

Microstructure-Sensitive Deformation Modeling and Materials Design with Physics-Informed Neural Networks

Mahmudul Hasan,^{*} Zekeriya Ender Eger,[†] Arulmurugan Senthilnathan,[‡] and Pınar Acar[§]
Virginia Tech, Blacksburg, Virginia 24061

<https://doi.org/10.2514/1.J062708>

Microstructure-sensitive materials design has become popular among materials engineering researchers in the last decade because it allows the control of material performance through the design of microstructures. In this study, the microstructure is defined by an orientation distribution function. A physics-informed machine learning approach is integrated into microstructure design to improve the accuracy, computational efficiency, and explainability of microstructure-sensitive design. When data generation is costly and numerical models need to follow certain physical laws, machine learning models that are domain-aware perform more efficiently than conventional machine learning models. Therefore, a new paradigm called the physics-informed neural network (PINN) is introduced in the literature. This study applies the PINN to microstructure-sensitive modeling and inverse design to explore the material behavior under deformation processing. In particular, we demonstrate the application of PINN to small-data problems driven by a crystal plasticity model that needs to satisfy the physics-based design constraints of the microstructural orientation space. For the first problem, we predict the microstructural texture evolution of copper during a tensile deformation process as a function of initial texturing and strain rate. The second problem aims to calibrate the crystal plasticity parameters of the Ti-7Al alloy by solving an inverse design problem to match the PINN-predicted final texture prediction and the experimental data.

Nomenclature

A	= orientation distribution function
b	= bias of the network
$\tilde{C}_t$	= cell state candidate
f_t, i_t, o_t	= forget, input, and output gate of long short-term memory network, respectively
h^β	= single slip hardening rate
J	= total loss of long short-term memory model
L	= number of time steps during the deformation process
L	= velocity gradient
h_t	= hidden state of long short-term memory network
M	= number of orientation distribution function
P	= property matrix
q	= latent hardening rate
q	= volume normalization vector
R	= fundamental region in the orientation distribution function orientation space
R	= lattice rotation
$RMSE$	= root mean squared error
r	= crystallographic orientation
S	= lattice spin
s_o, h_o, a, s_s	= slip system parameters

U	= hidden weight matrix at long short-term memory network gate
v	= lattice reorientation velocity
W	= weight matrix at long short-term memory network gate
x_t	= input of long short-term memory network
Y	= orientation distribution function output by physics-based simulator
$\hat{Y}$	= predicted orientation distribution function by physics-informed neural network
α_1	= strain rate
δ^{ab}	= Kronecker delta function
$\langle \chi \rangle$	= volume-averaged material property
ω_1, ω_2	= weight parameters in the loss function

I. Introduction

RESEARCH on multiscale materials modeling is focused on examining the connection between thermomechanical processes, structures, and material properties by investigating physical behavior at multiple lengths and time scales. Understanding the multiscale material response by utilizing entirely experimental approaches is infeasible due to the overwhelming amount of parameters/conditions that must be investigated [1]. Therefore, computational approaches are required to assess the multiscale characteristics and performance of materials. The multiscale computational efforts have been increasingly visible since the introduction of the integrated computational materials science (ICME) paradigm [2]. The ICME paradigm, in particular, supports the application of next-generation numerical methodologies to improve the current understanding of materials modeling and design. To achieve this goal, we build novel physics-informed and data-driven crystal plasticity surrogate models of processing-microstructure-property linkages for two metallic materials using a long short-term memory (LSTM) network.

Extracting useful information from existing data through various computing resources has become an important paradigm in different scientific disciplines, including image recognition, cognitive science, and genomics [3]. Like other fields, machine learning (ML) has also become a popular approach for materials design and discovery [4]. In the past, data-driven ML has been applied to different studies in the field, including the polycrystalline materials design [5,6], materials discovery [7,8], and microstructure design for obtaining specific design parameters [9,10]. For instance, an ML-based crystal plasticity model representation for titanium–7wt%aluminum (Ti-7Al) alloy

Presented as Paper 2023-0539 at the AIAA SciTech 2023 Forum, National Harbor, MD, & Online, January 23–27, 2023; received 6 December 2022; revision received 15 December 2023; accepted for publication 1 February 2024; published online 11 March 2024. Copyright © 2024 by the authors. Published by the American Institute of Aeronautics and Astronautics, Inc., with permission. All requests for copying and permission to reprint should be submitted to CCC at www.copyright.com; employ the eISSN 1533-385X to initiate your request. See also AIAA Rights and Permissions www.aiaa.org/randp.

^{*}Graduate Research Assistant, Ph.D. Student, Department of Mechanical Engineering; currently Postdoctoral Research Associate, University of Pittsburgh. Student Member AIAA.

[†]Graduate Research Assistant, Ph.D. Student, Department of Mechanical Engineering.

[‡]Graduate Research Assistant, Ph.D. Student, Department of Mechanical Engineering; currently Postdoctoral Researcher, Vanderbilt University. Student Member AIAA.

[§]Assistant Professor, Department of Mechanical Engineering. Member AIAA.

was developed by Acar [11] using data-driven supervised learning techniques. While these data-driven predictive ML models are widely utilized, their prediction accuracy is immediately related to the quality and size of the training data.

Many engineering models are built upon underlying physics. The solutions of these physics-based models also need to satisfy boundary conditions, initial conditions, and physics-derived design constraints. For traditional data-driven modeling, a large amount of data are required to build a high-fidelity model [12]. However, due to computational time requirements, it is not efficient for some models, like crystal plasticity modeling, to produce sufficient data to train a high-fidelity ML model. Moreover, traditional ML models may not be *explainable* as they may not be aware of the underlying physics-based design constraints [13]. To incorporate the problem physics within the ML model and train the model with small data, a new framework called the physics-informed neural network (PINN) was introduced by Raissi et al. [3]. Before this study, Lee and Kang [14] first applied a neural network algorithm to solve the differential equations. After that, Lagaris et al. [15] solved ordinary and partial differential equations using artificial neural networks. They used a similar methodology to solve boundary value problems with irregular boundaries [16]. Later, Sirignano and Spiliopoulos [17] developed a deep learning algorithm to solve high-dimensional (up to 200 dimensions) partial differential equations (PDEs) with boundary and initial conditions. PINN has become very popular within the scientific community since Raissi et al. published a detailed framework of PINN for solving PDEs [18] and the discovery of PDEs [19]. Later, they published an integrated version of the previous two papers [3]. Subsequently, PINN was adopted in different engineering applications, such as solid mechanics [20–22], crystal plasticity modeling [23,24], fluid mechanics [25–29], heat transfer [30], nano-optics, metamaterials [31,32], medicine [33–35], and power systems [36]. Recently, PINN was also applied to quantify the model uncertainty [37–39].

In order to optimize the processing paths of deformation (up to 100 combinations) for a targeted metallic microstructure, Dornheim et al. [40] recently formulated a model-free deep reinforcement learning algorithm. Instead of relying on prior samples, their algorithm can communicate with processing simulations during optimization. They expanded the technique to address multi-objective optimization problems. In a different study, Honarmandi et al. [41] proposed a novel approach based on batch Bayesian optimization to solve the inverse problem of determining the material processing requirements using microstructure data. They created a surrogate model based on Gaussian process regression to take the place of the computationally expensive process models and integrated it into inverse design optimization using both low-fidelity and high-fidelity phase field models. In this study, inspired by PINN, we develop physics-informed and data-driven surrogate models to replace costly material processing simulations to predict the final deformed textures and their evolution over time. Models are trained with small data and customized to incorporate the underlying physics-derived constraints. Therefore, this paper presents a combined approach of microstructure-sensitive deformation modeling and materials design with physics-informed ML. The example problems will be presented for two different crystal structures, i.e., hexagonal close-packed (hcp) and cubic. Two example materials, Ti-7Al and copper (Cu), are chosen as they have wide engineering applications under high thermomechanical stresses, including aerospace systems. In the first problem, we develop a surrogate model for Cu that predicts the final texture and its evolution at different time steps when a tensile force is applied to the material for a certain time with different strain rates. Next, we define an inverse design problem to solve the optimum slip and twin system parameters of Ti-7Al to achieve a final texture prediction that matches the experimental texture data. In order to do that, another surrogate model is developed to predict the deformed texture as a function of crystal plasticity parameters when the material is under compression. In both cases, an LSTM neural network is implemented to develop a data-driven model. The LSTM network is a good choice because our study involves time-dependent simulation data. Moreover, the physics-based constraints can be implemented in the LSTM network

[42–45]. The organization of this paper is as follows: Sec. II describes the mathematical modeling of the deformation processing and texture evolution. It also discusses the necessary mathematical background of the physics-informed LSTM network. Results for forward modeling and inverse design using PINN are discussed in Sec. III. The summary of the paper, along with potential future works, is included in Sec. IV.

II. Mathematical Modeling

A. Deformation Process and Texture Evolution Modeling

A polycrystalline material is composed of many crystals with different crystallographic orientations, and these orientations define the microstructural texture. The orientation distribution function (ODF) is utilized to represent the microstructural texture using a local finite element discretization scheme. For each independent crystal orientation (considering the crystallographic symmetry) represented by the ODF, there is an associated volume fraction, which indicates the proportion of the total volume of the microstructure oriented in that particular direction. This definition requires the implementation of a normalization constraint that is expressed in Eq. (1):

$$\int_R A(r, t) dv = 1 \quad (1)$$

In Eq. (1), $A(r, t)$ represents the ODF, which is a function of crystal orientation r and time t . This normalization constraint ensures that the volume fractions associated with all independent crystal orientations sum up to 1 (or, in other words, the probability of having all orientations in a given microstructure is equal to 1), and thus, the ODF defines a valid probability density representation of the orientation space.

