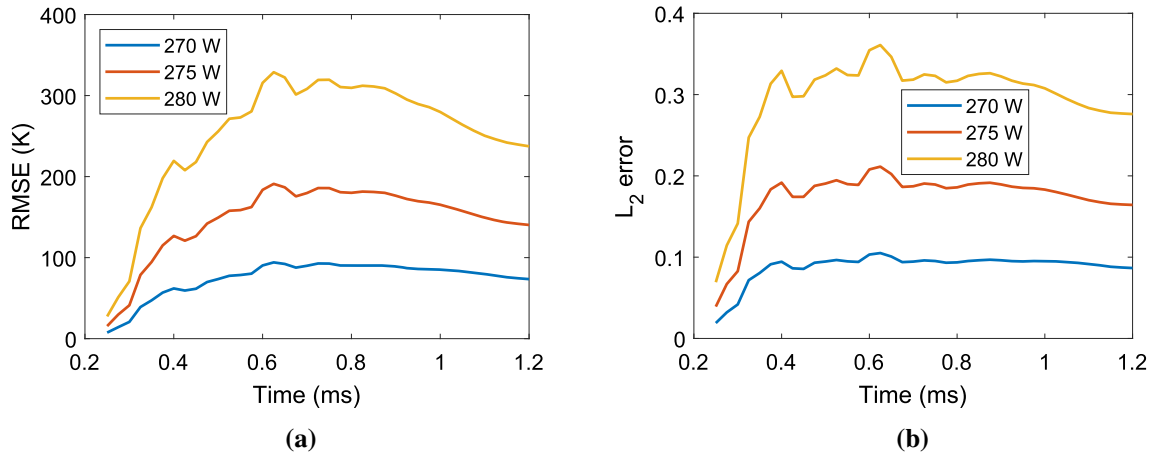


Fig. 12 The prediction accuracy of the model for an extrapolation scenario for laser powers of {270, 275, 280} W

explicit time integration provides the linearization for the nonlinear equation that can help simplify the derivation of the covariance function. By using the TT, however, the linearization cannot be provided by explicit time integration anymore and thus it is quite complex to obtain the covariance function for learning PIGP. Although the results of inverse modeling by PINN and PIGP are encouraging, there are still several issues regarding the accuracy and efficiency of both methods in the forward problem (as discussed in detail in the work of Pang and Karniadakis (2020)).

Prediction of melt pool shape

The shape of the melt pool is another important factor in process planning. The depth of the melt pool decides the bonding between the newly formed layer and the previous layer, and the hatch spacing is determined based on the melt pool width to avoid any nonmelted region between the adjacent scans. The laser power needs to be controlled to achieve a high-density part based on the melt pool shape. Hence, it is important to predict the melt pool shape for process planning, and a surrogate model helps to reduce the computational time for process optimization. One of the output variables of the high-fidelity model is *liquid-phase*, which represents whether a grid point in the model is in the melted state or not. The value of the *liquid-phase* varies from 0 to 1. The value 1 represents that a grid point is in the liquid state and the value 0 represents that a grid point is either in solid state or in vapor state. The value between 0 and 1 represents a transition phase. The melt pool boundary is defined by plotting a contour for the *liquid-phase* value of 0.5. Figure 13 shows the 3D shape of the melt pool at different time instances of the simulation. Figure 14 shows the cross-sectional view of the melt pool for size comparison and it can be seen that a high laser power logically creates a larger melt pool.

The *liquid-phase* data of 21 values of laser power is used for training the TT-GPR surrogate model to predict the melt pool shape. Figure 15 shows the comparison between the high-fidelity model and TT-GPR prediction of the melt pool shape for the laser power of 225 W. The solid line represents the melt pool boundary from the high-fidelity model, and the dashed line is the TT-GPR prediction. It can be seen that the TT-GPR prediction very well matches the high-fidelity model simulation result.

