npj | Computational Materials

www.nature.com/npjcompumats

ARTICLE OPEN

Check for updates

Mechanical behavior predictions of additively manufactured microstructures using functional Gaussian process surrogates

Robert Saunders[1,2]✉, Celia Butler[3,4], John Michopoulos[1], Dimitris Lagoudas[2], Alaa Elwany[2] and Amit Bagchi[1]

Relational linkages connecting process, structure, and properties are some of the most sought after goals in additive manufacturing (AM). This is desired especially because the microstructural grain morphologies of AM components can be vastly different than their conventionally manufactured counterparts. Furthermore, data collection at the microscale is costly. Consequently, this work describes and demonstrates a methodology to link microstructure morphology to mechanical properties using functional Gaussian process surrogate models in a directed graphical network capable of achieving near real-time property predictions with single digit error magnitudes when predicting full stress–strain histories of a given microstructure. This methodology is presented and demonstrated using computationally generated microstructures and results from crystal plasticity simulations on those microstructures. The surrogate model uses grain-level microstructural descriptors rather than whole microstructure descriptors so that properties of new, arbitrary microstructures can be predicted. The developed network has the potential to scale to predict mechanical properties of grain structures that would be infeasible to simulate using finite element methods.

npj Computational Materials (2021)7:81 ; https://doi.org/10.1038/s41524-021-00548-y

INTRODUCTION

The process used to manufacture a material governs its morphological structure, which in turn drives the values and spatial distributions of the properties of the processed material and consequently its performance. Process–structure–property (PSP) relational linkages are necessary for designing, developing, and tailoring a material to exhibit desired properties, and ultimately performance, for a targeted application. Establishing PSP linkages typically involves building and testing materials from a given process until the desired properties are achieved. However, the process of generating and representing the data needed to establish these PSP linkages is often very time intensive as it requires extensive experimentation and/or complex, multiscale simulations[1–3]. Furthermore, these processes usually generate large amounts of data that present analysis and management difficulties. Recent advances in the field of data science and data analytics have made managing, interpreting, and extracting information from “big data” a more tractable activity[4,5]. The process of applying data science principles to materials science and engineering is referred to as *materials informatics*[6,7] and typically involves the use of machine learning (ML) and artificial intelligence (AI) techniques[8–10]. The advancement and adoption of materials informatics has provided notable potential to streamline and accelerate the determination of process–structure[11], structure–property[12,13], and even full PSP[14] linkages in conventionally manufactured materials.

One of the most prominent, emerging fields that call for PSP linkages and materials informatics is additive manufacturing (AM). The term AM represents a collection of advanced manufacturing processes used to build three-dimensional (3D) objects by progressively adding materials in a point by point, line by line, and utimately a layerwise fashion[15]. AM has gained significant popularity in both experimental[15,16] and computational[17] domains recently as it has the potential to remove many of the design constraints imposed by traditional manufacturing processes[18]. However, the texture morphologies produced by AM, particularly in metals at the microscale, can vastly differ from their conventionally manufactured counterparts[19]. These differences often lead to undesired (or unexpected to say the least) properties in the finished part. With adequate understanding PSP linkages in AM, these differences can be used to enhance the resulting properties to previously unattainable levels[19]. As with conventional materials, many works have attempted to understand PSP relational linkages in AM using high fidelity, multiscale simulations[20,21], which can achieve high accuracy predictions. However, these simulations are typically computationally expensive, making them better suited for understanding the underlying physics rather than being used for rapid production and/or qualification[22]. This challenge can be addressed through proper use of materials informatics and ML[23,24].

The emphasis in the current work is on understanding and generating ML-based structure–property linkages from simulated AM microstructures coupled with crystal plasticity finite element (CPFE) simulations. The CPFE method is a powerful tool for modeling the elastoplastic mechanical response of anisotropic, heterogeneous, polycrystalline aggregates by taking into account the effects of various microstructural features[25,26]. There are a number of ways in which an ML model can be trained to represent a CPFE model and link structure to properties. One method is to use deep learning (DL) to learn the constitutive model response[27–29]. This method is applicable beyond CPFE and can incorporate true physical constraints of the problem during learning. However, DL requires a large training data set that can be infeasible to generate using the computationally CPFE model. Additionally, it is not clear whether this method will yield acceptable performance when applied to polycrystalline microstructures, especially those with the complexity of AM microstructures, as this has not been addressed in existing literature studies. An alternative method to learning the constitutive model is to use microstructural features and directly relate them to certain quantities of interest (QoIs)[30–33]

[1]US Naval Research Laboratory, Washington, DC, USA. [2]Texas A&M University, College Station, TX, USA. [3]Synopsys Northern Europe Ltd, Exeter, UK. [4]University of Exeter, Exeter, UK. ✉email: robert.saunders@nrl.navy.mil

(e.g., elastic modulus, yield strength) or the full stress–strain behavior[34–36]. In an ML framework, it is conceptually straightforward to relate microstructural features directly to QoIs and this is very useful to characterize a given material or fit a given constitutive model. However, in characterizing the material by only a few QoIs a significant amount of information about the stress–strain history is lost and the constitutive model must be chosen a priori. It is anticipated that relating the microstructure morphological features to the full stress–strain history will provide benefits over just predicting certain QoIs, but it is conceptually more difficult to predict the stress–strain response. Liu and Wu[34] propose a DL-like approach termed deep material network (DMN) where phases in a representative volume element (RVE) are characterized and then propagated through homogenization and rotation operations. The operations are done such that the analytical characterization, homogenization, and rotation act nearly identically to the operations involved in a conventional artificial neural network (NN). Another approach by Frankel et al.[35,36] directly implements a hybrid convolution, long short-term memory recurrent NN (ConvLSTM) to process a microstructure image and predict its full, spatially resolved stress–strain history. While the approaches of both methods differ substantially, both are able to show very high prediction accuracy on withheld data. However, both are based on DL, which as previously mentioned, has a large training data requirement. In the DMN and ConvLSTM models, hundreds of RVEs were generated for training, validation, and testing. In those works, generating hundreds of cases was feasible due to the relative simplicity of the constitutive models, dimensionality of the problem, and microstructures being examined. As mentioned before, AM microstructures do not exhibit such simplicity so the generation of hundreds of simulations for training will be prohibitively time consuming, even on the most advanced high-performance computing systems. Additionally, DL-based models do not have an inherent uncertainty quantification method.

The recognition of the fact that CPFE models are displacement driven and the outputs of stress and strain are derived, continuous functions offers an opportunity for a new approach. Functional data analysis (FDA) is an area of statistics, which handles data that reside in an infinite dimensional space (i.e., functions such as continuous time series data)[37]. As with traditional statistical methods, FDA has two methodologies, parametric[38] and non-parametric modeling[39,40]. The latter set of methods deals with the case of modeling infinite dimensional functional data using non-parametric methods, which also follow a general infinite dimensional assumption. These methods are thus applicable to Gaussian processes (GPs) among other non-parametric methods. A number of authors have studied GPs with functional data and shown success in developing a functional predictive capability[41–46]. Li et al.[47] even recognized the applicability of the functional Gaussian process (fGP) to AM thermal process simulations. As already pointed out, in CPFE a function (i.e., displacement) can be related to another function (i.e., stress/strain), but there is an additional requirement for the existence of a set of scalar parameters, such as grain morphology descriptors and constitutive model parameters that do not change with the displacement. The drawback of the fGP models developed in the previous works is that they are restricted to function-on-scalar (i.e., functional input, scalar output) or function-on-function GP regression, but the current problem requires an approach to model function-on-mixed scalar and functional data. Recent developments by Wang and Xu[48] have addressed this restriction and allow for mixed scalar and functional input variables along with functional and/or scalar output variables.

In this work, an fPG framework is developed based on the fGP model of Wang and Xu[48] for predicting the stress–strain behavior of AM microstructures as they are related to microstructural morphology features. The GP-based system provides a fast, flexible, less data intensive alternative to existing DL methods and as a natural outcome, provides a predictive mean and variance for the stress–strain history. Additionally, GP-based models, such as the one developed here, are easily generalizable to multi-output[49] and/or multi-fidelity[50–52] variants. The framework developed herein has the additional novelty of predicting stress–strain history on a per grain basis, meaning that the microstructures used for training can be much smaller (i.e., fewer grains) than the microstructures to be approximated. The development of this framework will be shown and then trained using simulated data. The framework will then be applied to previously unseen microstructures generated by the same method as the training/testing data. Finally, the fGP network will be used to demonstrate how grain size and shape influence mechanical properties without the use of costly CPFE models.