The ODFs can be updated as a function of time during deformation using a crystal plasticity constitutive model, which makes it a computationally efficient method. The probabilities are developed specifically from time $t = 0$, corresponding to an initial ODF derived from the initial microstructural texture. Rodrigues axis-angle parameterization of the orientation space approach is applied to depict ODFs [46]. The parameterization is obtained from the scaling of the axis of rotation, n , which is represented as $n = r / \tan(\theta/2)$, where r and θ are the orientation and angle of rotation, respectively. To calculate the meso-scale (volume-averaged, homogenized) properties, a local finite element discretization approach is used along with Rodrigues parameterization. The volume-averaged properties ($\langle \chi \rangle$) of polycrystalline materials can be calculated using the single crystal properties, $\chi(r)$, and the orientation information by considering the homogeneity of the orientations in the elementary volume. At any given time, the homogenized property $\langle \chi \rangle$ can be calculated using the following equation:

$$\langle \chi \rangle = \int_R \chi(r) A(r) dv \quad (2)$$

In both Eqs. (1) and (2), R defines the fundamental region in the orientation space. As mentioned earlier, a finite element discretization approach is applied to model the ODFs over the Rodrigues orientation space. Here, R is a reduced space that is derived from the original space as a result of the crystallographic symmetries in the given polycrystal system (e.g., cubic symmetry or hexagonal symmetry). The discretization of the ODF is shown in Fig. 1, which contains N independent nodes with N_{elem} finite elements and N_{int} integration points per element. Figure 1a represents the hexagonal microstructure, and Fig. 1b depicts the cubic microstructure. The number of independent nodal points (the nodes shown with red color in Fig. 1) of an HCP microstructure is 50 with the Rodrigues representation, while the number of independent ODFs is 76 for the cubic microstructure.

Using a local finite element discretization scheme in the Rodrigues fundamental region, Eq. (2) can be written as

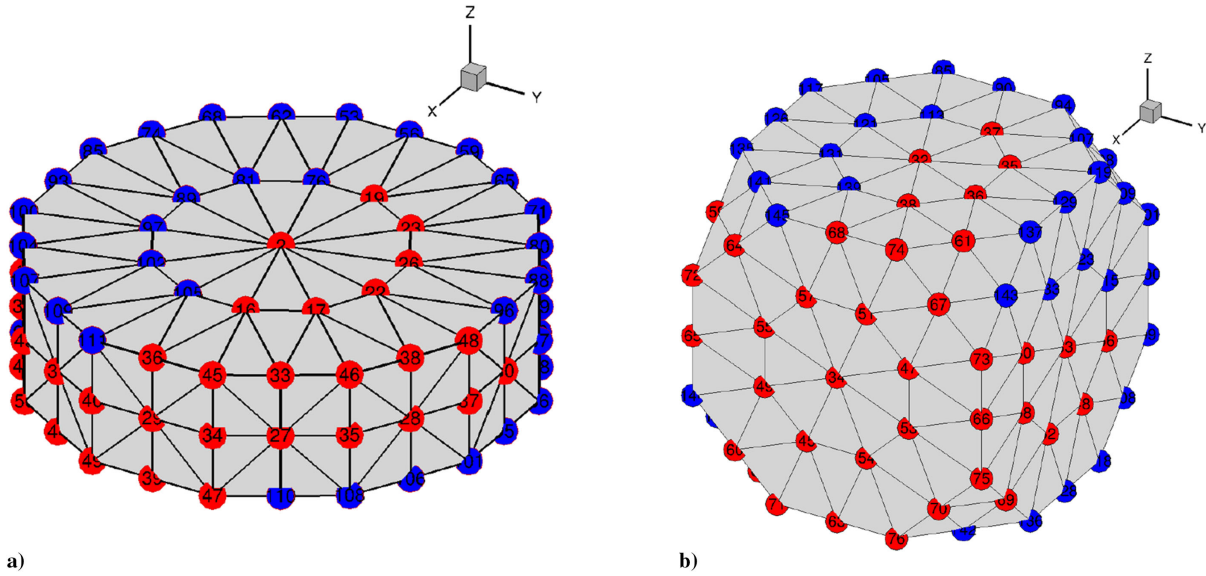


Fig. 1 Finite element discretization of the orientation space for a) hexagonal close-packed (HCP) and b) cubic microstructures. The red-colored nodal points show the independent ODF values, while the blue-colored nodes indicate the dependent ODFs as a result of the crystallographic symmetries.

$$\langle \chi \rangle = \sum_{n=1}^{N_{\text{elem}}} \sum_{m=1}^{N_{\text{int}}} \chi(r_m) A(r_m) \omega_m |J_n| \frac{1}{(1 + r_m \cdot r_m)^2} \quad (3)$$

where $A(r_m)$ is the ODF value at the m th integration point with global coordinate r_m (orientation vector) of the n th element. $|J_n|$ is the Jacobian determinant of the n th element, and ω_m is the integration weight of the m th integration point. Equation (3) can also be written in the linear form as

$$\langle \chi \rangle = \mathbf{p}^T \mathbf{A} \quad (4)$$

where $\mathbf{p}$ is the property matrix that is a product of the single-crystal material properties and finite element discretization of the orientation space, and $\mathbf{A}$ is the column vector of the ODF values for the independent nodes of the finite element mesh (see Fig. 1). Similarly, the linear form of Eq. (1) is given below as shown in Ref. [47]:

$$\mathbf{q}^T \mathbf{A} = 1 \quad (5)$$

where $\mathbf{q}$ is the constant column vector obtained from the finite element discretization.

The texture of a polycrystalline microstructure changes under applied loads during a deformation process. This change is reflected in the ODF values through the ODF conservation equation [48], which is given below:

$$\frac{\partial A(\mathbf{r}, t)}{\partial t} + \nabla A(\mathbf{r}, t) \cdot \mathbf{v}(\mathbf{r}, t) + A(\mathbf{r}, t) \nabla \cdot \mathbf{v}(\mathbf{r}, t) = 0 \quad (6)$$

Equation (6) also satisfies the volume normalization constraint of Eq. (5) at a given time. Here, $\mathbf{v}(\mathbf{r}, t)$ is the reorientation velocity. The microstructure constitutive model can calculate texture evolution in terms of a velocity gradient ($\mathbf{L}$) definition [see Eq. (7)], which is linked to $\mathbf{v}(\mathbf{r}, t)$ by the Taylor macro-micro linking hypothesis. A rate-independent constitutive model is adopted to compute the reorientation velocity [48]. The evolution of current texture $A(\mathbf{r}, t)$ from the initial texture $A(\mathbf{r}, 0)$ is solved by the constitutive model and finite element representation in the Rodrigues orientation space.

Each deformation process, such as tension/compression and shear, generates a particular ODF as output after applying a load for a specific amount of time. The macro velocity gradient $\mathbf{L}$ for a particular process is used by the crystal plasticity solver to explore the ODF evolution during that process. The velocity gradient of a crystal with the orientation, $\mathbf{r}$, can be written as

$$\mathbf{L} = \mathbf{S} + \mathbf{R} \sum_{\alpha} \dot{\gamma}^{\alpha} \bar{\mathbf{T}}^{\alpha} \mathbf{R}^T \quad (7)$$

where $\mathbf{S}$ represents the lattice spin, $\mathbf{R}$ indicates the lattice rotation, and $\dot{\gamma}^{\alpha}$ and $\bar{\mathbf{T}}^{\alpha}$ indicate the shearing rate and Schmid tensor for the slip system α , respectively. The macro velocity gradient expression of Eq. (7) can be written in the following matrix form for the tension/compression process [Eq. (8)], and its derivation is skipped here for brevity, which can be found in Ref. [48].

$$\mathbf{L} = \alpha_1 \begin{bmatrix} 1 & 0 & 0 \\ 0 & -0.5 & 0 \\ 0 & 0 & -0.5 \end{bmatrix} \quad (8)$$

where α_1 relates to the strain rate of the tension/compression process. Additionally, the slip hardening model is integrated into the crystal plasticity simulations as explained next:

$$h^{\alpha\beta} = [q + (1 - q)\delta^{\alpha\beta}]h^{\beta} \quad (\text{no sum on } \beta) \quad (9)$$

where h^{β} is a single slip hardening rate, q is the latent-hardening ratio (which is equal to 1.4 for non-coplanar slip systems), and $\delta^{\alpha\beta}$ is the Kronecker delta function. For the single-slip hardening rate, the following specific form is used:

$$h^{\beta} = h_o \left(1 - \frac{s^{\beta}}{s_s} \right)^a \quad (10)$$

where h_o , a , and s_s are slip hardening parameters. While the cubic microstructures of Cu only involve 12 slip planes, the HCP Ti-7Al demonstrates a more complex deformation behavior. Accordingly, the basal $\langle a \rangle$, prismatic $\langle a \rangle$, pyramidal $\langle a \rangle$, and pyramidal $\langle c + a \rangle$ slip systems, in addition to the $\{10\bar{1}2\} \langle 1011 \rangle$ twinning mechanism, are modeled for Ti-7Al alloy.

B. Physics-Informed Neural Networks

Neural networks are used to estimate the outcomes of a function; therefore, they can be embedded into the physical systems to approximate the solution while satisfying any initial or boundary conditions and physics-derived constraints. In the case of PINNs, the loss function is modified to accommodate the physical laws, initial/boundary conditions, or any design constraints present in the system.

The feed forward neural networks (FNNs) are the most basic network structures composed of different layers with many nodes, which map inputs to the outputs by adding weighted inputs with bias. As FNN relays information in only one direction, problems containing parallel time series data related to each other cannot be predicted accurately with this type of network. Therefore, we are utilizing a special type of recurrent neural network (RNN) called LSTM network to correlate the processing parameters and texture evolution with time during the deformation process.