The prediction accuracy of the melt pool surrogate model is expressed in RMSE and L_2 error plots, as shown in Fig. 16. The maximum value of RMSE in Fig. 16a is 0.031 for 225 W laser power. In Fig. 16b, the L_2 error is higher for the 125 W case, and the maximum value is 0.32 in the period between 0.3 to 0.8 ms, represented by a solid line. The value of the liquid-phase varies from 0 to 1, as already mentioned at the beginning of this subsection. Since a subdomain (as shown earlier in Fig. 5) is considered for the development of the surrogate model, at a time of 0.25 ms, only a small volume of the subdomain is in the liquid state. For a laser power of 125 W, the melt pool is relatively small at a time of 0.25 ms. That means that, among the 6.4×10^5 grid points, only a very few grid points have non-zero values. The values of the numerator and denominator of the L_2 error are nearly equal to zero. This results in numerical issues during the evaluation of the L_2 error. The section of the graph in Fig. 16b with numerical issue is represented in dashed line. A similar numerical issue can be observed for the 175 W and 225 W graphs. Such regions have been highlighted with dashed lines in Fig. 16b.

Computational advantage of TT-GPR surrogate model

A high-fidelity single-track powder-scale simulation takes 96 min using 36 CPU threads on a dual-processor (Intel(R)

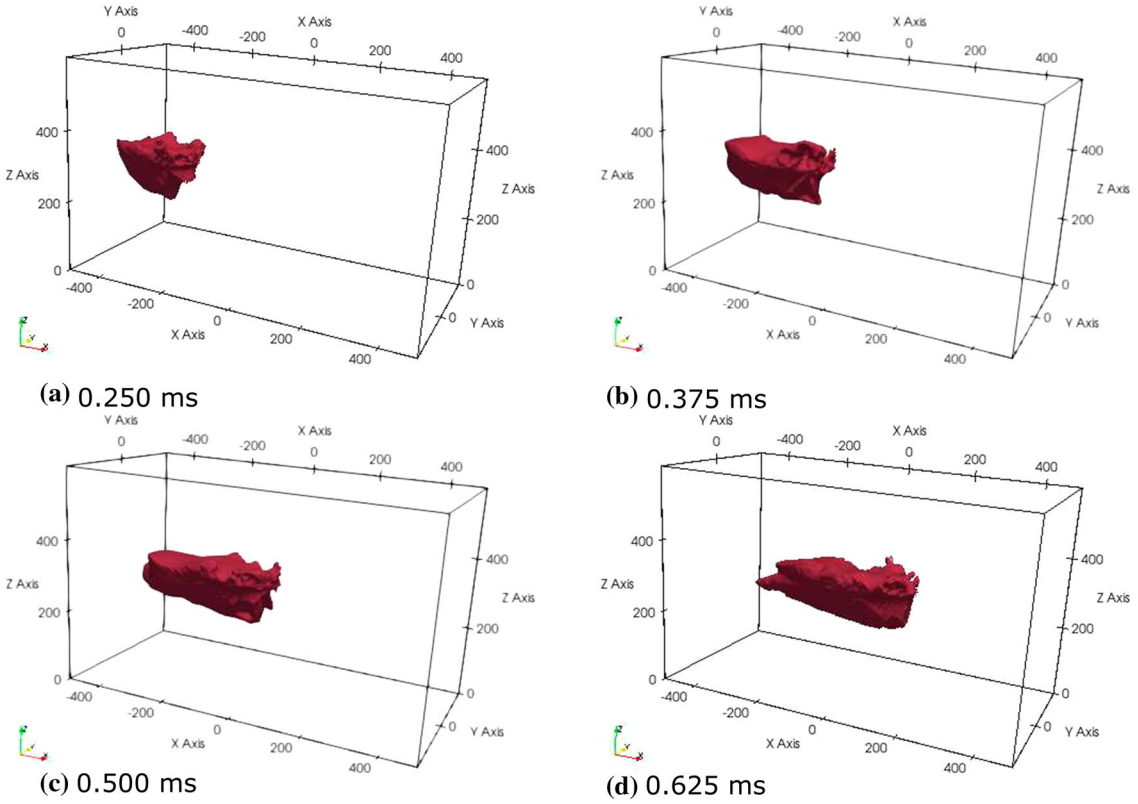


Fig. 13 3D view of the melt pool at different time instances for 225 W laser power