RESULTS

fGP framework

The set of inputs needed for a CPFE model are the uniform kinematic displacement boundary conditions ($\boldsymbol{u}$, vector of functional variables) applied to the faces of an RVE over the duration of the simulation, constitutive model parameters (θ, scalar variables) that define the material behavior, and the microstructure morphology (non-functional variables, i.e., scalar, vector, or tensor variables). Additionally, a loading parameter (λ, scalar functional variable), such as amplitude over time, is used to incorporate history dependence into the model for situations where displacement or other quantities may be non-unique or non-monotonic (e.g., loading–unloading experiments). During each time increment of the CPFE simulation, a step in displacement is taken based on the value specified by the loading parameter, and along with the previous state of stress and strain in an element, a new element strain is computed followed by a stress update for the element in the current increment. The output of this process at the end of the simulation is a stress–strain curve for each of the six stress/strain components at each finite element in the simulation. The stress and strain outputs at each element can be taken as is or processed further to obtain values such as equivalent strain and equivalent (von Mises) stress. These equivalent values, or the individual components, can then be homogenized over the whole RVE, over individual grains, or other subsets of the RVE. The process as described is shown in the directed graph of Fig. 1 and the same process can be emulated using fGPs. The same inputs to the CPFE model can be used as input to the fGP described in section. The loading parameter and displacement are treated as functional inputs while the constitutive model parameters and microstructural features are treated as non-functional inputs. Note that the model as defined uses displacement as the driving deformation mechanism, but this could equivalently be replaced with a specified force or traction on an RVE face. These inputs are then used to train an fGP model that predicts the functional equivalent strain (ϵ, denoted strain from hereon), and this in turn is used alongside the previous inputs to train a second fGP, which predicts the functional equivalent stress (σ, denoted stress from hereon). The choice here to use equivalent stress and strain was made in order to obtain a scalar valued function that considers all components of stress/strain. However, this choice is inconsequential and any individual component of stress/strain could have also been used. In fact, all components of stress/strain could be considered by either training one fGP network per component or by modifying the fGP normality assumption to instead follow a multivariate normal. Further discussion on this extension is omitted as the implementation of a multivariate fGP is beyond the scope of this work.

In the illustrated graphical network, any set of microstructural features, such as those found by Mangal and Holm[53] and those

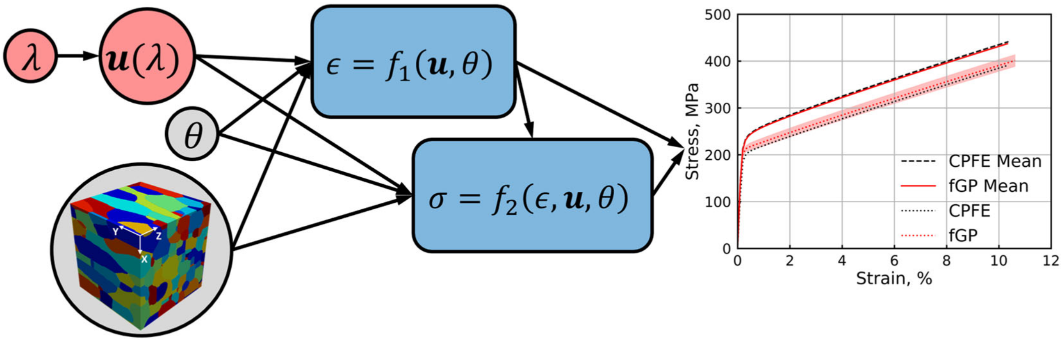


Fig. 1 fGP graphical network. A functional Gaussian process graphical network, which uses a loading parameter (λ), displacement ($\boldsymbol{u}$), constitutive model parameters (θ), and the microstructure morphology to predict stress (σ) and strain (ϵ) in each grain and a mean response over a whole RVE.

discussed by Bostanabad et al.[54], that describe the RVE can be used. In AM, microstructures in many cases exhibit epitaxially grown columnar grains with different grain sizes and aspect ratios[55]. As such, in this work, a grain size- and shape-dependent CPFE model[56] is used to generate the needed crystal plasticity data. It follows that the features needed to describe grain size and shape, such as equivalent spherical diameter and grain volume, be used to represent the microstructural RVE as a set of non-functional scalar parameters. In general, this feature representation is done at the level of the whole RVE, but in this work, the feature representation is done at the level of each grain in an RVE. This, first, means that each time-consuming computationally intensive CPFE simulation results in multiple stress–strain curves for each grain, rather than a single stress–strain curve for the whole RVE. This increases the amount of data available for training and testing the fGP models, which will help improve the predictive capabilities of the models. The drawback to this method of data collection is that the effect of boundary conditions and grain interactions is not considered. However, as will be shown in the following sections, these effects do not appear to significantly hinder the performance of the trained fGP model when given a sufficient amount of training data. Second, working at the level of the individual grain helps to directly relate grain size and shape features to the output stress–strain curve, rather than using distributions of grain size and shape for a whole RVE. By ignoring microstructure grain distributions and working with the individual grains, there are two means by which uncertainty is reduced. First, uncertainty in the microstructure feature distributions is effectively uncoupled from the mechanical property prediction, since there is typically not uncertainty associated with individual grains in a microstructure from the process–structure linkage. Second, by predicting individual grain behavior and then homogenizing, the variance of the predictive distribution, in general, is decreased since the variance of the mean decreases with sample size. Additionally, using the homogenized RVE response tends to mask the effect of small and elongated grains, which in the chosen constitutive model generally have higher stresses than large and equiaxial grains. This point is illustrated in Fig. 2 where some individual grains within a given RVE can have a stress of more than three times that of the homogenized RVE value at the same equivalent strain.

Network training and evaluation

To train the network in Fig. 1, microstructural RVEs must be generated to be processed by the CPFE model. In AM, a single representative microstructure that is capable of representing the whole microstructure is generally not possible to construct. However, RVEs can be constructed that contain a range of features (grain sizes, shapes, orientations, etc.) seen in the whole AM microstructure and that can be considered representative of the overall bulk material. As such, in this work, an RVE is defined as a volume element that contains a set of features representative of a larger AM build made with similar process parameters. For network training, 50 microstructural RVEs containing approximately 100 grains each are generated using a continuum diffuse interface model (CDIM)[57] and then meshed using Simpleware ScanIP (Synopsys, Mountain View, CA, USA). Complete details of the data generation process are given in the Methods section Data generation.

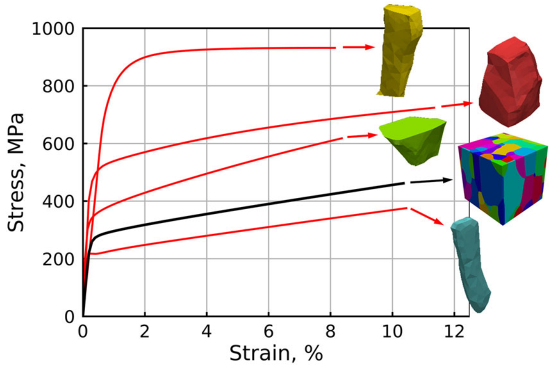


Fig. 2 Homogenized RVE vs. individual grain stress–strain. Comparison of homogenized RVE stress–strain behavior (black line) compared to individual grain behaviors from the same RVE (red lines) demonstrating how extreme gain behaviors can be masked through homogenization.

The generated RVEs contain features representative of those seen in single track AM microstructures[58] and a selection of the considered RVEs is shown in Fig. 3a. While the RVEs may not strictly resemble typically AM microstructures, they do provide a range of features, such as grain shape and size variations, that are seen in single track AM microstructures and that are needed to train the fGP network. The generated RVEs are cubic and have dimensions in the range of 0.1–0.8 mm^3. Periodic boundary conditions are specified for all faces along each axis and the loading parameter here is linear with time and monotonically increases from 0 to 10% of the RVE edge length. The displacement is mapped such that it results in a linear, monotonically increasing displacement along the *Y*-axis in the *Y*-direction. The model setup represents the loading portion of a uniaxial tension test. If more complex loading behavior was desired, a simple change in definition of the loading parameter (amplitude) and $\boldsymbol{u}$ (direction) could accomplish that. The loading parameter definition will be valid as long it is a real, unique, continuous function and $\boldsymbol{u}$ simply maps that amplitude to a specific direction on an RVE face (i.e., loading the *Y*-axis in the *Y*-direction is tension/compression and in the *X*- or *Z*-directions is shear). For instance, to address

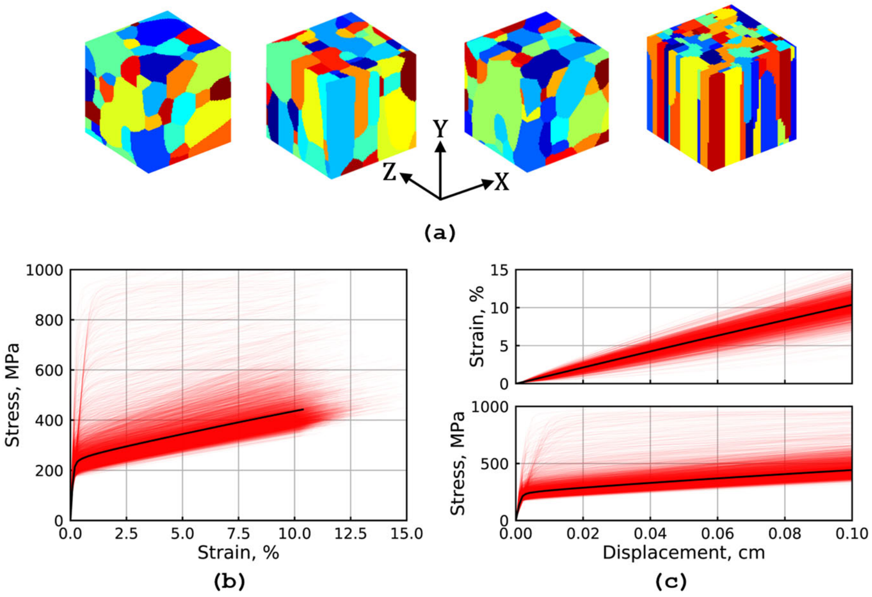


Fig. 3 fGP training data. a Selection of four RVEs from the 50 generated and used for training. **b** Stress–strain curves of each grain (red lines) in the training data set along with the mean response of all the training data (black line). **c** Training stress and strain data shown against the commanded displacement.

loading–unloading, the loading parameter would be specified as increasing (with no requirement on linearity) from some time t_0 until a future time t_1 then decreasing until another future time t_2. This type of specification could be extended to capture any non-monotonic or non-proportional behavior including cyclic or hysteretic behavior. Uniaxial tension is chosen here for simplicity and demonstration of the fGP network concept. The properties of 316L stainless steel are used for the constitutive model.