1. Long Short-Term Memory Network

Unlike FNN, RNN has nodes in its layers that communicate with the nodes of the previous layer, which creates a cycle. Therefore, it can act as a memory to fit the sequence of input variables. This is achieved by backpropagation, ensuring that the weights in the previous layers are updated based on the derivative of the estimated error at the output layer with respect to the weights. The adjustment is performed during each training epoch to facilitate the gradual improvement of the model's predictive capabilities. As the inputs are not independent of each other, relations between the features could be captured directly. It is also able to handle inputs and outputs of different sizes. However, there are still major issues when implementing the network standalone, such as gradients exploding or vanishing and processing large sequences.

In order to tackle the disadvantages of the RNN structure, the LSTM network is proposed. The main additions are the gates in the form of activation functions that provide the ability to select which information to discard or to keep in the memory. As summarized in Fig. 2, the cell is composed of a group of neural networks that operate with three gates. The input gate decides whether new information should enter the cell, the forget gate releases the information that is considered not important, and the output gate decides if the whole process starting from the input gate should affect the output. These gates also reduce the training time, thus helping the whole neural

network system to handle a long sequence effectively. Equations (11–14) give the expressions at the forget (f_t), input (i_t), and output (o_t) gates and cell state candidate ($\tilde{C}_t$), respectively.

$$f_t = \sigma(\mathbf{W}^{(f)}x_t + \mathbf{U}^{(f)}h_{t-1} + \mathbf{b}^{(f)}) \quad (11)$$

$$i_t = \sigma(\mathbf{W}^{(i)}x_t + \mathbf{U}^{(i)}h_{t-1} + \mathbf{b}^{(i)}) \quad (12)$$

$$o_t = \sigma(\mathbf{W}^{(o)}x_t + \mathbf{U}^{(o)}h_{t-1} + \mathbf{b}^{(o)}) \quad (13)$$

$$\tilde{C}_t = \tanh(\mathbf{W}^{(g)}x_t + \mathbf{U}^{(g)}h_{t-1} + \mathbf{b}^{(g)}) \quad (14)$$

In the equations, $\mathbf{W}$ is the gate weight matrix, $\mathbf{U}$ corresponds to the hidden unit weight matrix, and $\mathbf{b}$ stands for the bias. After the multiplication of the input x_t and the hidden state h_t with the weights, they become inputs for their respective activation functions that are sigmoid and tanh denoted as σ and $\tanh$, respectively. Next, these gates are employed to compute the new memory and the cell output by Eqs. (15) and (16), respectively [49]. This involves elementwise multiplication ($\odot$) to ensure that gate values either nullify corresponding values when close to zero or allow them to pass when they are close to one.

$$C_t = \tilde{C}_t \odot i_t + f_t \odot C_{t-1} \quad (15)$$

$$h_t = o_t \odot \tanh(C_t) \quad (16)$$

For modeling the texture evolution of the Cu microstructure during a tensile process, the first LSTM layer takes the initial ODFs and strain rate as the input. Thus, the input vector length N is equal to $M + 1$, where M stands for the length of the ODF data. The output of the first layer is then converted into a matrix so that it has the same shape as the output that has multiple time steps, each of which contains

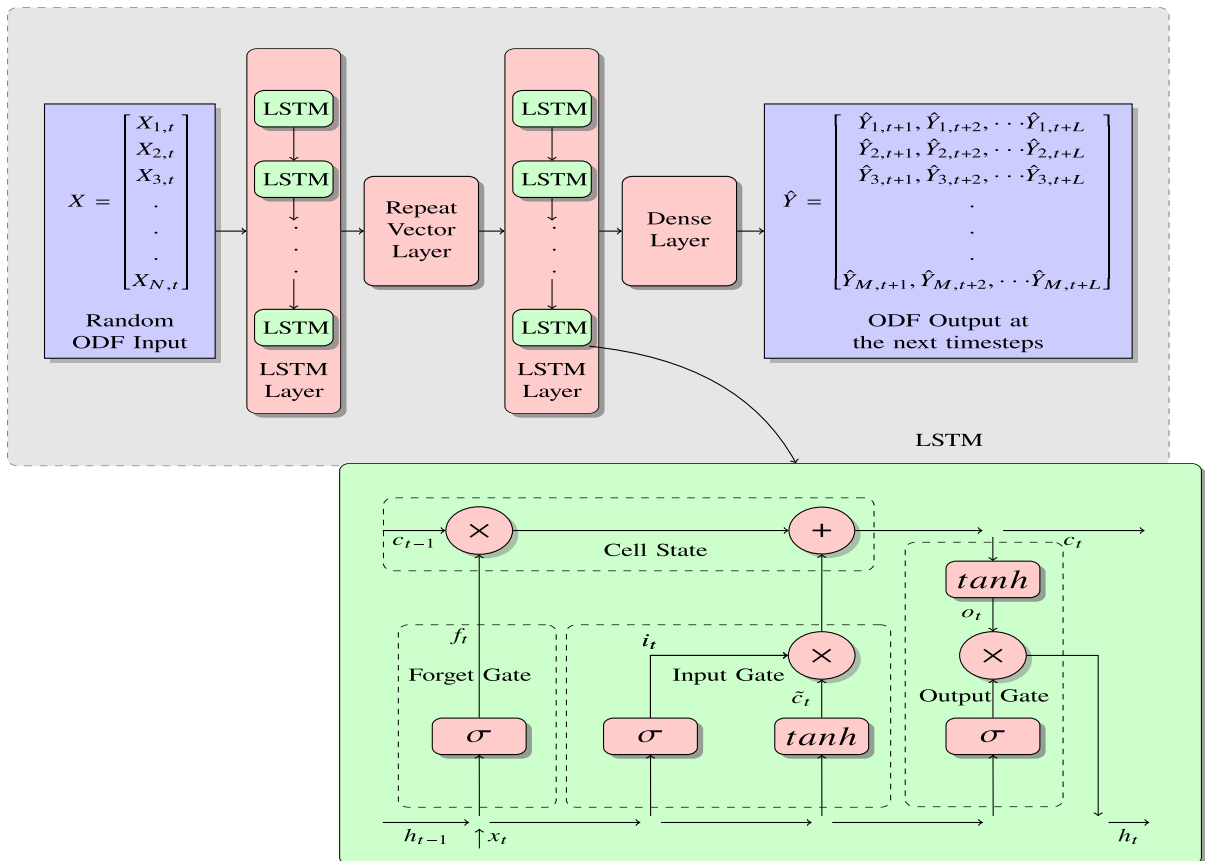


Fig. 2 LSTM architecture adaptation for a multiple parallel time-series problem to predict the evolution of ODFs and the internal block structure of an LSTM cell.

ODF features. On the other hand, for the second problem that explores the optimum crystal plasticity parameters of Ti-7Al alloy to match the final texture data, the first LSTM layer inputs the slip and twin system parameters. The final LSTM layer takes the output of the previous layer and predicts the final value through the dense layer. The reason for utilizing LSTM for the second problem is the realization of the dependence of the output parameters. Given the inherent relationships and dependencies in the output, we deduced that LSTM, with its unique ability to retain information over extended sequences [50], would be better suited for this problem. In addition, the consideration of both input and output parameters as a series provides a general framework that can be extended to address other time-dependent problems using LSTM. Building such a general time-dependent surrogate model framework can be leveraged to predict the output (and the evolution of the output) at different time step(s).

The procedure of LSTM for the first problem is as follows: The predicted output $\hat{Y}$ for each time step, based on the strain rate and the initial ODF, is evaluated via a loss function where it is compared with the actual ODF (Y) for the corresponding time step obtained from the physics-based simulator. The loss function is modified to incorporate the physics-based normalization constraint [see Eq. (5)], which needs to be satisfied by the ODFs at each time step. Therefore, we have added an extra term in the loss function that accounts for the physics-informed loss, which becomes Total Loss = Physics-Informed Loss + Data-Driven Loss. The total loss (J) can be expressed as

$$J = \sum_{i=1}^L \left[\omega_1 (1 - q \cdot \hat{Y}_i)^2 + \omega_2 \frac{1}{M} \sum_{j=1}^M (Y_{i,j} - \hat{Y}_{i,j})^2 \right] \quad (17)$$

where the first term ensures that the accuracy search does not violate the physics-based constraint, and the second term ensures the minimum error in model prediction. As described in Fig. 2, L stands for the maximum time step of the simulation data, which is equal to 10 in the first problem. For each time step, the physical loss [loss for the physics-informed term in Eq. (17)] is calculated, and the values of each predicted and real ODF are subtracted to find the mean squared error loss. Next, the summation over 10 time steps is carried out. For the second problem, we only consider the final-step (10th-step) ODFs as output; therefore, L is set equal to 1. The weight parameters (ω_1, ω_2) are tuned to 10 and 1, respectively, to improve the prediction accuracy while concurrently satisfying the physics-informed constraint in the predicted ODF.

III. Results and Discussion

A. Problem Statement

Crystal plasticity modeling is a computational approach that investigates the mechanical behavior of crystalline materials at the microscale under large deformations caused by external forces. It considers the interactions between crystal defects, such as

dislocations, and the crystal lattice and provides insights into how materials deform under different loading conditions. However, the crystal plasticity simulations tend to be computationally intensive, demand extensive data for accurate calibration, and are sensitive to model parameters. To address these challenges, we propose to develop data-driven surrogate models informed by underlying physics (i.e., orientation space definition and evolution in our work) to offer a more efficient, versatile solution and extend the applicability to materials with limited information.

Knowledge about the texture evolution of copper (Cu) microstructures under tensile loads is crucial for applications in the aerospace industry, where copper is employed in critical components like gaskets and seals, bearings, heat exchangers, etc. [51]. Under varying loading, it is essential to understand how the microstructure evolves during tensile deformation, which aids in optimizing the material's performance and reliability in aerospace structures. Therefore, we propose to build a surrogate model for exploring the plastic deformation of Cu microstructures under tensile loads. In particular, this surrogate model predicts the texture evolution of Cu during the deformation process. The inputs of the surrogate model are defined as the initial texture (in terms of ODFs) and strain rates of the process. The outputs are the deformed textures (in terms of ODFs) in 10 time steps. Figure 3a summarizes the schematic of the first problem.