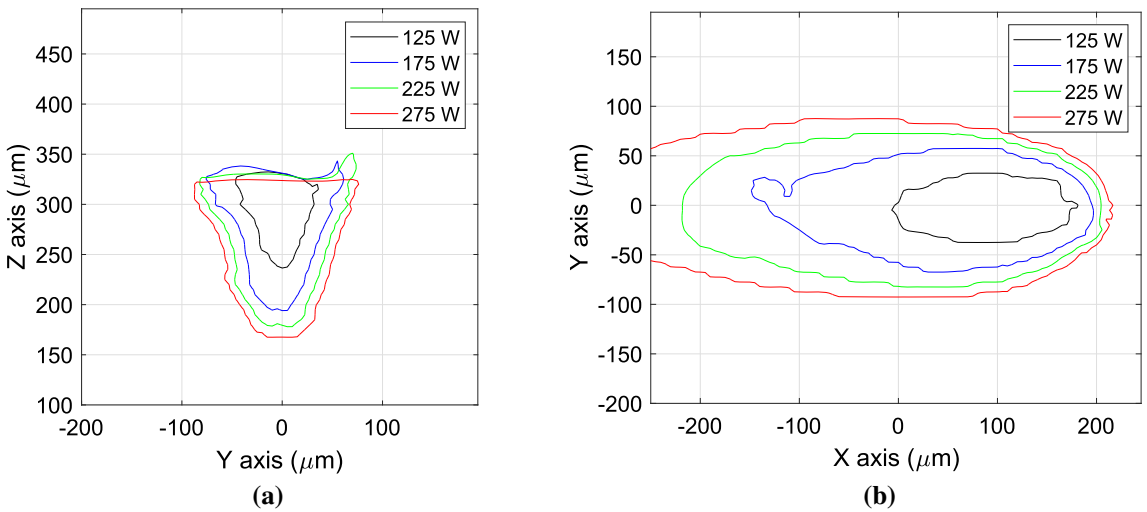


Fig. 14 Comparison of melt pool size at a time of 0.7 ms for different values of laser power. **a** z-plane cross-section at z = 300 μm and **b** x-plane cross-section at x = 100 μm

Xeon(R) E5-2698 v4 @ 2.20GHz) workstation. The TT-GPR training is performed on an Intel(R) Core(TM) i9-10900K @ 3.7 GHz workstation and the training time is a mere 40 s. Then the model is saved for further use. The TT-GPR surrogate model takes 3.1 s to predict the thermal history of the powder-bed for a given value of laser power. This time includes loading of the trained model and prediction of the thermal history. The TT-GPR model takes only 0.2 s to predict the thermal history after loading the model. Here we can achieve an approximate computational speedup of 10^4. This shows the substantial computational advantage of the TT-GPR surrogate model over the high-fidelity powder-scale model with a very negligible compromise in the accuracy. Note that the TT-GPR model is developed for a subdomain,

