MTECH PROJECT PROGRESS REPORT

Machine Learning enabled surrogate Crystal Plasticity Model for prediction of intergranular crack initiation

by

Angaddeep Singh

(2017MED1003)

Under the supervision of

Dr. Dhiraj K. Mahajan & Dr. Manish Agrawal

Department of Mechanical Engineering

Indian Institute of Technology Ropar

September 2021

Certificate

I, hereby submit my MTech Project progress report, detailing the work done by me in this semester. I certify that this is my original work, and the material referred from other sources (books, manuals, journal, conference proceedings, etc.) have been duly acknowledged.

Date: 30-9-21

Angaddeep Singh
(2017MED1003)

The student has worked under my/our supervision for the above mentioned work. I/We have read this progress report; it meets my/our expectations and accurately reflects the work done by the student.

Date: 30-9-21

Dr. Dhiraj K. Mahajan & Dr. Manish Agrawal

Acknowledgements

I would like to express my sincere gratitude to my advisors Dr. Dhiraj K. Mahajan and Dr. Manish Agrawal for their valuable guidance, patience, motivation and immense knowledge.

Abstract

Fatigue is a complicated process involving inelastic deformation and formation of cracks. The Crack initiation process entails about eighty percent of the useful lifetime of a material and still experimental prediction of crack initiation remains a tedious task. Granular microstructural features play a crucial role in determining the locations of crack initiation, hence it becomes quite important to understand the different mesoscale structural properties that can help us to design materials in a way to delay crack initiations. Several crystal plasticity models have been trying to understand the different mechanisms involved in fatigue deformation, but the issue with them is the extensive requirement of computational resources.

Recent developments in machine learning have shown promising results in this direction. Several machine learning models have been built to correlate microstructural features with mechanical properties like tensile strength and toughness. However, no major research has been done to use machine learning for predicting crack initiation under fatigue cycles.

Since the first step in any machine learning project involves data preparation, the workflow for data preparation through Crystal Plasticity-FEM is also discussed briefly.

Table of Contents

List of Figures

Chapter 1

Introduction

1.1. Motivation

Fatigue is a complicated process involving inelastic deformation and formation of cracks. Microstructural features exert a great degree of influence on the locations of crack initiation. Around eighty percent of the useful lifetime of a material goes into the crack initiation process, however determination of cracks through experimentation remains a very difficult process.Moreover, the analysis of experimental data is a very complex process because of the sheer number of interactions possible between different grains. Several crystal plasticity models have been trying to understand the different mechanisms involved in fatigue deformation, but the issue with them is the extensive requirement of computational resources. Although Crystal Plasticity models have been used in a wide range of applications, when it comes to simulating a database of thousands of structures, it becomes difficult to depend on it. Recent developments in machine learning have been trying to solve this problem by developing surrogate models that can fast track the simulation time.

Machine learning approaches seem to be like a promising alternative because of their ability to model complex features. Thus far, there has been various research in machine learning to link microstructural features with mechanical properties like ultimate tensile strength and toughness. The aim of the project will be to understand the factors which influence the formation of crack initiation under fatigue loading.

The understanding of these factors would help us to design materials much quicker and with less requirement of resources when compared to experimentation and Crystal plasticity-FEM.

1.2. Background

Two ways to use machine learning to design new materials:

1 Supervised Learning: It basically tries to find the unknown function that links

known inputs to unknown outputs. Wherein known inputs will be microstructural features and unknown outputs will be mechanical properties. After the model is trained, mechanical properties of a material can be predicted for given microstructural features. The model would be less computationally expensive than traditional numerical methods like FEA, thus enabling us to design material much faster.

2 Unsupervised Learning: It's motive is finding patterns within unlabelled data. In many problems we don't have any curated database, in such cases unsupervised learning can help us to group the data, reduce the noise in data or create augmented versions of data to increase the size of our database. One of the challenges while designing new materials in conventional approaches is that the material can only be found from a limited number of structures in the available database. Hence, if we can increase the size of the candidate material database, our probability of finding the desired material would increase. This is where deep learning approaches come to the rescue, deep generative models like Variational autoencoders and GANs can help us to construct a database by creating new microstructures images by slightly augmenting the current database.