The second problem aims to explore the large-deformation behavior of Ti-7Al. To achieve this goal, estimating the slip and twin system parameters (namely crystal plasticity parameters) for Ti-7Al is vital for building a high-fidelity model that can be used to develop high-performance Ti-Al alloys in structural applications. Ti-7Al is extensively utilized in aerospace and other engineering fields due to its excellent strength-to-weight ratio [13]. However, there is no universal agreement on the crystal plasticity parameters of Ti-7Al [11]. Therefore, an accurate estimation of its crystal plasticity parameters under compressive loading conditions is essential for optimizing the alloy's deformation behavior and ensuring superior mechanical properties. This is anticipated to contribute to the advancement of materials used in critical components of aircraft and other high-stress applications. The surrogate model we develop for this problem takes slip and twin system parameters as input and the final deformed textures (in terms of ODFs) after compression as output. Next, this trained surrogate model is introduced within the optimization algorithm to calibrate the slip and twin system parameters of Ti-7Al using the experimental data of the microstructural texture. The second problem overview is also described by a flow diagram in Fig. 3b. In both cases, the training datasets are generated using the physics-based crystal plasticity simulations utilizing the constitutive model developed by Sundararaghavan and Zabarar [52].

B. Forward Model of Crystal Plasticity Simulations

The objective of this surrogate model is to predict the texture evolution and the final texture of tensile deformation using any given

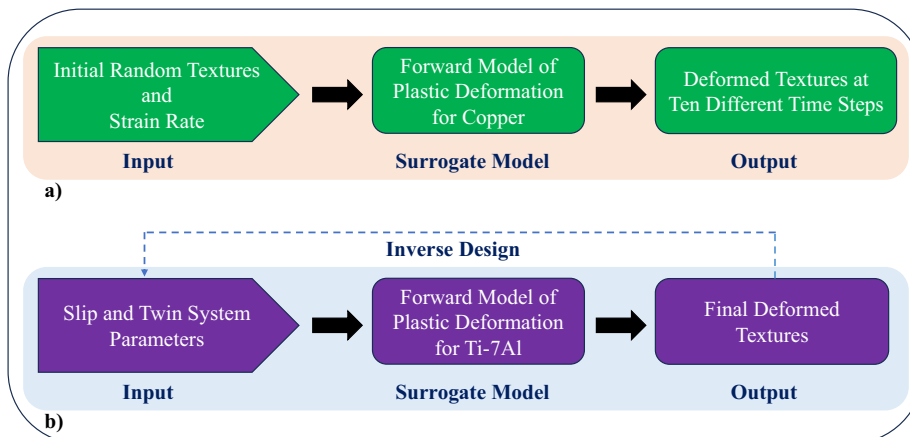


Fig. 3 Schematic of the a) forward model for texture evolution of copper microstructures under tensile loads, and b) inverse design of slip and twin system parameters of Ti-7Al for compression.

initial texture and strain rate. Copper is considered as an example material that has 76 independent ODFs. We generated 1200 training data points using the physics-based simulations from random initial textures with different strain rates varying from 0.1 to 1 s^{-1} . Each simulation took around 15–20 minutes on a desktop computing platform. The tensile force is applied for 0.1 s in each case. Training data contains the final deformed texture with nine intermediate time steps of textures in terms of ODF snapshots. Note that 200 data samples are reserved for testing the performance of the model. Among the 1000 data samples, 85 and 15% of the data are used for training and validation, respectively. The structure of the network used for training this model consists of 3 LSTM layers of 200 nodes, followed by a dense layer and wrapped by a time-distributed layer. The training is completed in 598 s on a cluster equipped with an AMD EPYC 7702 CPU clocked at 3.35 GHz using four cores. Adam Optimizer [53] was used for the training that had 2000 epochs with a batch size of 77 and a learning rate of 0.0001. The trained model is then used to test different sets of ODFs. To emphasize the significance of implementing a physics-informed constraint, the comparison between the performance of LSTM-based surrogate models is evaluated for both conditions, i.e., with and without the application of this physics-informed constraint. The training and validation accuracy of the models in terms of mean squared error is reported in Fig. 4. Convergence of the LSTM model with physics-informed loss occurs at 790 epoch (Fig. 4a). Conversely, Fig. 4b shows that the LSTM model converges faster (150 epochs) without physics-based loss. Next, the results were examined across all test cases by keeping all of the settings the same during the training phase (Fig. 5). Despite a somewhat comparable prediction quality that loosely favors LSTM without the physics-informed loss (Fig. 5b), the error related to the imposed constraint is notably diminished when employing LSTM with the physics-informed loss, as shown in Fig. 5a. The *loss* is measured for each test case as an average of predictions for all time steps. When the model is trained with the physics-informed loss, it is able to decrease the associated loss significantly, as expected, without sacrificing the prediction accuracy for the data-driven loss. The maximum deviation from the constraint for the LSTM with physics-informed loss is 1.3%, whereas it is 4.6% if the physics-informed loss is not included. Therefore, by considering the physics-informed constraint as a strict mathematical constraint, LSTM with physics-informed loss is found to achieve more consistent and feasible predictions. In addition, the predictions for exemplary ODF values by the LSTM with physics-informed loss demonstrate comparable trends as the actual ODF data even though exemplary cases include the worst cases (and also random cases) in terms of the prediction accuracy (Fig. 5c). The root mean squared error (RMSE) values of the predictions are 0.27, 0.23, 0.10, and 0.12, respectively.

Furthermore, a random initial texture is chosen to run the process simulation using both physics-based and developed surrogate models for the strain rate of 1 s^{-1} . The final texture and two intermediate steps of ODF evolution from both models are shown for comparison in Fig. 6. It is evident from Fig. 6 that the physics-informed surrogate model is able to capture the trend of ODF evolution in time with a minimum error (RMSE values are 0.44, 0.65, and 0.85 for the 3rd, 7th, and 10th steps, respectively). Moreover, the predicted ODFs in each step satisfy the volume normalization constraint of Eq. (5). It is observed that the RMSE value of the prediction is increasing with time. This is because the prediction of the next step is also based on the predicted previous step. Therefore, the error is accumulating in the further steps. In addition, these values are higher for predicting the high strain rate processes. The possible reason behind this finding is that larger changes occur in the ODF values from the initial time step to the final time step when the strain rate is high. In such cases, the surrogate model needs more data from the physics-based simulations to achieve high prediction accuracy.

C. Inverse Design of Crystal Plasticity Parameters

In our previous study [13], a high-fidelity crystal plasticity surrogate model was developed for Ti-7Al using conventional neural networks. A two-step solution was proposed to develop an inverse problem that yielded optimum crystal plasticity parameters by minimizing the difference between experimental microstructure data and the neural network predictions of the same. However, only 50 data points were generated using PRISMS-Plasticity software [54] to train the model, as the data generation was costly. Moreover, the design variables (ODFs) were required to satisfy the normalization constraints. As a result, the overall prediction quality of the conventional neural network was lower than desired. Therefore, in this study, we train the model using physics-informed LSTM, as it has shown very good potential in process modeling for copper. Even though the deformed texture of the final time step is used for training, it is also dependent on the textures of the previous steps. The findings of the current study are also compared to the previous data-driven predictions.

The material of interest, Ti-7Al, can demonstrate slip and twin deformation behavior. Therefore, to run crystal plasticity simulations, the slip and twin parameters are defined as the inputs to the software. These parameters have possible ranges (shown in Table 1) that were previously obtained using the experimental true stress-strain curve data for compression in the preliminary studies of our group [11,13].

Like the forward process modeling of Cu, we have used the same process simulator with the constitutive model developed by Sundararaghavan and Zabarar [52] to generate the training data samples for a compression process. The crystal plasticity parameters are defined as the input, and the ODFs are the output. A total of 50 random combinations of the slip and twin parameters are chosen within the

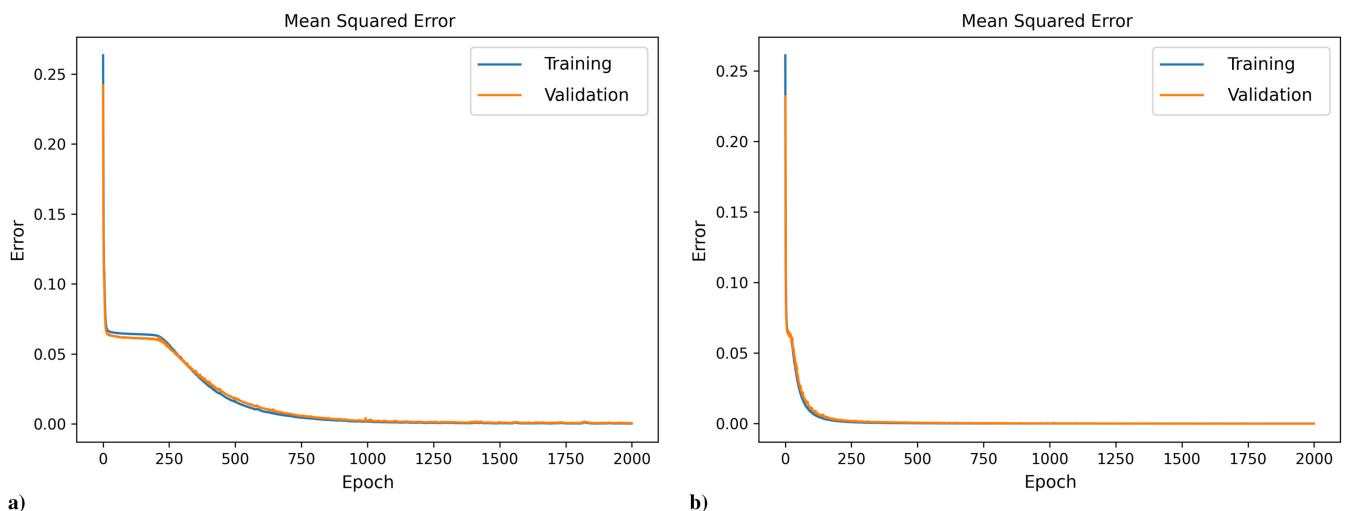


Fig. 4 Training performance of the network using LSTM a) with physics-informed loss and b) without physics-informed loss. Convergence occurs at around 790 epochs for (a) and 150 epochs for (b).