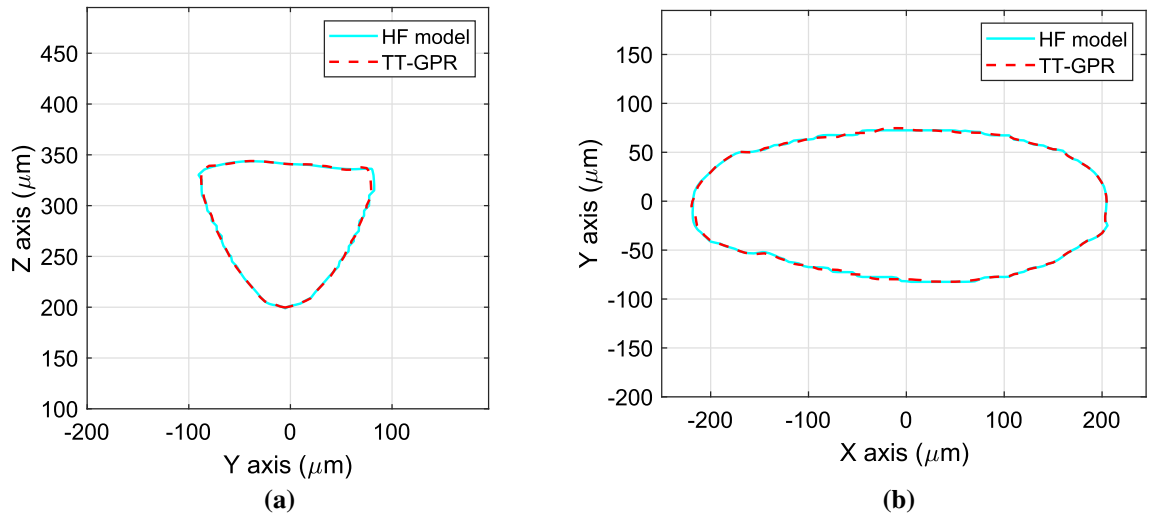


Fig. 15 Comparison of melt pool shape between high-fidelity model simulation and TT-GPR prediction for the laser power of 225 W. **a** z-plane cross-section at z = 300 μm, **b** x-plane cross section at x = 0 μm

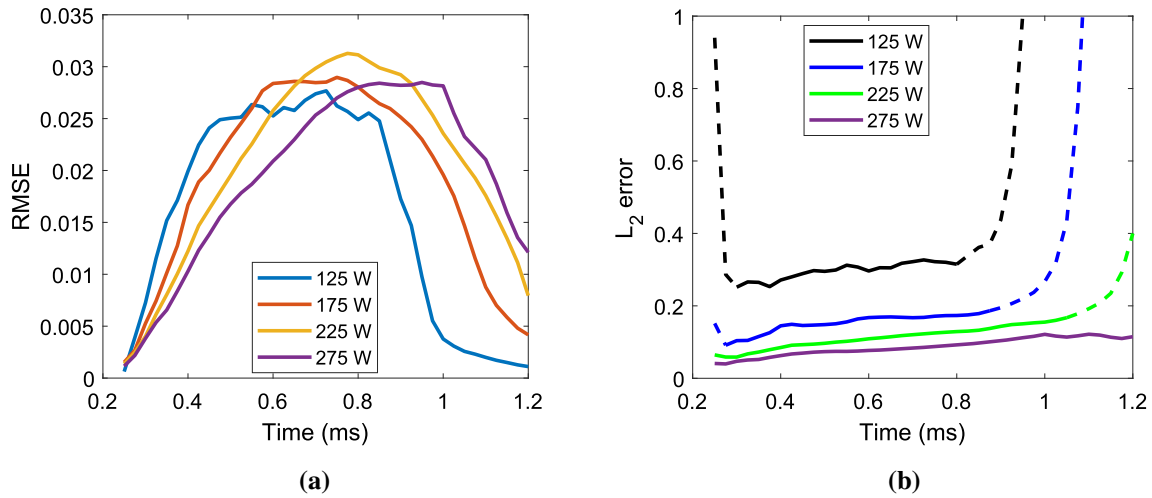


Fig. 16 RMSE and L_2 error plots for the testing data sets. Note that the dashed lines in the L_2 error plots are due to numerical issues

which is smaller than the domain considered in the HF model, and the TT-GPR model predicts the temperature field at every 500-time increment. Also, note that the computational speedup value is a rough estimation because we are running the HF model and TT-GPR on two different workstations with different hardware and software specifications. Therefore, the proposed TT-GPR model is a reliable alternative to be used for inverse design and process parameter optimization tasks where the thermal history can be used as an input to the phase-field and thermal stress solvers.