Running the simulations, extracting the stress–strain curves for each grain, and using 70% of the curves for training with the other 30% withheld for model evaluation, yield the data shown in Fig. 3b, c. The training and test sets are chosen at random. The mean strain of all these grains is around 10%, as expected based on the displacement magnitude and approximate size of the RVEs, with a stress of approximately 450 MPa at that strain. The range of strain for individual grains is between 8 and 15% with stress in the range of 325 MPa on the low end and, on the high end, some grains exceeding 1500 MPa. The importance of capturing grain stresses and strains far from the mean is based on the expectation that many grain with behaviors far from the mean will tend to have a high stress or strain energy density and will have a high probability of being a failure initiation site. Therefore it is crucial to be able to predict their behavior. This also gives reason as to why it is necessary to train the fGP framework on a per grain basis.

For the purpose of training the fGP models, only the properties in the constitutive model that are directly dependent on grain size and shape are considered. These are the grain yield strength, initial strain hardening modulus, and grain boundary resistance. Note that other constitutive model parameters, such as the stiffness tensor components, could readily be used in the parameter set but they are omitted here for simplicity. Additionally note that any derived constitutive model quantities (e.g., slip system strength, material axis rotation) are not explicitly considered, rather they are implicitly captured in the fGP since they will manifest as changes in the final stress–strain behavior. The microstructural features used to represent each grain are the grain total volume, three radii of an ellipsoid used to approximate the grain shape, and three angles representing the grains orientation relative to a defined, global axis. This set of morphological features is chosen as they all can be directly and simply related to the size, shape, and orientation of the grains in the AM microstructure. As with the constitutive model parameters, additional morphological parameters (e.g., texture, crystallographic orientation, Schmid factor) could be considered but are not implemented here for simplicity and interpretability. The selection of the features chosen for this work are specific for the AM process and the constitutive model used. In other manufacturing processes, other features such as Schmid factor may be more important or more relevant and these could be considered in the fGP in those instances. Furthermore, in ML, choosing a large number of features can be detrimental to model performance as it results in more hyper-parameters that must be tuned during training and more features can lead to lower model accuracy if those features are not strongly correlated to the output. The model, as described, results in 12 total hyper-parameters to be trained and training is done via maximum likelihood estimation (MLE). The number of functional principal components (Methods section Functional Gaussian process, Eq. (7)) used is $J = 3$ and this captures >99% of the variability in the strain and the stress.

The overall results of the trained model on the withheld data can be seen in Fig. 4b, where the predicted mean nearly overlaps the mean of the CPFE data, and corresponding error rates are shown in Table 1. Overall, the fGP network is able to predict the strain to around 8% error and the stress within 5.3% error. These errors correspond to a prediction accuracy within 0.35% strain and approximately 20 MPa stress. The prediction of the stress will in general be better than the prediction of the strain for two reasons. First, there is non-linearity in the constitutive model, which inherently means that uncertainty will between stress and strain will be different. Second, since the stress fGP will have additional information via the mean predicted strain, which helps to further differentiate data points that would otherwise be similar. Note, however, that when considering the full predictive strain distribution as the input to the stress fGP, uncertainty propagation methods would need to be utilized and may result in a less accurate predicted mean stress with a higher predicted variance.

A selection of four grain stress–strain behaviors and the corresponding fGP mean prediction along with 95% prediction interval for those grains is shown in Fig. 4a. These results show mostly expected behavior in that if the grain stress–strain behavior is close to the mean behavior, the fGP can provide a good approximation, and as the grain behavior moves further away from the mean, the prediction intervals gets larger. In some

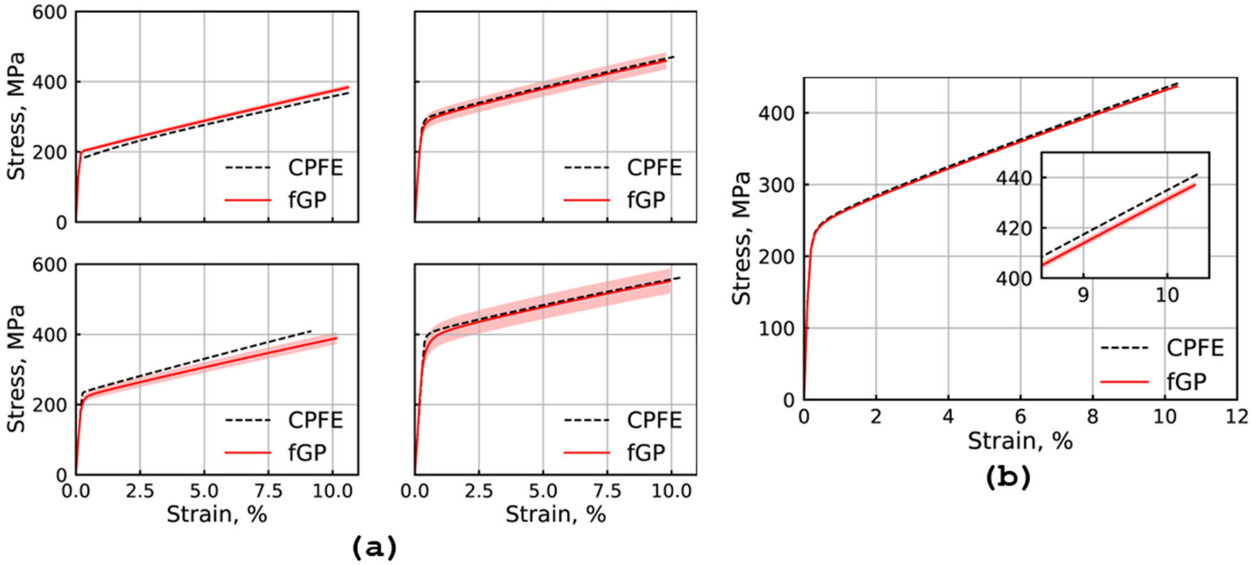


Fig. 4 fGP performance on withheld data. a fGP prediction with 95% prediction interval for four select grain data withheld from training. **b** fGP prediction with 95% prediction interval for the mean of all withheld data.

Table 1. Withheld data error metrics.

	MSE	MAE	MAPE	SMSE
Strain	3.282e−5	3.546e−3	7.834%	15.898
Stress	1326.6 MPa^2	19.971 MPa	5.3%	22.995

Error metrics for the trained fGP strain and stress models on the withheld grain data set. *MSE* mean squared error, *MAE* mean absolute error, *MAPE* mean absolute percent error, *SMSE* standardized mean squared error[68].

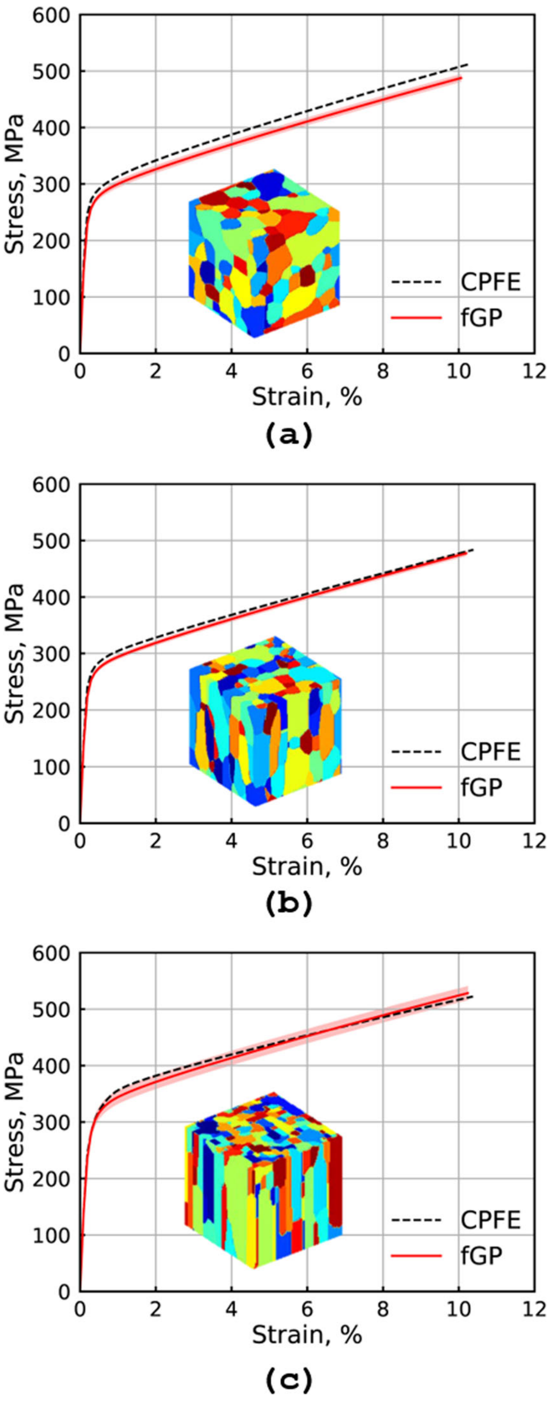
cases, such as the left two images of Fig. 4a, the predicted behavior has very narrow prediction intervals and the CPFE data does not lie within those intervals. However, the important behaviors (e.g., modulus, yield, and hardening) of individual grains tend to be captured very well by the fGP.