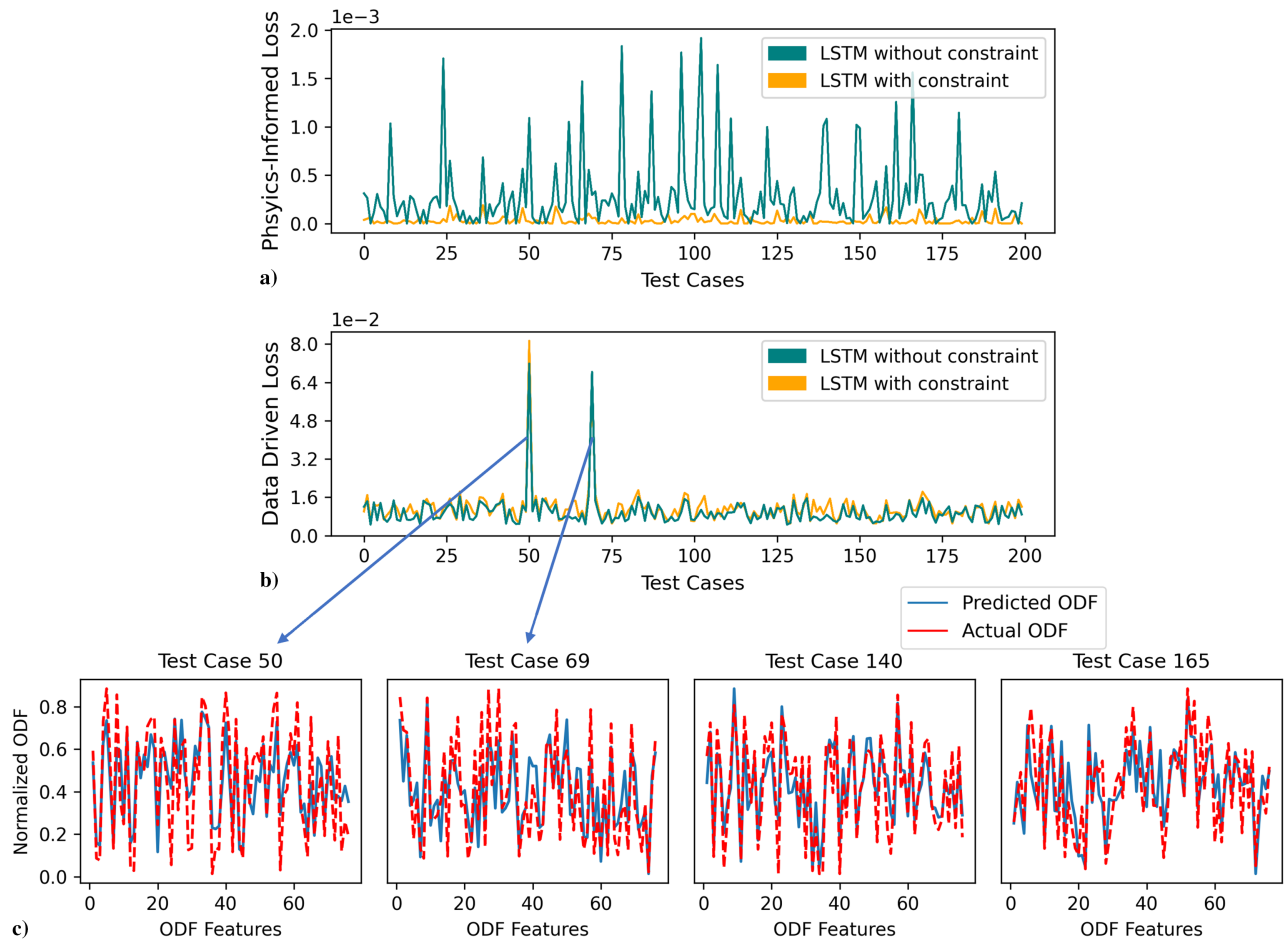


Fig. 5 Comparison between the performances of the surrogate models trained using LSTM with and without physics-informed loss. It is presented in terms of a) physics-informed and b) data-driven loss. c) To further show the performance of the surrogate model trained with physics-informed loss, two of the worst cases and two random cases are selected to visualize the normalized actual and predicted ODFs.

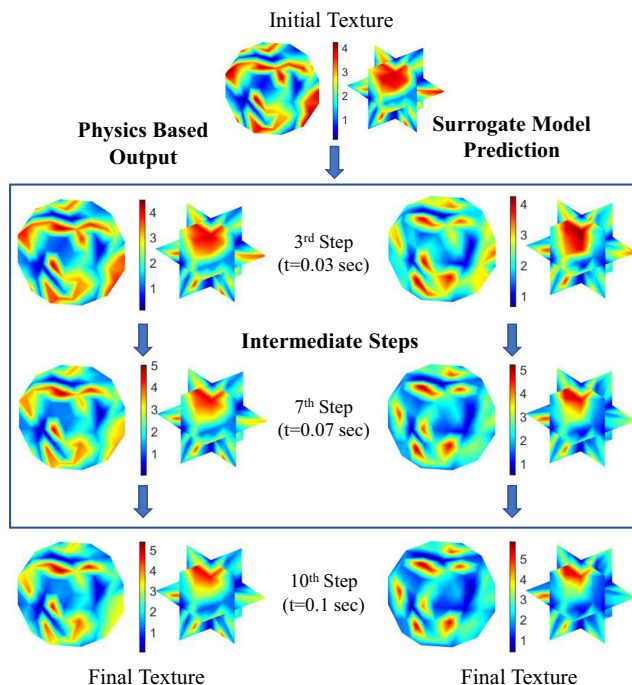


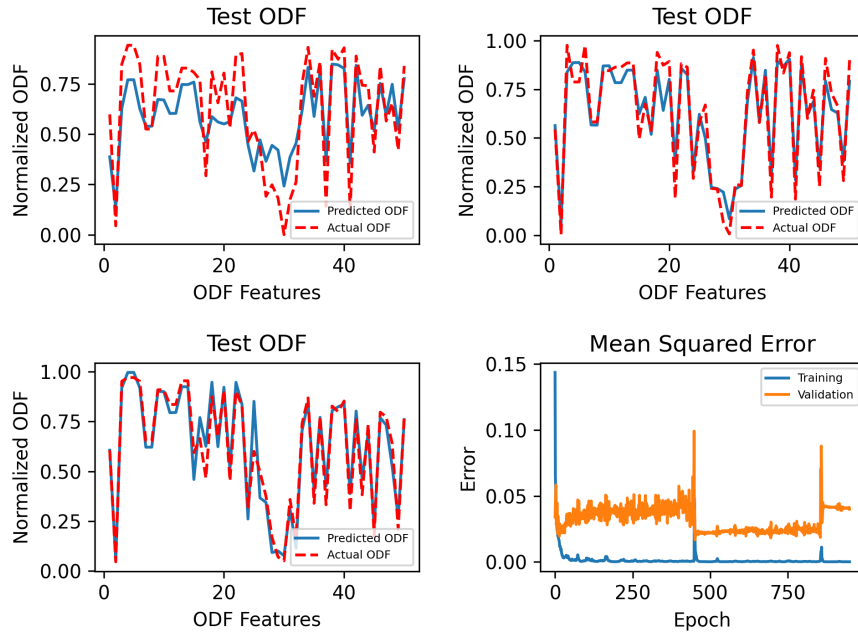
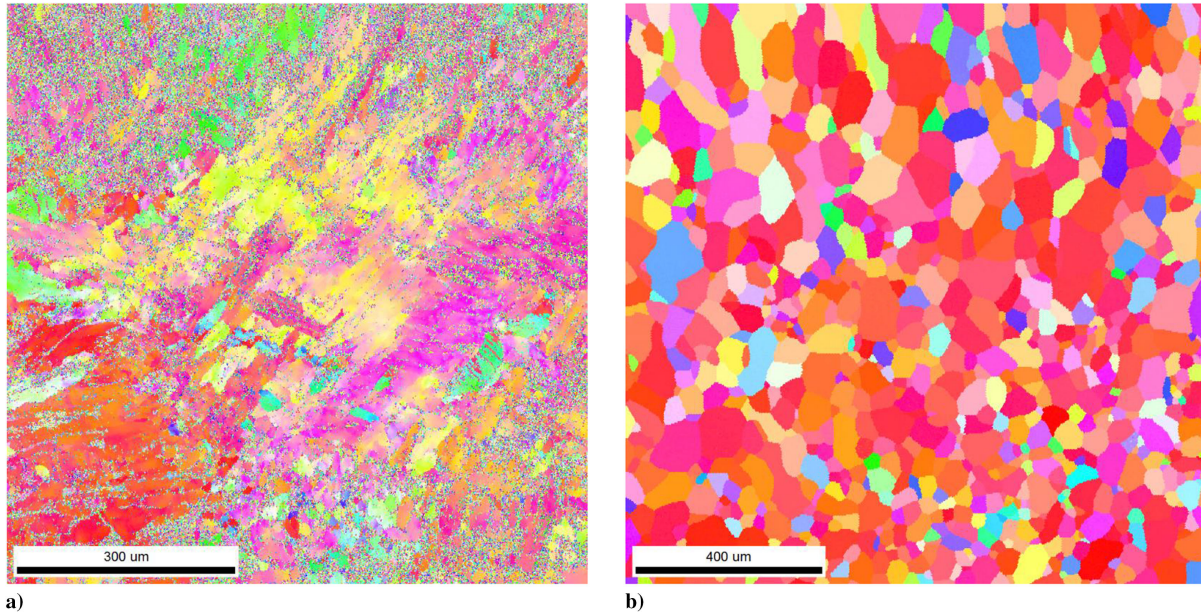
Fig. 6 Comparison of the ODFs in Rodrigues orientation space at different time steps obtained by physics-based model and physics-informed ML model.