Conclusions of the study

In this study, a computationally efficient approach is proposed to develop a surrogate model to address a highly-dimensional output variable. The Tensor Train (TT) method is integrated with the Gaussian Process Regression (GPR) machine learning technique to predict the temperature field and melt pool shape in the LPBF process. The laser power is considered as the input variable in the surrogate model. The training data is generated by running the high-fidelity model for different values of the laser power. The surrogate model's prediction for the test data is compared with the simulation results of the high-fidelity model. The prediction accuracy is assessed using multiple error quantification metrics. The surrogate model's predictions show very good agreement with the high-fidelity model simulation results. However, the prediction accuracy of the TT-GPR method is found to be less in an extrapolation scenario. The significant advantage of the TT-GPR surrogate model is its high computation speed (a speed up of 10^4). In fact, a powerful multi-processor worksta-

tion is required to run the high-fidelity powder-scale model, and it takes more than 1.5 hours for one simulation. On the other hand, the TT-GPR surrogate model takes less than a second of time to predict the temperature field for a specified value of laser power in a normal workstation. Hence, the TT-GPR model can be used for inverse design and LPBF process optimization tasks without compromising on the accuracy if our framework is further extended to multiple inputs, not just the laser power as we show here. In this study, only one process parameter was considered as the input to the surrogate model. In future work, different process parameters such as scan speed, hatch spacing, layer thickness, scan pattern, etc., and material properties will be considered as the input parameters for the surrogate model so that a comprehensive process planning and optimization task can be accomplished through heuristic inverse design accounting for multiple objectives and constraints including mechanical property, geometrical non-idealities, print time etc.

Acknowledgements This research is fully supported by A*STAR under its Industry Alignment Fund (IAF) "Programmatic Proposal (PP) project entitled "Industrial Digital Design and Additive Manufacturing Workflows (IDDAMW)" [Award No. A19E1a0097]. Nagarajan Raghavan would also like to acknowledge the A *STAR Explainable Physics based AI Program (ePAI) under Programmatic Proposal Grant No. A20H5b0142 for technical and logistical support to the work.

Appendix A: Gaussian process regression

GPR is a Bayesian surrogate modelling approach in which a Gaussian prior function is placed over the function of interest, i.e., the columns of the first core in Algorithm 1 (Section "Tensor train—Gaussian process regression (TT-GPR) surrogate model"), using a covariance function $\kappa(.,.,\xi_c)$ with vector hyper-parameter, ξ_c. This work utilizes the popular squared exponential function:

$$\kappa\left(\theta,\theta',\xi_c\right) = g_{c_0}\exp\left[-0.5\left(\frac{\theta-\theta'}{g_{c_1}}\right)^2\right] \tag{A1}$$

where $\xi_c = \left[g_{c_0}, g_{c_1}\right]$ is vector of hyper-parameter for the column c. For a training set of parameters $\left\{\theta^{(i_1)}\right\}, i_1 = 1, ..., n_1$ and the column c of the first core $y_c = C_1(:,c)$, the log likelihood can be defined as:

$$\log L\left(y_c|\theta,\xi_c\right) = -0.5 y_c^T\left(K_{yyc}\right)^{-1} y_c - 0.5\log|K_{yyc}| - 0.5 n_1 \log\left(2\pi\right) \tag{A2}$$

$$K_{yyc} = K_{ffc} + \sigma_c^2 I_{n_1\times n_1} \tag{A3}$$

$$\left[K_{ffc}\right]_{ij} = \kappa\left(\theta^{(i)},\theta^{(j)},\xi_c\right),\ i=1,\ldots,n_1,\ j=1,\ldots,n_1 \tag{A4}$$

where K_{yyc} and K_{ffc} are the covariance matrices for column c, $I_{n_1\times n_1}$ is an identity matrix, $[\ ]_{ij}$ denote the element at ith row jth column of a matrix, and $|.|$ denotes the determinant of the matrix. The cth GPR model is trained by minimizing the log likelihood with respect to the vector hyperparameters, $\{\xi_c, \sigma_c\}$. The trained model can then be used to predict the target at any new test process parameter $\theta^{(*j)}$:

$$\mu_c\left(\theta^{(*j)}\right) = k_*\left(K_{yyc}\right)^{-1} y_c \tag{A5}$$

$$k_* = \left[\kappa\left(\theta^{(*j)},\theta^{(1)}\right),\ldots,\kappa\left(\theta^{(*j)},\theta^{(n)}\right)\right] \tag{A6}$$

More details on the standard GPR can be found elsewhere in Refs. (Rasmussen et al., 2005; Kocijan, 2016).