As mentioned above and as seen in Fig. 4a, the prediction of strain is more difficult than the prediction of stress. This can be attributed to a couple of training data deficiencies. First, in Fig. 3b, it can be noted that there is a high density of data around the mean, which will tend to bias the fGP toward that region. Next, it can be noted that, in general, the strain-displacement behavior is approximately linear for the majority of cases, but many grains can have significant non-linearity in this behavior. These grains tend to have the lowest prediction accuracy as they are outliers relative to the rest of the data. The reason for the presence of these non-linearities is likely due to boundary/traction conditions around the grain. This could be the result of a grain being on the boundary of the RVE or the result of a grain being constrained by its neighboring grains. The latter problem could be addressed potentially by accounting for physical grain boundary conditions in the fGP. For instance, one could create a metric that defines the surface area of the grain in contact with another grain. However, this would be quite challenging and may not result in a significant increase in model accuracy. To address the former problem, one would need to expand the training data set to include more cases that exhibit a nonlinear strain response, which could be accomplished via different load paths, and would decrease uncertainty in the model. However, this could prove challenging as well since the fGP suffers from the need to perform an $\mathcal{O}(n^3)$ matrix inversion during training. Therefore with more than a few thousand data, a significant training time penalty will be incurred. This is known in the standard GP regression problem and exacerbated in the fGP problem where training must occur on each of the J principal components. However, methods exist to circumvent this issue and these will be discussed later.

Recall that each RVE used during training contained approximately 100 grains. The fGP network is now applied to three RVEs of a more realistic size, containing upwards of 300 grains each, generated and simulated via the same process as before (i.e., generation via CDIM, meshing via ScanIP, and CPFE simulation). The generated RVEs in these cases are 0.125 mm^3. RVEs of this size generally take 60–90 h of computational time to generate the stress–strain response for a whole RVE on a high-performance computing system using 48 CPUs. This is in contrast to the 10–15 h that the 100 grain RVEs used for training take on the same computing system. Additionally, a larger RVE will result in the boundary conditions having less influence on the overall RVE behavior and, in general, yield a more representative behavior. As mentioned, the fGP does not explicitly take into account the boundary conditions meaning that the learned behavior may be significantly influenced by the boundary conditions. Before running the CPFE simulations, fGP predictions were made using RVE features. Once the CPFE model was run, the stress–strain results for the whole RVE were extracted as shown in Fig. 5 alongside the fGP predictions.

The corresponding error metrics are shown in Table 2 and in general show slightly higher error rates, as would be expected in ML on previously unseen data. However, the error rates are still of the same magnitude as those seen on the withheld training data and are sub-18% in the worst case, being sub-10% in the majority of cases, and in some cases are actually lower than the rates seen on the withheld training data. RVE 1 (Fig. 5a) shows the highest error rates of the three generated RVEs and can be attributed to a lack of sensitivity in the trained model. RVE 1 is nearly equiaxial and, as such, the distribution of grain sizes and shapes is relatively small. In the CPFE model, these small variations between grains are easily captured. However, in the fGP (as well as many other ML models), small local variations in input parameters are treated as similar to one another, even when they may not be, due to the characteristic length scale of the covariance being larger than the relative distance between some points.

Even considering the introduction of a small amount of error in the results, the reduction in computational cost (by three orders of magnitude) makes the fGP framework much more tractable than running the CPFE model. As mentioned, RVEs of a sufficient size, such as those in Fig. 5, can take between 60 and 80 h to simulate



(a)

(b)

(c)

Fig. 5 Stress–strain data for 300 grain RVEs. Stress–strain results for three 300 grain RVEs not used for the training of the fGP network. CPFE results took approximately 3 days of computation time on average while fGP predictions with 95% prediction intervals took seconds. **a** RVE 1. **b** RVE 2. **c** RVE 3.

on an HPC system. In contrast, the fGP network data generation, training, and prediction took between 500 and 700 h of time (50 RVEs at 10–15 h on the same HPC system, 8 h for training on a desktop, and prediction time was negligible on a desktop). While there is a significant time investment to construct the fGP network, the fGP network was trained in the same time as roughly ten CPFE simulations. However, the fGP network can provide a much more expansive data set than 10 CPFE simulations.

Table 2. Three hundred grain RVE error metrics.

	MSE	MAE	MAPE	SMSE
Total				
Strain	5.924e−5	4.954e−3	10.645%	7.422
Stress	6710.9 MPa^2	39.965 MPa	8.284%	19.094
RVE 1				
Strain	1.24e−4	8.064e−3	17.626%	19.028
Stress	5415.9 MPa^2	47.129 MPa	10.486%	41.846
RVE 2				
Strain	3.816e−5	4.124e−3	8.601%	2.548
Stress	3271.9 MPa^2	27.626 MPa	6.327%	8.982
RVE 3				
Strain	2.193e−5	3.004e−3	6.429%	1.706
Stress	10847.2 MPa^2	44.633 MPa	8.107%	8.375

Error metrics for the trained fGP strain and stress models on the three new 300 grain RVEs as well as the total error rates for the combined data set consisting of approximately 900 grains.

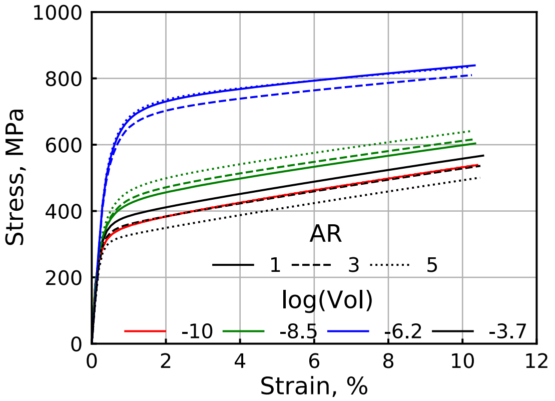


Fig. 6 Grain size and shape analysis. Stress–strain data from the fGP graphical network for the 200 grain “microstructures” generated via specifying a grain size and shape distribution. AR average aspect ratio, log(Vol) logarithm of the average grain volume.

To demonstrate this, the fGP will be used to examine how grain size and shape influence mechanical properties.

Grain size and shape effects

To demonstrate the uses of the fGP network for future problems that may require many mechanical property predictions for various microstructures or microstructure distributions (e.g., optimization or Bayesian sampling), a simple set of data with varied grain size and shape distributions will be created and mechanical properties predicted without the use of costly CPFE simulations. The data set consists of “microstructures” with average aspect ratios (shapes) of 1, 3, or 5 and average grain volumes (sizes) of 1.891e−5, 2.029e−4, 2.178e−3, 2.338e−2 mm^3. Distributions of grain sizes and shapes are generated via 200 random draws from a log-normal distribution with means as specified and standard deviations of 0.3 for the grain shape and 1 for the grain size. A full factorial analysis is performed to generate 12 representative microstructures with 200 grains each. Each of the 12 microstructure distributions is them simulated and homogenized via the fGP network to generate predicted mechanical behaviors as shown in Fig. 6. Note that 95%

prediction intervals are available for each curve but are omitted for figure clarity.

The generation and mechanical property prediction of this data (all 12 microstructures) took approximately 1 h on a standard desktop computer. The majority of this time was spent converting the input grain aspect ratio and volume into the necessary fGP inputs, which requires a numerical double integration. In contrast, if one were to attempt to simulate the same 12 RVEs with 200 grains using CPFE methods, each simulation would take upwards of a full day on an HPC system. The benefit of using the fGP network becomes even larger when considering that the time required has neglected the additional time needed to generate microstructural RVEs and mesh those RVEs before the CPFE model can be run. The drawback of the fGP is that there is a small amount of error introduced to be able to obtain these results so rapidly. However, this error will be small (<10% as demonstrated on the 300 grain RVEs) so long as the fGP is being used in an interpolative manner i.e., the microstructural features being input are in the range of those used to train the model. In this case, this condition holds since the fGP was trained with data that had higher and lower volumes and aspect ratios. If this condition is not met, then the accuracy of the fGP network will quickly diminish. This limitation of the fGP network is demonstrated on a microstructure generated by a cellular automata finite element (CAFE) model in Supplementary Fig. 2.