given ranges in Table 1 to generate training data. Each physics-based crystal plasticity simulation of Ti-7Al took around 5–6 h, depending on the values of slip and twin parameters on a desktop computational platform. Using this small dataset, another physics-informed surrogate model is trained using the LSTM network in Fig. 2. In this case, the structure consists of two LSTM layers of 100 nodes, followed by a dense layer and wrapped by a time-distributed layer. The training is completed in 276 s on a cluster equipped with an AMD EPYC 7702 CPU clocked at 3.35 GHz using four cores. The combinations are split into 80%–10%–10% as training, validation, and test sets where the learning rate is set to 0.01. To prevent overfitting, training is stopped early at the 900th epoch, along with a batch size of only 2. Unlike the first approach, in this case, the input layer has 20 features (slip and twin system parameters), and the output layer has 50 independent ODFs of Ti-7Al. Similar to the previous model, these ODFs need to satisfy the volume normalization constraint, which is incorporated into the training by customizing the loss function [see Eq. (17)].

The accuracy of the trained model in terms of comparison between the actual and predicted normalized ODFs and mean squared error for training and validation data are reported in Fig. 7. The first test case shows a little discrepancy between the actual and predicted ODFs, with an RMSE value of 0.27. However, the other two cases exhibit promising agreement between the actual and predicted ODFs, with RMSE values of 0.03 and 0.05, respectively. Next, we estimate the optimum crystal plasticity parameters that provide the best match with the given experimental ODF values. The ODF values for the experimental microstructures, shown in Fig. 8, are derived from the Euler angle information using the

Table 1 Optimum ranges for the crystal plasticity parameters for compression [11,13]

Slip system	s_0 , MPa	h_0 , MPa	s_s , MPa	a
Basal $\langle a \rangle$	[200, 349.95]	[200, 299.5]	[1500, 1784.2]	[1.3, 2.0149]
Prismatic $\langle a \rangle$	[220, 399.33]	[200, 299.5]	[1500, 1784.2]	[1.3, 2.0149]
Pyramidal $\langle a \rangle$	[900, 1199.7]	[200, 299.5]	[1500, 1784.2]	[1.3, 2.0149]
Pyramidal $\langle c + a \rangle$	[800.2, 1199.1]	[200, 299.5]	[1500, 1784.2]	[1.3, 2.0149]
Twinning	[609.88, 999.28]	[800.12, 1110]	[1500, 1784.2]	[3.6584, 3.9998]

**Fig. 7** Comparison of the actual and predicted normalized ODFs for three different test cases along with the convergence of mean squared error for both training and validation data.**Fig. 8** Experimental EBSD image of the Ti-7Al alloy sample at 13.5% compressive strain a) after compression b) after recrystallization [55].

closest simplex search technique to group the orientation information [56]. The EBSD images of the Ti-7Al sample are obtained at 13.5% compressive strain [55]. Figure 8a was taken after the end of the compression process, and Fig. 8b was collected after recrystallization of the microstructure. Next, an inverse optimization problem is defined to solve this problem. Therefore, the objective of this

optimization problem is to minimize the mean squared error between the predicted and experimental ODFs.

For the first problem, we solved the optimum slip and twin system parameters (see Table 2) that can provide the closest match of the optimum ODFs to the given experimental ODFs after compression. This problem was also solved by the conventional neural network

Table 2 Optimum slip and twin system parameters obtained from PINN-based inverse optimization providing the best match with the experimental ODFs after compression

Slip system	s_0 , MPa	h_0 , MPa	s_s , MPa	a
Basal $\langle a \rangle$	316.04	202.8	1620.9	1.63
Prismatic $\langle a \rangle$	309.8	277.9	1650.1	1.84
Pyramidal $\langle a \rangle$	940.2	257.3	1538.9	1.4
Pyramidal $\langle c + a \rangle$	971.2	269.6	1600.8	1.62
Twinning	936.2	975.8	1707.3	3.8

before [13]. However, the prediction accuracy was found to be insufficient. The RMSE value of the current prediction for all 50 ODFs is 0.41. However, this value was 1.24 in the previous study (almost three times higher), which was a purely data-driven neural network framework [13]. Figure 9 reports the PINN-predicted ODFs in the Rodrigues orientation space, along with the previous prediction by the conventional neural network and the experimental ODFs for comparison. It is evident that the physics-informed LSTM has improved prediction accuracy compared to the conventional neural network with the incorporation of the problem physics. Therefore, the calibrated slip and twin system parameters of Ti-7Al shown in Table 2 can be used for the crystal plasticity simulations to understand the alloy's large deformation behavior under compression loads in the future.

Next, we calibrated the crystal plasticity parameters, shown in Table 3, using inverse design optimization that aimed to find the best-matching ODFs with the known experimental ODFs after recrystallization of the deformed microstructure. Even though the temperature was not assigned as an independent design variable during the training of the surrogate model, the performance of the trained model was also assessed for the experimental texture data after recrystallization. The optimum ODFs from the prediction and the experimental ODFs in the orientation space are displayed in Fig. 10. The physics-informed surrogate model also performs well in this case; however, the RMSE value of the prediction is 0.67, which is larger than in the previous case. The incorporation of the temperature effect in the surrogate model is expected to decrease this error value and can be explored in the future.

The surrogate models in both cases reduce the computational time significantly, from hours to only an average of 5 μ s for a single simulation, leveraging the accuracy slightly, as reported. Considering the relatively small error values for the surrogate models of both problems, these several orders of magnitude savings in computing times through the use of the surrogate model make it a desirable approach to build the presented framework simulating microstructural texture evolution under processing. The LSTM is

Table 3 Optimum slip and twin system parameters obtained from PINN-based inverse optimization that provide the best match with the experimental ODFs after recrystallization

Slip system	s_0 , MPa	h_0 , MPa	s_s , MPa	a
Basal $\langle a \rangle$	325.5	288.02	1782.4	1.91
Prismatic $\langle a \rangle$	247.8	286.9	1706.7	1.53
Pyramidal $\langle a \rangle$	1142.1	293.4	1605.03	1.52
Pyramidal $\langle c + a \rangle$	942.7	272.3	1585.7	1.55
Twinning	962.8	915.8	1549.07	3.75

also found to be powerful, first owing to the time dependency of the problem and, second, the inherent relationship between the ODF features. In addition, the introduction of physics-informed loss showed that the constraint error decreased sharply, and over all test cases, a cumulative 95.1% decrease was achieved without causing a change in MSE. Though the surrogate model predictions are found to improve the prediction accuracy compared to the previous data-driven ML results, there are still errors potentially arising from i) epistemic uncertainty associated with the crystal plasticity simulations and ii) aleatoric uncertainty arising from the experimental measurements of the microstructural texture. With more generated data, the model is anticipated to be improved further in the future. Overall, this study can be useful for material deformation modeling and design for critical applications.

IV. Conclusions

This study presents the application of physics-informed neural networks in microstructure-sensitive materials design. The developed physics-informed LSTM network provides very good accuracy for predicting the texture evolution of copper under the tensile deformation process, with the lowest RMSE value of 0.44 for all 76 ODFs. In another problem, to identify the crystal plasticity parameters of Ti-7Al given the after-deformation experimental texture, PINN shows promising results for the crystal plasticity parameters calibration as its RMSE value is three times smaller than the RMSE value of the conventional neural network prediction. Therefore, this work has provided insight for future works that would involve the crystal plasticity modeling of metals by considering the uncertainty of the microstructures using a physics-informed neural network. The developed surrogate models are demonstrated to capture the microstructural texture evolution in different time steps of different deformation processes while accounting for the physics-derived design constraints of the orientation space. This methodology can further be extended in the future to predict the changes not only in the microstructural features but also in time-dependent

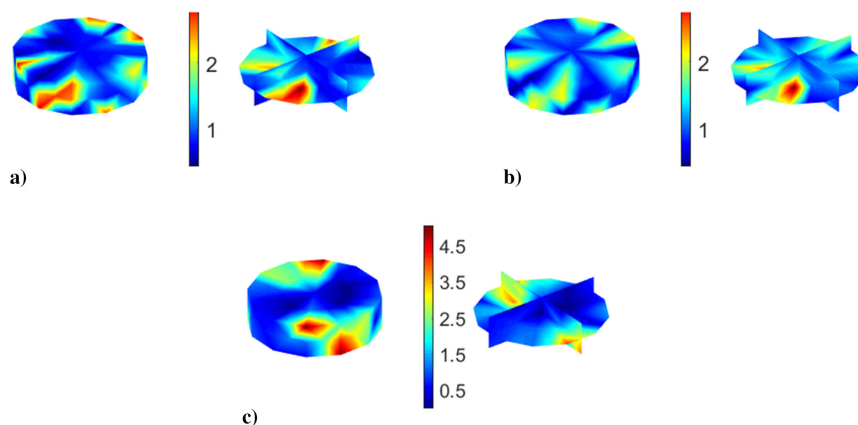


Fig. 9 ODFs in the Rodrigues orientation space: a) experimental ODFs after compression, b) ODFs predicted by PINN, and c) ODFs predicted by a conventional neural network.