References

Agazhanov, A. S., Samoshkin, D. A., & Kozlovskii, Y. M. (2019). Thermophysical properties of Inconel 718 alloy. *Journal of Physics: Conference Series*. https://doi.org/10.1088/1742-6596/1382/1/012175, Article No. 012175

Anandkumar, A., Ge, R., Hsu, D., et al. (2014). Tensor decompositions for learning latent variable models. *The Journal of Machine Learning Research, 15*(1), 2773–2832. https://doi.org/10.48550/ARXIV.1210.7559

Devesse, W., De Baere, D., & Guillaume, P. (2014). The isotherm migration method in spherical coordinates with a moving heat source. *International Journal of Heat and Mass Transfer, 75*, 726–735. https://doi.org/10.1016/J.IJHEATMASSTRANSFER.2014.04.015

Dolgov, S., Kalise, D., & Kunisch, K. K. (2021). Tensor decomposition methods for high-dimensional Hamilton-Jacobi-Bellman equations. *SIAM Journal on Scientific Computing, 43*(3), A1625–A1650. https://doi.org/10.1137/19M1305136 arXiv:1908.01533.

Duong, P. L. T., Park, H., & Raghavan, N. (2018). Application of multi-output Gaussian process regression for remaining useful life prediction of light emitting diodes. *Microelectronics Reliability, 88–90*, 80–84. https://doi.org/10.1016/J.MICROREL.2018.07.106

Duong, P. L. T., Hussain, S., Jhon, M. H., et al. (2020a). Data driven prognosis of fracture dynamics using tensor train and Gaussian process regression. *IEEE Access, 8*, 222256–222266. https://doi.org/10.1109/ACCESS.2020.3042830

Duong, P. L. T., Raghavan, N., Hussain, S., et al. (2020b) Tensor train decomposition for data-driven prognosis of fracture dynamics in composite materials. In *IEEE Aerospace Conference*

Proceedings. IEEE Computer Society. https://doi.org/10.1109/AERO47225.2020.9172575

Eagar, T. W., & Tsai, N. S. (1983). Temperature fields produced by traveling distributed heat sources. *Welding Research Supplement, 62*, 346–355.

Eldén, L. (2007). Matrix methods in data mining and pattern recognition. *Society for Industrial and Applied Mathematics*. https://doi.org/10.1137/1.9780898718867

Fakhari, A., Mitchell, T., Leonardi, C., et al. (2017). Improved locality of the phase-field lattice-Boltzmann model for immiscible fluids at high density ratios. *Physical Review E*. https://doi.org/10.1103/PhysRevE.96.053301 Article No. 053301.

Fannes, M., Nachtergaele, B., & Werner, R. F. (1992). Finitely correlated states on quantum spin chains. *Communications in Mathematical Physics, 144*(3), 443–490. https://doi.org/10.1007/BF02099178

Fu, G., Zhang, D., He, A., et al. (2018). Finite element analysis of interaction of laser beam with material in laser metal powder bed fusion process. *Materials, 11*(5), 765. https://doi.org/10.3390/ma11050765

Goh, G. D., Sing, S. L., & Yeong, W. Y. (2020). A review on machine learning in 3D printing: Applications, potential, and challenges. *Artificial Intelligence Review, 54*(1), 63–94. https://doi.org/10.1007/S10462-020-09876-9

Kalogeris, I., & Papadopoulos, V. (2020). Diffusion maps-based surrogate modeling: An alternative machine learning approach. *International Journal for Numerical Methods in Engineering, 121*(4), 602–620. https://doi.org/10.1002/NME.6236

Karagoz, R., & Batselier, K. (2020). Nonlinear system identification with regularized Tensor Network B-splines. *Automatica*. https://doi.org/10.1016/J.AUTOMATICA.2020.109300 Article No. 109300.