With regard to the results of the mechanical behavior, the mechanical properties show trends that are consistent with the theory. A brief overview of this consistency is given here but the interested reader is referred to refs. [56,59] for full information on the constitutive theory. First, it can be noted that at a very small average volume ($\log(\text{Vol}) = -10$) the aspect ratio has almost no influence on the mechanical behavior, since the grain is already "saturated". As average grain volume increase (regardless of aspect ratio), there is an initial increase in yield strength and a decrease in the hardening modulus, but at a high enough volume the yield strength decreases and the hardening modulus increases. This is indicative of the grain boundary effect being the dominant effect at intermediate volumes, but above a certain volume threshold, the grain boundary effect starts to diminish since the grain boundary volume is small relative to the overall volume. The effect of aspect ratio is confounded by the size effect, but generally shows that smaller grains with high aspect ratios produce a higher stress and this trend inverts with large grain sizes. The noted points are all consistent with the theory and the CPFE results presented in ref. [56], which gives further credence to the accuracy of the fGP model and its ability to emulate the CPFE model well.

This has been a simple demonstration of the fGP network and its potential time-saving capabilities, but this is a relatively small problem with only 12 microstructures that could be solved using CPFE in a longer but doable time frame. However, even in this simple context, one can begin to see how the fGP network can be used in PSP linkages to determine a desired microstructural feature distribution that results in a specific mechanical behavior. For instance, if the goal was to maximize yield strength then from Fig. 6, a target should be to achieve a microstructure with an average volume of 2.178e−3 mm^3 ($\log(\text{Vol}) = -6.2$) and either an equiaxial structure ($AR = 1$) or high aspect ratio structure ($AR = 5$). The true benefits of the fGP network are realized when considering an optimization problem or Bayesian sampling, where hundreds or thousands of microstructure feature distributions may be needed to find an optimal solution or the desired parameter distributions. With a CPFE model alone, this would be intractable, if not entirely impossible. With the fGP, this is not the case and the parameter space being explored can be thoroughly searched.

DISCUSSION

This work has demonstrated the development and application of an fGP-based graphical network. The fGP network emulates the simulation process for a CPFE model, where displacement and other input parameters are used to determine strain then stress in a grain. The fGP was trained using data from 50 RVEs generated using a CDIM and simulated using a grain size and shape-dependent CPFE constitutive model. Additionally, the amount of data was increased from the available RVEs by training the fGP network on a per grain basis rather than a per RVE basis. The fGP network was able to accurately predict new data from a test set and performed well on new RVEs generated by the same means as the training data, with differing numbers of grains and, therefore, different boundary conditions. This result demonstrated the possible capabilities of the fGP network to predict unseen data and the performance suggests that large CPFE models, which are computationally too expensive to simulate, could be approximated well by the fGP network trained using very small, more manageable RVEs with similar features. This is consistent with the theory of using small statistical volume elements (SVEs, volume elements which individually do not capture the average response of the material) to approximate a much larger domain[60]. However, the more traditional approaches require extracting SVEs from the microstructure of interest, simulating those SVEs, and then homogenizing to approximate the behavior of the domain of interest. The approach taken here is much more general in that it learns feature sets applicable to all microstructures exhibiting similar feature sets and does not require the repeated simulation of SVEs.

Having shown the fGP network is capable of predicting unseen data, it was applied to a simple problem of simulating 12 "microstructures" and proved to be consistent with the crystal plasticity constitutive theory. The fGP network was able to make predictions on data three orders of magnitude faster than the corresponding CPFE model (minutes on a single CPU compared to hours/days on an HPC system). Additional time savings are realized when considering that the effort required to mesh and fully define a CPFE model is not required to run the fGP network. Of course, the fGP network also has some drawbacks and limitations, which primarily stem from training data and data being predicted. Since the fGP is network is data driven ML model, it can only accurately make predictions on features with similarities to those it has seen during training (i.e., it is an interpolative model, not extrapolative). The less similar the unseen features are to the features seen during training, the worse the prediction will be. However, this is the case with all ML models.

The graphical network developed here provides a simple yet powerful data driven methodology to capture the structure–performance relationship in AM PSP linkages. Due to its relative simplicity, it is extraordinarily flexible in that it is not limited to the CPFE constitutive model used in this work or even limited to GP-based methods. While a specific CPFE constitutive model was chosen for this work, the fGP network can be implemented on any crystal plasticity data, such as that generated by spectral methods[61] or any other crystal plasticity constitutive model[62]. Changes to constitutive model would require modifying the input features so that they are specific to the given model, which then necessitates retraining of the network, but the core concept and framework is still applicable. Since the graphical network directly emulates the crystal plasticity method and predicts stress as well as strain, constitutive models containing damage and failure as well as complex load histories can be used to train the fGP network. Damage and defects can also be incorporated into the fGP network via the RVE by the inclusion of voids and/or cracks inside the microstructure[63,64]. The defects would be captured in the fGP network during the feature selection process. The drawback to this process is that the network training can no longer be done on a per grain basis and must be done on a whole RVE basis to capture defect distributions.

As mentioned, the framework developed here is not limited to GP-based methods, but GP models have the benefit of being well studied and easily modified. The extension of a standard GP to the fGP shown in section was straightforward and other modifications such as extensions to multiple outputs, incorporating multiple fidelity data, and utilizing sparse methods for "big data"[65,66] can be incorporated into the fGP. The extension of the fGP to incorporate multiple fidelity data could allow for both fast spectral methods and traditional slower non-spectral methods to be used simultaneously in the data generation and training processes. Sparse methods are the most immediately relevant extension to the problem at hand where training the fGP on thousands of grains using full rank methods becomes intractable, especially considering many of the grains from different microstructures have similar input–output pairs and contribute no new information to the fGPs.

Modifications to the input and outputs of the network could enhance the fGP network further. This work has focused on a simple loading procedure where a single component of displacement was specified, a monotonic linear loading parameter was used, and the equivalent stress and strain were output as scalar functions. The specification of the input to include multiple components of displacement would be trivial and the fGP network does not need modification to account for this. Likewise, the specification of a generalized loading parameter is possible, as described above, as long as the loading parameter is a real valued, unique, continuous function. However, by specifying a nonlinear, non-smooth, or non-monotonic parameter, difficulty in data generation (i.e., CPFE simulation convergence) could be encountered. Incorporation of multiple displacement components and a complex loading parameter would allow the network to capture behavior such as loading–unloading scenarios and non-proportional loading. While this work has focused on the scalar functional equivalent stress and strain outputs, the fGP network can easily be extended to include multiple functional outputs corresponding to the six components of stress/strain, either by training multiple independent networks or by modifying the fGP to utilize a multivariate normality assumption. With proper specification of the training data using combined loads (e.g., tension–torsion), the trained fGP network could directly emulate the anisotropic material stiffness tensor[67]. The implications of this are that Bayesian methods could be used to interrogate the trained fGP network to determine the approximate full stiffness tensor values (i.e. material properties).

As has been mentioned throughout this work, the fGP graphical network has a number of potential improvements, extensions, and applications. In addition to the improvements discussed already, one of the first improvements needed is to retrain the network using data with a richer feature set. In doing so, the fGP network will be able to predict the behavior of microstructures that are more representative of those seen in multi-layer, multi-track AM builds rather than simple single tracks. Next, using a Bayesian approach, the uncertainty in the fGP strain prediction can be propagated through the fGP stress model, potentially improving the predictive capability of the network. In the same vein, a Bayesian approach can be used to sample the fGP network and determine which features have the most impact on mechanical properties (as was shown in section) and which features are most likely to result in high stresses or strain energy densities, potentially leading to failure. The outcome of this process will link the structures that result in certain properties and, when combined with a data driven process–structure model, could result in real-time PSP linkages.