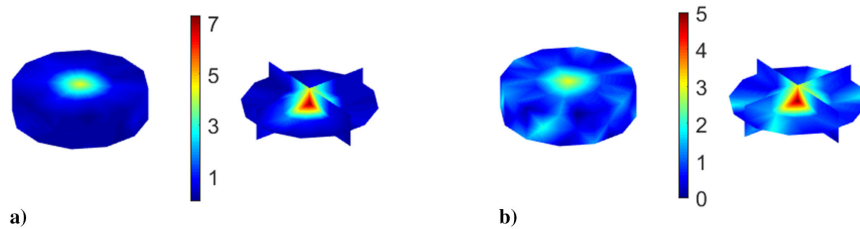


Fig. 10 ODFs in the Rodrigues orientation space: a) experimental ODFs after recrystallization and b) ODFs predicted by PINN.

material properties (e.g., stress/strain fields of microstructures) during the plastic deformation of metals.

Acknowledgments

The authors acknowledge the financial support received from the Air Force Office of Scientific Research (AFOSR) through the Young Investigator Program (YIP) under Grant FA9550-21-1-0120 (Program Officer: Ali Sayir) and from the National Science Foundation's (NSF) Civil, Mechanical and Manufacturing Innovation Award 2053840 (Program Director: Kathryn Jablowski). The authors would like to thank John Allison from the Materials Science and Engineering Department at the University of Michigan for providing experimental microstructure data of the Ti-7Al alloy.

References

- [1] Acar, P., "Machine Learning Approach for Identification of Microstructure-Process Linkages," *AIAA Journal*, Vol. 57, No. 8, 2019, pp. 3608–3614.
<https://doi.org/10.2514/1.J058244>
- [2] Allison, J., Backman, D., and Christodoulou, L., "Integrated Computational Materials Engineering: A New Paradigm for the Global Materials Profession," *Journal of the Minerals, Metals & Materials Society*, Vol. 58, No. 11, 2006, pp. 25–27.
<https://doi.org/10.1007/s11837-006-0223-5>
- [3] Raissi, M., Perdikaris, P., and Karniadakis, G. E., "Physics-Informed Neural Networks: A Deep Learning Framework for Solving Forward and Inverse Problems Involving Nonlinear Partial Differential Equations," *Journal of Computational Physics*, Vol. 378, Feb. 2019, pp. 686–707.
<https://doi.org/10.1016/j.jcp.2018.10.045>
- [4] Mueller, T., Kusne, A. G., and Ramprasad, R., "Machine Learning in Materials Science: Recent Progress and Emerging Applications," *Reviews in Computational Chemistry*, Vol. 29, April 2016, pp. 186–273.
<https://doi.org/10.1002/9781119148739.ch4>
- [5] Warde, J., and Knowles, D. M., "Use of Neural Networks for Alloy Design," *ISIJ international*, Vol. 39, No. 10, 1999, pp. 1015–1019.
<https://doi.org/10.2355/isijinternational.39.1015>
- [6] Liu, R., Kumar, A., Chen, Z., Agrawal, A., Sundararaghavan, V., and Choudhary, A., "A Predictive Machine Learning Approach for Microstructure Optimization and Materials Design," *Scientific Reports*, Vol. 5, No. 1, 2015, pp. 1–12.
<https://doi.org/10.1038/srep11551>
- [7] Agrawal, A., and Choudhary, A., "Perspective: Materials Informatics and Big Data: Realization of the 'Fourth Paradigm' of Science in Materials Science," *APL Materials*, Vol. 4, No. 5, 2016, Paper 053208.
<https://doi.org/10.1063/1.4946894>
- [8] Liu, Y., Zhao, T., Ju, W., and Shi, S., "Materials Discovery and Design Using Machine Learning," *Journal of Materiomics*, Vol. 3, No. 3, 2017, pp. 159–177.
<https://doi.org/10.1016/j.jmat.2017.08.002>
- [9] Kondo, R., Yamakawa, S., Masuoka, Y., Tajima, S., and Asahi, R., "Microstructure Recognition Using Convolutional Neural Networks for Prediction of Ionic Conductivity in Ceramics," *Acta Materialia*, Vol. 141, Dec. 2017, pp. 29–38.
<https://doi.org/10.1016/j.actamat.2017.09.004>
- [10] Paul, A., Acar, P., Liao, W.-K., Choudhary, A., Sundararaghavan, V., and Agrawal, A., "Microstructure Optimization with Constrained Design Objectives Using Machine Learning-Based Feedback-Aware Data-Generation," *Computational Materials Science*, Vol. 160, April 2019, pp. 334–351.
<https://doi.org/10.1016/j.commatsci.2019.01.015>
- [11] Acar, P., "Machine Learning Reinforced Crystal Plasticity Modeling Under Experimental Uncertainty," *AIAA Journal*, Vol. 58, No. 8, 2020, pp. 3569–3576.
<https://doi.org/10.2514/6.2020-1152>
- [12] Karniadakis, G. E., Kevrekidis, I. G., Lu, L., Perdikaris, P., Wang, S., and Yang, L., "Physics-Informed Machine Learning," *Nature Reviews Physics*, Vol. 3, No. 6, 2021, pp. 422–440.
<https://doi.org/10.1038/s42254-021-00314-5>
- [13] Hasan, M. M., Senthilnathan, A., and Acar, P., "Machine Learning Reinforced Multi-Scale Modeling of Microstructures under Uncertainties," *AIAA Scitech 2021 Forum*, AIAA Paper 2021-1140, 2021.
<https://doi.org/10.2514/6.2021-1140>
- [14] Lee, H., and Kang, I. S., "Neural Algorithm for Solving Differential Equations," *Journal of Computational Physics*, Vol. 91, No. 1, 1990, pp. 110–131.
[https://doi.org/10.1016/0021-9991\(90\)90007-N](https://doi.org/10.1016/0021-9991(90)90007-N)
- [15] Lagaris, I. E., Likas, A., and Fotiadis, D. I., "Artificial Neural Networks for Solving Ordinary and Partial Differential Equations," *IEEE Transactions on Neural Networks*, Vol. 9, No. 5, 1998, pp. 987–1000.
<https://doi.org/10.1109/72.712178>
- [16] Lagaris, I. E., Likas, A. C., and Papageorgiou, D. G., "Neural-Network Methods for Boundary Value Problems with Irregular Boundaries," *IEEE Transactions on Neural Networks*, Vol. 11, No. 5, 2000, pp. 1041–1049.
<https://doi.org/10.1109/72.870037>
- [17] Sirignano, J., and Spiliopoulos, K., "DGM: A Deep Learning Algorithm for Solving Partial Differential Equations," *Journal of Computational Physics*, Vol. 375, Dec. 2018, pp. 1339–1364.
<https://doi.org/10.1016/j.jcp.2018.08.029>
- [18] Raissi, M., Perdikaris, P., and Karniadakis, G. E., "Physics Informed Deep Learning (Part I): Data-Driven Solutions of Nonlinear Partial Differential Equations," arXiv preprint arXiv:1711.10561, 2017.
<https://doi.org/10.48550/arXiv.1711.10561>
- [19] Raissi, M., Perdikaris, P., and Karniadakis, G. E., "Physics Informed Deep Learning (Part II): Data-Driven Discovery of Nonlinear Partial Differential Equations," CoRR, abs/1711.10566, 2017, <http://arxiv.org/abs/1711.10566>.
- [20] Haghighat, E., Raissi, M., Moure, A., Gomez, H., and Juanes, R., "A Physics-Informed Deep Learning Framework for Inversion and Surrogate Modeling in Solid Mechanics," *Computer Methods in Applied Mechanics and Engineering*, Vol. 379, June 2021, Paper 113741.
<https://doi.org/10.1016/j.cma.2021.113741>
- [21] Arora, R., "Machine Learning-Accelerated Computational Solid Mechanics: Application to Linear Elasticity," arXiv Preprint arXiv:2112.08676, 2021.
<https://doi.org/10.48550/arXiv.2112.08676>
- [22] Zhang, E., Dao, M., Karniadakis, G. E., and Suresh, S., "Analyses of Internal Structures and Learning in Materials Using Physics-Informed Neural Networks," *Science Advances*, Vol. 8, No. 7, 2022, Paper eabk0644.
<https://doi.org/10.1126/sciadv.abk0644>
- [23] Fuhr, J. N., van Wees, L., Obstalecki, M., Shade, P., Bouklas, N., and Kasemer, M., "A Physics-Informed Deep Learning Framework for Inversion and Surrogate Modeling in Solid Mechanics: A Physics-Informed Deep Learning Framework for Inversion and Surrogate Modeling in Solid Mechanics," *Materialia*, Vol. 379, June 2022, Paper 101446.
<https://doi.org/10.1016/j.mtl.2022.101446>
- [24] Arora, R., Kakkar, P., Dey, B., and Chakraborty, A., "Physics-Informed Neural Networks for Modeling Rate- and Temperature-Dependent Plasticity," arXiv Preprint arXiv:2201.08363, 2022.
<https://doi.org/10.48550/arXiv.2201.08363>
- [25] Mao, Z., Jagtap, A. D., and Karniadakis, G. E., "Physics-Informed Neural Networks for High-Speed Flows," *Computer Methods in Applied Mechanics and Engineering*, Vol. 360, March 2020, Paper 112789.
<https://doi.org/10.1016/j.cma.2019.112789>
- [26] Cai, S., Mao, Z., Wang, Z., Yin, M., and Karniadakis, G. E., "Physics-Informed Neural Networks (PINNs) for Fluid Mechanics: A Review," *Acta Mechanica Sinica*, Vol. 37, No. 12, 2022, pp. 1727–1738.
<https://doi.org/10.1007/s10409-021-01148-1>