Khairallah, S. A., Anderson, A. T., Rubenchik, A., et al. (2016). Laser powder-bed fusion additive manufacturing: Physics of complex melt flow and formation mechanisms of pores, spatter, and denudation zones. *Acta Materialia, 108*, 36–45. https://doi.org/10.1016/j.actamat.2016.02.014

Kocijan, J. (2016). Modelling and control of dynamic systems using Gaussian process models. Advances in industrial control. Springer, Cham. https://doi.org/10.1007/978-3-319-21021-6.

Kolda, T. G., & Bader, B. W. (2009). Tensor decompositions and applications. *SIAM Review, 51*(3), 455–500. https://doi.org/10.1137/07070111X

Liu, H., Cai, J., & Ong, Y. S. (2018). Remarks on multi-output Gaussian process regression. *Knowledge-Based Systems, 144*, 102–121. https://doi.org/10.1016/j.knosys.2017.12.034

Lock, E. F. (2018). Tensor-on-tensor regression. *Journal of Computational and Graphical Statistics, 27*(3), 638–647. https://doi.org/10.1080/10618600.2017.1401544

Lu, L. X., Sridhar, N., & Zhang, Y. W. (2018). Phase field simulation of powder bed-based additive manufacturing. *Acta Materialia, 144*, 801–809. https://doi.org/10.1016/J.ACTAMAT.2017.11.033

Luu Trung Duong, P., Quang Minh, L., Abdul Qyyum, M., et al. (2018). Sparse Bayesian learning for data driven polynomial chaos expansion with application to chemical processes. *Chemical Engineering Research and Design, 137*, 553–565. https://doi.org/10.1016/J.CHERD.2018.08.006

Meng, L., McWilliams, B., Jarosinski, W., et al. (2020). Machine learning in additive manufacturing: A review. *JOM, 72*(6), 2363–2377. https://doi.org/10.1007/S11837-020-04155-Y

Mikula, J., Laskowski, R., Dai, L., et al. (2021). Advanced "Digital Twin" platform for powder-bed fusion additive manufacturing. Tech. rep., Hitachi Metals. https://www.hitachi-metals.co.jp/rad/rad02_2021.html

Moges, T., Yang, Z., Jones, K., et al. (2021). Hybrid modeling approach for melt-pool prediction in laser powder bed fusion additive manufacturing. *Journal of Computing and Information Science in Engineering*, *21*(5). https://doi.org/10.1115/1.4050044

Mondal, S., Gwynn, D., Ray, A., et al. (2020). Investigation of melt pool geometry control in additive manufacturing using hybrid modeling. *Metals, 10*(5), 683. https://doi.org/10.3390/met10050683

Oseledets, I. V. (2011). Tensor-train decomposition. *SIAM Journal on Scientific Computing, 33*, 2295–2317. https://doi.org/10.1137/090752286

Pang, G., & Karniadakis, G. E. (2020). *Physics-informed learning machines for partial differential equations: Gaussian processes versus neural networks* (pp. 323–343). Springer. https://doi.org/10.1007/978-3-030-44992-6_14

Raissi, M., Perdikaris, P., & Karniadakis, G. E. (2019). Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics, 378*, 686–707. https://doi.org/10.1016/J.JCP.2018.10.045

Rasmussen, C. E., Williams, & Christopher, K. I. (2005). *Gaussian processes for machine learning (adaptive computation and machine learning)*. The MIT Press.

Richardson, R. R., Osborne, M. A., & Howey, D. A. (2017). Gaussian process regression for forecasting battery state of health. *Journal of Power Sources, 357*, 209–219. https://doi.org/10.1016/j.jpowsour.2017.05.004 arXiv:1703.05687.