METHODS

Functional Gaussian process

A brief overview of GPs is first given before showing the extensions to a fGP. For a complete derivation of GPs, the interested reader is directed to the landmark work of Rasmussen and Williams[68]. First, let the input variables be denoted by $\boldsymbol{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)^{\mathrm{T}}$ and let $f(\cdot)$ be an unknown stochastic process. A GP is a non-parametric statistical model in which $f(\cdot)$ is to follow an n-dimensional multivariate Gaussian distribution such that,

$$p(f(\boldsymbol{x}_1), \ldots, f(\boldsymbol{x}_n)) \sim \mathcal{N}_n(\boldsymbol{\mu}, \boldsymbol{k}), \tag{1}$$

where $\boldsymbol{\mu}$ is the mean vector defined by the mean function $\mu(\boldsymbol{x}_i) = \boldsymbol{\mu}_i = \mathbb{E}[f(\boldsymbol{x}_i)]$ and $\boldsymbol{k}$ is the covariance defined by the covariance function $k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{k}_{ij} = cov[f(\boldsymbol{x}_i), f(\boldsymbol{x}_j)]$. Now, the GP can be denoted as $f(\cdot) \sim \mathcal{GP}(\mu(\cdot), k(\cdot, \cdot))$. The standard problem of nonlinear regression takes the form

$$y_i(\boldsymbol{x}_i) = f(\boldsymbol{x}_i) + \epsilon_i, \tag{2}$$

where f is as above and follows a GP, and ϵ_i are independent and identically distributed Gaussian random noise with 0 mean and σ^2 variance. It then follows that

$$\boldsymbol{y} = (y_1, \cdots, y_n)^{\mathrm{T}} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{K}), \tag{3}$$

where $\boldsymbol{\mu} = (\boldsymbol{\mu}_1, \cdots, \boldsymbol{\mu}_n)^{\mathrm{T}}$ and $\boldsymbol{K} = \boldsymbol{k}_{ij} + \sigma^2\boldsymbol{I}$, where $\boldsymbol{I}$ is the $n \times n$ identity matrix. The assumption of Normality is crucial to the GP framework as it allows the specification of a mean and covariance function that defines the presumed relationship between data points. As is common in many works, this work will assume that the mean function is $\mathbf{0}$. Furthermore, the covariance function will take the form of a Matérn 3/2 covariance as

$$k(\boldsymbol{x}, \boldsymbol{x}') = \eta\left(1 + \sqrt{3\sum_{k=1}^{p}\theta_k^2(x_k - x_k')^2}\right)\exp\left(-\sqrt{3\sum_{k=1}^{p}\theta_k^2(x_k - x_k')^2}\right). \tag{4}$$

The parameters $\{\eta, \theta_1, \ldots, \theta_p, \sigma^2\}$ make up the set of so-called hyperparameters, which allow "tuning" of the correlation between data points. The estimates of these parameters can be obtained through standard frequentist or Bayesian estimation. This work will utilize MLE throughout for simplicity. An additional outcome of the Normality assumption is that for a new input $\boldsymbol{x}^*$, the corresponding response is also Normally distributed and its mean and variance can be found as

$$\begin{aligned} y^* &= k(\boldsymbol{x}^*, \boldsymbol{X})^{\mathrm{T}}\boldsymbol{K}^{-1}\boldsymbol{y}, \\ \sigma^{2*} &= k(\boldsymbol{x}^*, \boldsymbol{x}^*) - k(\boldsymbol{x}^*, \boldsymbol{X})\boldsymbol{K}^{-1}k(\boldsymbol{X}, \boldsymbol{x}^*). \end{aligned} \tag{5}$$

In the derivation above, it has been assumed that $\boldsymbol{x}_i$ takes the form of a vector with scalar components. However, the data need not be of a scalar form and can take the form of functional data. With some modification to the above derivation, functional data can be incorporated into the GP. For the purposes of this work, the functional GP (fGP) will be restricted to functional outputs only. Extensions to scalar outputs are discussed following the derivation of the fGP.

The functional response $Y(t)$ can be defined as an L_2-continuous stochastic process on $\mathcal{T}$ such that the functional regression can be written as

$$Y_i(t) = f(\boldsymbol{X}_i(\cdot), \boldsymbol{z}_i) + \epsilon_i(t), \; t \in \mathcal{T}, \tag{6}$$

where $\boldsymbol{X}_i(\cdot)$ are now the q-dimensional functional parameters, $\boldsymbol{z}_i$ now represent p-dimensional scalar parameters, and the Gaussian noise is now functional as well with mean zero and variance σ_ϵ^2. Utilizing functional principal component analysis (fPCA), $Y_i(t)$ can be decomposed as

$$Y_i(t) = \mu(t) + \sum_{j=1}^{J}\beta_{ij}\phi_j(t) + \epsilon_i(t), \; t \in \mathcal{T}, \tag{7}$$

where $\mu(t)$ is the functional mean of the stochastic process, the summation term is the decomposition of the stochastic process covariance truncated to the first J terms, $\phi_j(t)$ are the stochastic process covariance eigenfunctions, and β_{ij} is the jth principal component of the ith sample. The functional mean and eigenfunctions of the previous equation do not depend on the sample i and as such can be determined using FDA methods without the use of a GP. Therefore, the problem of determining $Y_i(t)$ can be restated as two problems. First, determine the mean and eigenfunctions of the stochastic process as well as the noise variance using FDA. Second, relate the stochastic process principal components to the input parameters. The first process relies solely on the response data and acts as a linear shift and scaling of the data such that it has a zero mean. The second process can be stated as

$$\beta_{ij} = g_j(\boldsymbol{X}_i(\cdot), \boldsymbol{z}_i) + e_{ij}, \tag{8}$$

where $e_{ij} \sim \mathcal{N}(0, \sigma_j^2)$ with σ_j^2 being the Gaussian random noise variance of the jth principal component and g_j is an fGP for the jth principal component. Following the process above for a standard GP, it can now be stated that

$$\boldsymbol{\beta}_j = (\beta_{1j}, \cdots, \beta_{nj})^{\mathrm{T}} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{K}_j), \tag{9}$$

where, with slight change in notation from above, $\boldsymbol{K}_j = \boldsymbol{k}_{lmj} + \sigma_j^2 \boldsymbol{I}$. The covariance $\boldsymbol{k}_{lmj}$ will again take the form of a Matérn 3/2 as

$$\begin{aligned} k_j(\boldsymbol{X}, \boldsymbol{X}', \boldsymbol{z}, \boldsymbol{z}') = \eta_j \quad & \left(1 + \sqrt{3 \sum_{k=1}^{p} \theta_{kj}^2 (z_k - z_k')^2}\right) \left(1 + \sqrt{3 \sum_{k=1}^{q} \omega_{kj}^2 ||X_k - X_k'||_k^2}\right) \\ & \exp\left(-\sqrt{3 \sum_{k=1}^{p} \theta_{kj}^2 (z_k - z_k')^2} - \sqrt{3 \sum_{k=1}^{q} \omega_{kj}^2 ||X_k - X_k'||_k^2}\right). \end{aligned} \tag{10}$$

As before, the scalar term containing z_k is a standard Euclidean distance measure between data points that satisfies the properties of a metric space. However, the functional data term containing X_k does not satisfy the requirements for a metric space so traditional distance measures are not sufficient. A semi-metric space is a relaxed version of a metric space and measures of distance can be developed in that space as discussed by Ferraty and Vieu[39]. This work will utilize the fPCA based semi-metric which defines the distance between functional data as

$$||X - X'||_r^2 = \sum_{k=1}^{r} \left(\int [X(t) - X'(t)] v_k(t) \mathrm{d}t \right)^2, \tag{11}$$

where v_k are the orthonormal eigenfunctions of the largest r eigenvalue covariances, $\mathbb{E}[X(s)X(t)]$. Further discussion and practical implementation details are omitted here but can be found in ref. [39]. Having specified the covariance, the hyper-parameter set can be identified as $\{\eta_j, \theta_{1j}, \ldots, \theta_{pj}, \omega_{1j}, \ldots, \omega_{qj}, \sigma_j^2\}$ for every one of the J truncated principal components. In order to determine the predictive functional response, $Y^*(t)$, given a set of inputs ($\boldsymbol{X}^*(t)$, $\boldsymbol{z}^*$), the predictive mean, $\boldsymbol{\beta}_j^*$, and variance, σ_j^{2*}, must be found as

$$\begin{aligned} \boldsymbol{\beta}_j^* &= k_j(\boldsymbol{X}^*(t), \boldsymbol{X}(t), \boldsymbol{z}^*, \boldsymbol{z})^{\mathrm{T}} \boldsymbol{K}_j^{-1} \boldsymbol{\beta}_j, \\ \sigma_j^{2*} &= k_j(\boldsymbol{X}^*(t), \boldsymbol{X}^*(t), \boldsymbol{z}^*, \boldsymbol{z}^*) \\ & \quad - k_j(\boldsymbol{X}^*(t), \boldsymbol{X}(t), \boldsymbol{z}^*, \boldsymbol{z}) \boldsymbol{K}_j^{-1} k_j(\boldsymbol{X}(t), \boldsymbol{X}^*(t), \boldsymbol{z}, \boldsymbol{z}^*). \end{aligned} \tag{12}$$

Now, the predictive mean and variance of the functional response can be found as

$$\begin{aligned} Y^*(t) &= \hat{\mu}(t) + \sum_{j=1}^{J} \boldsymbol{\beta}_j^* \phi_j(t), \\ \sigma^{2*}(t) &= \hat{\sigma}_\mu^2(t) + \sum_{j=1}^{J} \sigma_j^{2*} \phi_j^2(t) + \hat{\sigma}_\epsilon^2, \end{aligned} \tag{13}$$

where $\sigma_\mu^2(t)$ is the variance of the functional mean $\mu(t)$ and the $\hat{(\cdot)}$ notation has been introduced to denote values estimated using FDA methods. As an aside, the fGP utilized here has the capability to model any combination of functions/scalars to functions/scalars. The derivation above has shown function-on-function/scalar regression but one could reduce this to a case of function-on-scalar or function by simply eliminating, respectively, the first or second summation term in Eq. (10). Additionally, for scalar-on-function/scalar regression, the kernel of the standard GP (Eq. (4)) can simply be replaced by the functional kernel (Eq. (10)). The fGP is implemented in a Python class, while the FDA methods used to determine the functional mean and variance as well as the functional Gaussian noise variance are implemented in Matlab.