- [27] Yang, X., Zafar, S., Wang, J.-X., and Xiao, H., "Predictive Large-Eddy-Simulation Wall Modeling via Physics-Informed Neural Networks," *Physical Review Fluids*, Vol. 4, No. 3, 2019, Paper 034602. <https://doi.org/10.1103/PhysRevFluids.4.034602>
- [28] Almajid, M. M., and Abu-Al-Saud, M. O., "Prediction of Porous Media Fluid Flow Using Physics Informed Neural Networks," *Journal of Petroleum Science and Engineering*, Vol. 208, Jan. 2022, Paper 109205. <https://doi.org/10.1016/j.petrol.2021.109205>
- [29] Wessels, H., Weissenfels, C., and Wriggers, P., "The Neural Particle Method—An Updated Lagrangian Physics Informed Neural Network for Computational Fluid Dynamics," *Computer Methods in Applied Mechanics and Engineering*, Vol. 368, Aug. 2020, Paper 113127. <https://doi.org/10.1016/j.cma.2020.113127>
- [30] Cai, S., Wang, Z., Wang, S., Perdikaris, P., and Karniadakis, G. E., "Physics-Informed Neural Networks for Heat Transfer Problems," *Journal of Heat Transfer*, Vol. 143, No. 6, 2021, Paper 060801. <https://doi.org/10.1115/1.4050542>
- [31] Chen, Y., Lu, L., Karniadakis, G. E., and Dal Negro, L., "Physics-Informed Neural Networks for Inverse Problems in Nano-Optics and Metamaterials," *Optics express*, Vol. 28, No. 8, 2020, pp. 11,618–11,633. <https://doi.org/10.1364/OE.384875>
- [32] Fang, Z., and Zhan, J., "Deep Physical Informed Neural Networks for Metamaterial Design," *IEEE Access*, Vol. 8, Dec. 2019, pp. 24,506–24,513. <https://doi.org/10.1109/ACCESS.2019.2963375>
- [33] Shaier, S., Raissi, M., and Seshaiyer, P., "Data-Driven Approaches for Predicting Spread of Infectious Diseases Through DINNs: Disease Informed Neural Networks," arXiv Preprint arXiv:2110.05445, 2021. <https://doi.org/10.48550/arXiv.2110.05445>
- [34] Goswami, S., Li, D. S., Rego, B. V., Latorre, M., Humphrey, J. D., and Karniadakis, G. E., "Neural Operator Learning of Heterogeneous Mechanobiological Insights Contributing to Aortic Aneurysms," arXiv Preprint arXiv:2205.03780, 2022. <https://doi.org/10.48550/arXiv.2205.03780>
- [35] Sahli Costabal, F., Yang, Y., Perdikaris, P., Hurtado, D. E., and Kuhl, E., "Physics-Informed Neural Networks for Cardiac Activation Mapping," *Frontiers in Physics*, Vol. 8, Feb. 2020, p. 42. <https://doi.org/10.3389/fphy.2020.00042>
- [36] Misyris, G. S., Venzke, A., and Chatzivasileiadis, S., "Physics-Informed Neural Networks for Power Systems," 2020 *IEEE Power & Energy Society General Meeting (PESGM)*, Inst. of Electrical and Electronics Engineers, New York, 2020, pp. 1–5. <https://doi.org/10.1109/PESGM41954.2020.9282004>
- [37] Oszkinat, C., Luczak, S. E., and Rosen, I., "Uncertainty Quantification in Estimating Blood Alcohol Concentration from Transdermal Alcohol Level with Physics-Informed Neural Networks," *IEEE Transactions on Neural Networks and Learning Systems*, Vol. 34, No. 10, 2022, pp. 8094–8101. <https://doi.org/10.1109/TNNLS.2022.3140726>
- [38] Lütjens, B., Crawford, C. H., Veillette, M., and Newman, D., "PCE-PINNs: Physics-Informed Neural Networks for Uncertainty Propagation in Ocean Modeling," arXiv Preprint arXiv:2105.02939, 2021. <https://doi.org/10.48550/arXiv.2105.02939>
- [39] Zhang, D., Lu, L., Guo, L., and Karniadakis, G. E., "Quantifying Total Uncertainty in Physics-Informed Neural Networks for Solving Forward and Inverse Stochastic Problems," *Journal of Computational Physics*, Vol. 397, Nov. 2019, Paper 108850. <https://doi.org/10.1016/j.jcp.2019.07.048>
- [40] Dornheim, J., Morand, L., Zeitvogel, S., Iraki, T., Link, N., and Helm, D., "Deep Reinforcement Learning Methods for Structure-Guided Processing Path Optimization," *Journal of Intelligent Manufacturing*, Vol. 33, No. 1, 2022, pp. 333–352. <https://doi.org/10.1007/s10845-021-01805-z>
- [41] Honarmandi, P., Attari, V., and Arroyave, R., "Accelerated Materials Design Using Batch Bayesian Optimization: A Case Study for Solving the Inverse Problem from Materials Microstructure to Process Specification," *Computational Materials Science*, Vol. 210, July 2022, Paper 111417. <https://doi.org/10.1016/j.commatsci.2022.111417>
- [42] Zhang, R., Liu, Y., and Sun, H., "Physics-Informed Multi-LSTM Networks for Metamodeling of Nonlinear Structures," *Computer Methods in Applied Mechanics and Engineering*, Vol. 369, Sept. 2020, Paper 113226. <https://doi.org/10.1016/j.cma.2020.113226>
- [43] Alhajeri, M. S., Abdullah, F., Wu, Z., and Christofides, P. D., "Physics-Informed Machine Learning Modeling for Predictive Control Using Noisy Data," *Chemical Engineering Research and Design*, Vol. 186, Oct. 2022, pp. 34–49. <https://doi.org/10.1016/j.cherd.2022.07.035>
- [44] Pombo, D. V., Bacher, P., Ziras, C., Bindner, H. W., Spataru, S. V., and Sørensen, P. E., "Benchmarking Physics-Informed Machine Learning-Based Short Term pv-Power Forecasting Tools," *Energy Reports*, Vol. 8, Nov. 2022, pp. 6512–6520. <https://doi.org/10.1016/j.egyr.2022.05.006>
- [45] Li, X., Chen, S., Zhang, J., Gao, J., and Bai, Y., "A Physics-Informed Deep Learning Paradigm for Transient Power Angle Stability Assessment," *IEEE Journal of Radio Frequency Identification*, Vol. 6, Oct. 2022, pp. 948–952. <https://doi.org/10.1109/JRFID.2022.3213882>
- [46] Kumar, A., and Dawson, P., "Computational Modeling of FCC Deformation Textures over Rodrigues' Space," *Acta Materialia*, Vol. 48, No. 10, 2000, pp. 2719–2736. [https://doi.org/10.1016/S1359-6454\(00\)00044-6](https://doi.org/10.1016/S1359-6454(00)00044-6)
- [47] Acar, P., and Sundararaghavan, V., "Utilization of a Linear Solver for Multiscale Design and Optimization of Microstructures," *AIAA Journal*, Vol. 54, No. 5, 2016, pp. 1751–1759. <https://doi.org/10.2514/1.J054822>
- [48] Sundararaghavan, V., and Zabarar, N., "On the Synergy Between Texture Classification and Deformation Process Sequence Selection for the Control of Texture-Dependent Properties," *Acta Materialia*, Vol. 53, No. 4, 2005, pp. 1015–1027. <https://doi.org/10.1016/j.actamat.2004.11.001>
- [49] Smagulova, K., and James, A. P., "Overview of Long Short-Term Memory Neural Networks," *Deep Learning Classifiers with Memristive Networks*, Springer, New York, 2020, pp. 139–153. https://doi.org/10.1007/978-3-030-14524-8_11
- [50] Santhanam, S., "Context Based Text-Generation Using LSTM Networks," arXiv Preprint arXiv:2005.00048, 2020. <https://doi.org/10.48550/arXiv.2005.00048>
- [51] Singer, F., Deisenroth, D., Hymas, D., and Ohadi, M., "Additively Manufactured Copper Components and Composite Structures for Thermal Management Applications," 2017 16th IEEE Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems (ITHERM), Inst. of Electrical and Electronics Engineers, New York, 2017, pp. 174–183. <https://doi.org/10.1109/ITHERM.2017.7992469>
- [52] Sundararaghavan, V., and Zabarar, N., "Linear Analysis of Texture-Property Relationships Using Process-Based Representations of Rodrigues Space," *Acta Materialia*, Vol. 55, No. 5, 2007, pp. 1573–1587. <https://doi.org/10.1016/j.actamat.2006.10.019>
- [53] Chandriah, K. K., and Naraganahalli, R. V., "RNN/LSTM with Modified Adam Optimizer in Deep Learning Approach for Automobile Spare Parts Demand Forecasting," *Multimedia Tools and Applications*, Vol. 80, No. 17, 2021, pp. 26,145–26,159. <https://doi.org/10.1007/s11042-021-10913-0>
- [54] Yaghoobi, M., Ganesan, S., Sundar, S., Lakshmanan, A., Rudraraju, S., Allison, J. E., and Sundararaghavan, V., "PRISMS-Plasticity: An Open-Source Crystal Plasticity Finite Element Software," *Computational Materials Science*, Vol. 169, Nov. 2019, Paper 109078. <https://doi.org/10.1016/j.commatsci.2019.109078>
- [55] Trump, A. M., and Allison, J. E., "The Influence of Aluminum Concentration on Static Recrystallization in Alpha Titanium Alloys," *Proceedings of the 13th World Conference on Titanium*, Wiley, Hoboken, NJ, 2016, pp. 695–696. <https://doi.org/10.1002/9781119296126.ch115>
- [56] Acar, P., and Sundararaghavan, V., "Uncertainty Quantification of Microstructural Properties Due to Experimental Variations," *AIAA Journal*, Vol. 55, No. 8, 2017, pp. 2824–2832. <https://doi.org/10.2514/1.J055689>