For any machine learning project, data preparation remains one of the most crucial steps.

Generating data through experiments becomes quite hard in case of machine learning projects, since the number of data points to be produced are quite high. Hence, generating synthetic microstructures and simulating them through FEM makes our lives much easier. Open source software DREAM.3D allows us to generate synthetic microstructures which can be simulated in FEM softwares like ABAQUS. An open source CPFEM software: PRISMS-Plasticity which allows better parallel processing than abaqus is also discussed.

Chapter 2

Literature Review

This project requires an understanding of 2 domains: 1. Different machine learning algorithms that can be used for correlating granular microstructural features with mechanical properties; 1. Mechanisms and factors involved in crack initiation under fatigue loading .

Significant progress has been made in the direction of using machine learning models to correlate microstructural features with mechanical properties.

Generative models, CNNs, RNNs, gaussian processes have been demonstrated in Kim and Park (2021), Pandey and Pokharel (2020), Yang and Li (2017), Heriot and Spear (2020) to correlate microstructures with properties like Ultimate tensile strength.

Mangal and Holm (2017) demonstrated the use of random forest learning algorithm for predicting the locations of stress hotspots under uniaxial tension by taking use of grain orientation and neighborhood information, which is very similar to our aim of determining fatigue crack initiation.

Experimental research has also been reviewed from Aman Arora and Mahajan (2020), wherein it was demonstrated that we can take use of microstructural parameters to predict fatigue crack initiation.

An open source framework PRISMS-Fatigue has been discussed which will be used in the data preparation for machine learning model.

2.1. Exploration of optimal microstructure and mechanical properties in continuous microstructure space using a variational autoencoder, Kim and Park (2021)

This paper tried to use a combination of Variational autoencoder and Gaussian process regression which would enable us to get the optimal microstructures given some target mechanical properties.

The steps involved in this paper were:

1 Synthetic microstructure generation: DP steel microstructures were generated using software DREAM.3D. Information about grain size, orientation angles, morphology was provided to the software as inputs. Material characterization was performed using SEM and ESBD was fed in DREAM.3D.

2 Variational Autoencoder: An autoencoder is a type of deep neural network which consists of two parts:

Encoder: Converts the given data into an encoding, ignoring the noise and performing dimensionality reduction

Decoder: Converts the encoding back into reconstructed form.

Autoencoder is capable of handling much very complicated sets of features when compared to traditional dimensionality reduction methods such as PCA

Variational autoencoder goes one step further, by providing a continuous latent space which describes the data in a probabilistic manner, encoding the data in the form of a normal distribution.

Thus, using VAE we get a continuous microstructure space, which can be used to increase our candidate database.