Data generation

The crystal plasticity data in this work are based on a microstructural-informed CPFE model developed by Saunders et al.[56]. The microstructure-informed CPFE model is a phenomenological model, which has been modified to account for grain size and aspect ratio effects. This modification was directed specifically at capturing the non-conventional grain morphologies seen in AM parts. A brief description of the process used to generate, mesh, and simulate microstructures is given here but, for brevity, theoretical aspects of the constitutive model and full implementation details are omitted and the interested reader is referred to ref. [56].

For the purpose of computational efficiency, this work utilizes a synthetic microstructure generation method known as the CDIM[57]. The CDIM is capable of generating, in a matter of minutes, RVEs with features mimicking those seen in actual AM microstructures that can be used for crystal plasticity simulations. In contrast, alternative methods for generating AM-like microstructures, such as cellular automata (CA) or phase field method, can take on the order of hours or days even on high-performance computing systems. Once generated, the RVEs are automatically imported into Simpleware ScanIP (Synopsys, Mountain View, CA, USA) using a Python script, run through the Simpleware Scripting interface. The script then creates a single multi-label mask based on the greyscale information contained in the image data. Finally an unstructured volume mesh is generated and exported. The mesh comprises curved quadratic tetrahedral elements generated Simpleware's +FE Free algorithm. The density of elements is dependent on the geometry of the structure, with more smaller tetrahedra added where the surface requires greater representation. In regions of small or no geometric change, decimation is used to generate larger tetrahedral elements to reduce the size of the mesh files and so speed up simulation. The mesh settings chosen were optimized for the simulation hardware resources available. The simulation-ready meshes were exported in a *.inp format. Once meshed and exported, the RVE is processed again to incorporate periodic boundary conditions, assign constitutive model parameters, and apply the desired loading scenario. The RVE is simulated using Abaqus/Standard (Dassault Systems, Providence, RI, USA) and the constitutive model is implemented in a user material subroutine (UMAT).

In this work, the microstructure morphology variations are the only input parameters being varied. Thus to generate the necessary data to train an fGP model, the process described above must be run iteratively to generate microstructures exhibiting a variety of different size and aspect ratio grains. The primary inputs to the CDIM are three parameters which describe the aspect ratio of the grains being generated. Note that RVE size is not specified in the CDIM model and is introduced later by linearly scaling the RVE edge lengths from 0.1 to 0.8 mm. Since the RVE is a cube, this results in four parameters being varied. A simple Latin hypercube sampling strategy is implemented to generate a grain size-and-aspect ratio-space filling design.

DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author R.S., upon reasonable request, and subject to US Department of Defense security and export control restrictions.

CODE AVAILABILITY

The code that support the findings of this study are available from the corresponding author R.S., upon reasonable request, and subject to US Department of Defense security and export control restrictions.

Received: 17 December 2020; Accepted: 1 May 2021;
Published online: 03 June 2021

REFERENCES

1. Panchal, J. H., Kalidindi, S. R. & McDowell, D. L. Key computational modeling issues in integrated computational materials engineering. *Comput. Aided Des.* **45**, 4–25 (2013).
2. Lim, H. et al. Incorporating physically-based microstructures in materials modeling: bridging phase field and crystal plasticity frameworks. *Model. Simul. Mater. Sci. Eng.* **24**, 045016 (2016).
3. Diehl, M. et al. Identifying structure-property relationships through DREAM.3D representative volume elements and DAMASK crystal plasticity simulations: an integrated computational materials engineering approach. *JOM* **69**, 848–855 (2017).
4. Kalidindi, S. R. & De Graef, M. Materials data science: current status and future outlook. *Annu. Rev. Mater. Res.* **45**, 171–193 (2015).
5. Agrawal, A. & Choudhary, A. Perspective: Materials informatics and big data: realization of the "fourth paradigm" of science in materials science. *APL Mater.* **4**, 053208 (2016).
6. McDowell, D. L. & LeSar, R. A. The need for microstructure informatics in process-structure-property relations. *MRS Bull.* **41**, 587–593 (2016).