Shahabad, S. I., Zhang, Z., Keshavarzkermani, A., et al. (2020). Heat source model calibration for thermal analysis of laser powder-bed fusion. *International Journal of Advanced Manufacturing Technology, 106*(7–8), 3367–3379. https://doi.org/10.1007/s00170-019-04908-3

Tapia, G., Khairallah, S., Matthews, M., et al. (2018). Gaussian process-based surrogate modeling framework for process planning in laser powder-bed fusion additive manufacturing of 316L stainless steel. *International Journal of Advanced Manufacturing Technology, 94*(9–12), 3591–3603. https://doi.org/10.1007/s00170-017-1045-z

Tjandra, A., Sakti, S., & Nakamura, S. (2017) Compressing recurrent neural network with tensor train. In *2017 International Joint Conference on Neural Networks (IJCNN)* (pp. 4451–4458). https://doi.org/10.1109/IJCNN.2017.7966420

Tjandra, A., Sakti, S., Nakamura, S. (2018) Tensor decomposition for compressing recurrent neural network. In *2018 International Joint Conference on Neural Networks (IJCNN)* (pp. 1–8). https://doi.org/10.1109/IJCNN.2018.8489213

Vastola, G., Pei, Q. X., & Zhang, Y. W. (2018). Predictive model for porosity in powder-bed fusion additive manufacturing at high beam energy regime. *Additive Manufacturing, 22*, 817–822. https://doi.org/10.1016/J.ADDMA.2018.05.042

Vohra, M., Nath, P., Mahadevan, S., et al. (2020). Fast surrogate modeling using dimensionality reduction in model inputs and field output: Application to additive manufacturing. *Reliability Engineering & System Safety, 201*, Article No. 106986. https://doi.org/10.1016/j.ress.2020.106986

Wang, C., Tan, X. P., Tor, S. B., et al. (2020). Machine learning in additive manufacturing: State-of-the-art and perspectives. *Additive Manufacturing*. https://doi.org/10.1016/J.ADDMA.2020.101538. Article No. 101538

Wang, D., Zhao, G., Li, G., et al. (2020). Compressing 3DCNNs based on tensor train decomposition. *Neural Networks, 131*, 215–230. https://doi.org/10.1016/J.NEUNET.2020.07.028

Wang, Z., Liu, P., Ji, Y., et al. (2019). Uncertainty quantification in metallic additive manufacturing through physics-informed data-driven modeling. *JOM, 71*(8), 2625–2634. https://doi.org/10.1007/s11837-019-03555-z

Wei, M., Ding, W. J., Vastola, G., et al. (2022). Quantitative study on the dynamics of melt pool and keyhole and their controlling factors

in metal laser melting. *Additive Manufacturing, 54*, Article No. 102779. https://doi.org/10.1016/j.addma.2022.102779.

Yan, W., Smith, J., Ge, W., et al. (2015). Multiscale modeling of electron beam and substrate interaction: A new heat source model. *Computational Mechanics, 56*(2), 265–276. https://doi.org/10.1007/S00466-015-1170-1

Yan, W., Qian, Y., Ge, W., et al. (2018). Meso-scale modeling of multiple-layer fabrication process in Selective Electron Beam Melting: Inter-layer/track voids formation. *Materials & Design, 141*, 210–219. https://doi.org/10.1016/J.MATDES.2017.12.031

Yan, W., Lu, Y., Jones, K., et al. (2020). Data-driven characterization of thermal models for powder-bed-fusion additive manufacturing. *Additive Manufacturing, 36*, Article No. 101503. https://doi.org/10.1016/j.addma.2020.101503.

Yang, M., Wang, L., & Yan, W. (2021). Phase-field modeling of grain evolutions in additive manufacturing from nucleation, growth, to coarsening. *npj Computational Materials, 7*(1), 1–12. https://doi.org/10.1038/s41524-021-00524-6

Yang, Y., Krompass, D., Tresp, V. (2017) Tensor-train recurrent neural networks for video classification. In *34th International Conference on Machine Learning, ICML 2017* (pp. 3891–3900). https://dl.acm.org/doi/10.5555/3305890.3306083

Yang, Z., Lu, Y., Yeung, H., et al. (2020). From scan strategy to melt pool prediction: A neighboring-effect modeling method. *Journal of Computing and Information Science in Engineering*. https://doi.org/10.1115/1.4046335

Zhang, X. D. (2017). *Matrix analysis and applications*. Cambridge University Press. https://doi.org/10.1017/9781108277587