7. Kalidindi, S. R., Medford, A. J. & McDowell, D. L. Vision for data and informatics in the future materials innovation ecosystem. *JOM* **68**, 2126–2137 (2016).
8. Ramprasad, R., Batra, R., Pilania, G., Mannodi-Kanakkithodi, A. & Kim, C. Machine learning in materials informatics: recent applications and prospects. *NPJ Comput. Mater.* **3**, 54 (2017).
9. Dimiduk, D. M., Holm, E. A. & Niezgoda, S. R. Perspectives on the impact of machine learning, deep learning, and artificial intelligence on materials, processes, and structures engineering. *Integr. Mater. Manuf. Innov.* **7**, 157–172 (2018).
10. Lookman, T., Balachandran, P. V., Xue, D. & Yuan, R. Active learning in materials science with emphasis on adaptive sampling using uncertainties for targeted design. *NPJ Comput. Mater.* **5**, 21 (2019).
11. Brough, D. B., Wheeler, D., Warren, J. A. & Kalidindi, S. R. Microstructure-based knowledge systems for capturing process-structure evolution linkages. *Curr. Opin. Solid State Mater. Sci.* **21**, 129–140 (2017).
12. Paulson, N. H., Priddy, M. W., McDowell, D. L. & Kalidindi, S. R. Reduced-order structure-property linkages for polycrystalline microstructures based on 2-point statistics. *Acta Mater.* **129**, 428–438 (2017).
13. Yang, Z. et al. Deep learning approaches for mining structure-property linkages in high contrast composites from simulation datasets. *Comput. Mater. Sci.* **151**, 278–287 (2018).
14. Khosravani, A., Cecen, A. & Kalidindi, S. R. Development of high throughput assays for establishing process-structure-property linkages in multiphase polycrystalline metals: application to dual-phase steels. *Acta Mater.* **123**, 55–69 (2017).
15. DebRoy, T. et al. Additive manufacturing of metallic components—process, structure and properties. *Prog. Mater. Sci.* **92**, 112–224 (2018).
16. Herzog, D., Seyda, V., Wycisk, E. & Emmelmann, C. Additive manufacturing of metals. *Acta Mater.* **117**, 371–392 (2016).
17. Francois, M. et al. Modeling of additive manufacturing processes for metals: challenges and opportunities. *Curr. Opin. Solid State Mater. Sci.* **21**, 198–206 (2017).
18. Gao, W. et al. The status, challenges, and future of additive manufacturing in engineering. *Comput. Aided Des.* **69**, 65–89 (2015).
19. Wang, Y. M. et al. Additively manufactured hierarchical stainless steels with high strength and ductility. *Nat. Mater.* **17**, 63–71 (2018).
20. Smith, J. et al. Linking process, structure, property, and performance for metal-based additive manufacturing: computational approaches with experimental support. *Comput. Mech.* **57**, 583–610 (2016).
21. Markl, M. & Körner, C. Multiscale modeling of powder bed-based additive manufacturing. *Annu. Rev. Mater. Res.* **46**, 93–123 (2016).
22. Mukherjee, T. & DebRoy, T. A digital twin for rapid qualification of 3D printed metallic components. *Appl. Mater. Today* **14**, 59–65 (2019).
23. Yan, W. et al. An integrated process-structure-property modeling framework for additive manufacturing. *Comput. Methods Appl. Mech. Eng.* **339**, 184–204 (2018).
24. Yan, W. et al. Data-driven multi-scale multi-physics models to derive process-structure-property relationships for additive manufacturing. *Comput. Mech.* **61**, 521–541 (2018).
25. Roters, F. et al. Overview of constitutive laws, kinematics, homogenization and multiscale methods in crystal plasticity finite-element modeling: theory, experiments, applications. *Acta Mater.* **58**, 1152–1211 (2010).
26. Segurado, J., Lebensohn, R. A. & LLorca, J. Computational Homogenization of Polycrystals. In *Advances in Applied Mechanics: Volume 51* (ed Hussein, M. I.) 1–114 (Elsevier, 2018).
27. Jones, R. E., Templeton, J. A., Sanders, C. M. & Ostien, J. T. Machine learning models of plastic flow based on representation theory. *Comput. Model. Eng. Sci.* **117**, 309–342 (2018).
28. Wang, K., Sun, W. & Du, Q. A cooperative game for automated learning of elasto-plasticity knowledge graphs and models with AI-guided experimentation. *Comput. Mech.* **64**, 467–499 (2019).
29. Wang, K. & Sun, W. Meta-modeling game for deriving theory-consistent, microstructure-based traction-separation laws via deep reinforcement learning. *Comput. Methods Appl. Mech. Eng.* **346**, 216–241 (2019).
30. Jung, J., Yoon, J. I., Park, H. K., Kim, J. Y. & Kim, H. S. An efficient machine learning approach to establish structure-property linkages. *Comput. Mater. Sci.* **156**, 17–25 (2019).
31. Jung, J., Yoon, J. I., Park, H. K., Kim, J. Y. & Kim, H. S. Bayesian approach in predicting mechanical properties of materials: Application to dual phase steels. *Mater. Sci. Eng. A Struct. Mater.* **743**, 382–390 (2019).
32. Reimann, D. et al. Modeling macroscopic material behavior with machine learning algorithms trained by micromechanical simulations. *Front. Mater.* **6**, 3513–3520 (2019).
33. Whelan, G. & McDowell, D. L. Machine learning-enabled uncertainty quantification for modeling structure-property linkages for fatigue critical engineering alloys using an ICME workflow. *Integr. Mater. Manuf. Innov.* **9**, 376–393 (2020).
34. Liu, Z. & Wu, C. Exploring the 3D architectures of deep material network in data-driven multiscale mechanics. *J. Mech. Phys. Solids* **127**, 20–46 (2019).
35. Frankel, A., Tachida, K. & Jones, R. Prediction of the evolution of the stress field of polycrystals undergoing elastic-plastic deformation with a hybrid neural network model. *Mach. Learn. Sci. Technol.* **1**, 035005 (2020).
36. Frankel, A., Jones, R., Alleman, C. & Templeton, J. Predicting the mechanical response of oligocrystals with deep learning. *Comput. Mater. Sci.* **169**, 109099 (2019).
37. Morris, J. S. Functional regression. *Annu. Rev. Stat. Appl.* **2**, 321–359 (2015).
38. Ramsay, J. O. & Silverman, B. W. *Functional Data Analysis*. Springer Series in Statistics, 2nd edn (Springer New York, 2005).
39. Ferraty, F. & Vieu, P. *Nonparametric Functional Data Analysis*. Springer Series in Statistics (Springer New York, 2006).
40. Ling, N. & Vieu, P. Nonparametric modelling for functional data: selected survey and tracks for future. *Statistics* **52**, 934–949 (2018).
41. Morris, M. D. Gaussian surrogates for computer models with time-varying inputs and outputs. *Technometrics* **54**, 42–50 (2012).
42. Tang, X., Hong, Z., Hu, Y. & Lian, H. Gaussian process models for non parametric functional regression with functional responses. *Commun. Stat. Theory Methods* **44**, 3428–3445 (2015).
43. Poynor, V. & Munch, S. Combining functional data with hierarchical Gaussian process models. *Environ. Ecol. Stat.* **24**, 175–199 (2017).
44. Yang, J., Cox, D. D., Lee, J. S., Ren, P. & Choi, T. Efficient Bayesian hierarchical functional data analysis with basis function approximations using Gaussian-Wishart processes. *Biometrics* **73**, 1082–1091 (2017).
45. Luo, L., Yao, Y., Gao, F. & Zhao, C. Mixed-effects Gaussian process modeling approach with application in injection molding processes. *J. Process Control* **62**, 37–43 (2018).
46. Betancourt, J. et al. Gaussian process metamodeling of functional-input code for coastal flood hazard assessment. *Reliab. Eng. Syst. Saf.* **198**, 106870 (2020).
47. Li, J., Jin, R. & Yu, H. Z. Integration of physically-based and data-driven approaches for thermal field prediction in additive manufacturing. *Mater. Des.* **139**, 473–485 (2018).
48. Wang, B. & Xu, A. Gaussian process methods for nonparametric functional regression with mixed predictors. *Comput. Stat. Data. Anal.* **131**, 80–90 (2019).
49. Conti, S. & O'Hagan, A. Bayesian emulation of complex multi-output and dynamic computer models. *J. Stat. Plan. Inference* **140**, 640–651 (2010).
50. Kennedy, M. Predicting the output from a complex computer code when fast approximations are available. *Biometrika* **87**, 1–13 (2000).
51. Forrester, A. I., Sóbester, A. & Keane, A. J. Multi-fidelity optimization via surrogate modelling. *Proc. R. Soc. Lond. A Math. Phys. Sci.* **463**, 3251–3269 (2007).
52. Le Gratiet, L. & Cannamela, C. Cokriging-based sequential design strategies using fast cross-validation techniques for multi-fidelity computer codes. *Technometrics* **57**, 418–427 (2015).
53. Mangal, A. & Holm, E. A. Applied machine learning to predict stress hotspots I: face centered cubic materials. *Int. J. Plast.* **111**, 122–134 (2018).
54. Bostanabad, R. et al. Computational microstructure characterization and reconstruction: review of the state-of-the-art techniques. *Prog. Mater. Sci.* **95**, 1–41 (2018).
55. Rodgers, T. M., Lim, H. & Brown, J. A. Three-dimensional additively manufactured microstructures and their mechanical properties. *JOM* **72**, 75–82 (2020).
56. Saunders, R. N., Achuthan, A., Iliopoulos, A. P., Michopoulos, J. G. & Bagchi, A. *Effects of the microstructural Grain Size and Aspect Ratio on the Mechanical Properties of Additively Manufactured Parts Via Computational Analysis* (Defense Technical Information Center, 2020).
57. Michopoulos, J. G. et al. Towards computational synthesis of microstructural crystalline morphologies for additive manufacturing applications. In *37th Computers and Information in Engineering Conference*, Vol. 1, V001T02A030–V001T02A030 (American Society of Mechanical Engineers, 2017).
58. Birnbaum, A. J., Steuben, J. C., Barrick, E. J., Iliopoulos, A. P. & Michopoulos, J. G. Intrinsic strain aging, Σ3 boundaries, and origins of cellular substructure in additively manufactured 316L. *Addit. Manuf.* **29**, 100784 (2019).
59. Achuthan, A., Iliopoulos, A. P., Michopoulos, J. G., Saunders, R. & Bagchi, A. Towards a constitutive model that encapsulates microstructural features induced by powder additive manufacturing. In *37th Computers and Information in Engineering Conference*, Vol. 1, 1–11 (American Society of Mechanical Engineers, 2017).
60. Qidwai, S. M. et al. Estimating the response of polycrystalline materials using sets of weighted statistical volume elements. *Acta Mater.* **60**, 5284–5299 (2012).
61. Knezevic, M., Al-Harbi, H. F. & Kalidindi, S. R. Crystal plasticity simulations using discrete Fourier transforms. *Acta Mater.* **57**, 1777–1784 (2009).
62. Roters, F. et al. DAMASK—The Düsseldorf Advanced Material Simulation Kit for modeling multi-physics crystal plasticity, thermal, and damage phenomena from the single crystal up to the component scale. *Comput. Mater. Sci.* **158**, 420–478 (2019).
63. Taheri Andani, M., Karamooz-Ravari, M. R., Mirzaeifar, R. & Ni, J. Micromechanics modeling of metallic alloys 3D printed by selective laser melting. *Mater. Des.* **137**, 204–213 (2018).

64. Taheri Andani, M., Ghodrati, M., Karamooz-Ravari, M. R., Mirzaeifar, R. & Ni, J. Damage modeling of metallic alloys made by additive manufacturing. *Mater. Sci. Eng. A Struct. Mater.* **743**, 656–664 (2019).
65. Titsias, M. Variational learning of inducing variables in sparse Gaussian processes. In *Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics*, Vol. 5 of *Proceedings of Machine Learning Research* (eds. van Dyk, D. & Welling, M) 567–574 (PMLR, 2009).
66. Hensman, J., Fusi, N. & Lawrence, N. D. Gaussian processes for big data. In *Proceedings of the Twenty-Ninth Conference on Uncertainty in Artificial Intelligence*, 282–290 (2013).
67. Steuben, J., Michopoulos, J., Iliopoulos, A. & Turner, C. Inverse characterization of composite materials via surrogate modeling. *Compos. Struct.* **132**, 694–708 (2015).
68. Rasmussen, C. E. & Williams, C. K. I. Regression. In *Gaussian Processes for Machine Learning* (The MIT Press, 2005).

ACKNOWLEDGEMENTS

R.S., J.M., and A.B. acknowledge partial support for this work by the Office of Naval Research through the Naval Research Laboratory's (NRL) core funding. R.S. acknowledges partial support for this work by the NRL Edison Graduate Memorial Training Program. This work was supported in part by high-performance computer time and resources from the DoD High Performance Computing Modernization Program. R.S. acknowledges Dr. Bo Wang, University of Leicester, for sharing implementation details for the functional Gaussian process, and he also acknowledges Dr. Kirubel Teferra, NRL, for the generation of the CA data used in this work and Dr. Athanasios Iliopoulos, NRL, for sharing the CDIM code.

AUTHOR CONTRIBUTIONS

C.B. performed the segmentation and meshing of microstructures. R.S. performed the remainder of the technical work and writing of the manuscript under supervision of J.M., D.L., A.E., and A.B. All authors contributed to review and editing of the final manuscript.

COMPETING INTERESTS

The authors declare no competing interests.

ADDITIONAL INFORMATION

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s41524-021-00548-y.

Correspondence and requests for materials should be addressed to R.S.

Reprints and permission information is available at http://www.nature.com/reprints

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this license, visit http://creativecommons.org/licenses/by/4.0/.

This is a U.S. Government work and not under copyright protection in the US; foreign copyright protection may apply 2021