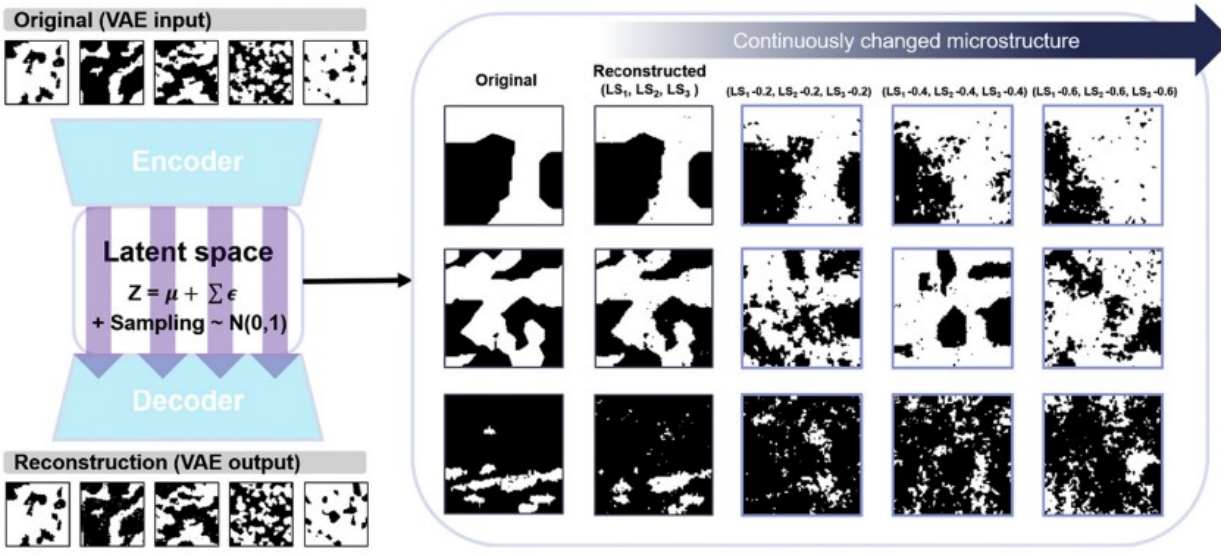


Fig 1. Schematic demonstration of continuous latent space created by VAE [1]

3 Finite Element Modelling: FEM was employed to get the values of mechanical properties such as UTS and toughness. 3D FEM simulation was performed

using ABAQUS, applying uniaxial tension on structures made of C3D8 elements.

4 Gaussian Process regression: GPR was trained to be used as a surrogate of FEM, linking microstructural features to mechanical properties using minimum number of simulations on Abaqus.

Input Variables(X): Latent Space generated by encoder, grain size of ferrite

Output Variables(Y): UTS, toughness by FEM

How GPR minimizes the number of trials is also very interesting.

There are two methods for implementing supervised learning:

- Restricting the possible functions that we consider. In this method, before training we consider our output to be a sum of basis functions multiplied by some weights. During training, we try to maximize the likelihood that the given training output points pass through our assumed function by changing the weights. It's quite important to consider the right order of the basis functions we assume, because if we take a very complex or simple function, it may lead to overfitting or underfitting respectively. For predictions, we only get the most probable value, but we can't know the uncertainty involved.
- Bayesian Modelling: Bayesian thinking says why don't we directly map the output points and consider all the possible sets of functions, instead of restricting ourselves to some basis functions and then finding the weights. So, in Bayesian modelling we give every possible function some prior probability, and the assigning of probabilities is decided by some heuristics like how smooth the output function will be.

  But considering every possible set of functions can be difficult as there could be infinite functions, this is where gaussian processes come into play. A gaussian process helps us to generalize every possible function in the form of a normal distribution.

  There are two advantages of this approach: First, we don't have to restrict ourselves to some functions, hence overfitting/underfitting doesn't happen. Secondly for every prediction, we get both the mean

and uncertainty associated with it. The uncertainty value can be useful to us in a whole range of applications, especially in cases where the cost of conducting the actual experiment is quite expensive.

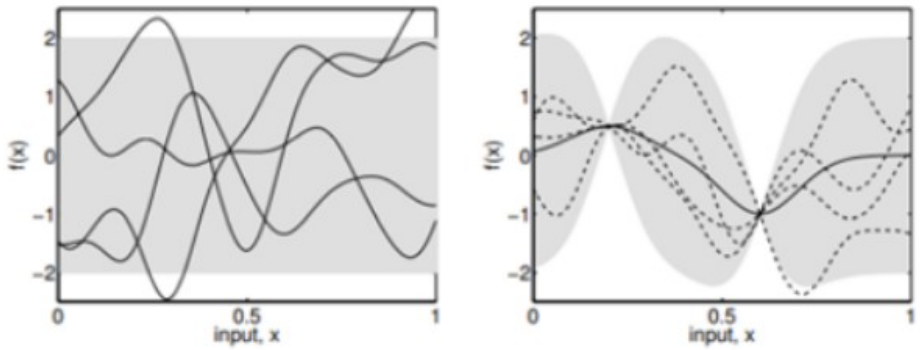


Fig 2. Uncertainty of points goes down after being included in training data[1]

Intuitively how GPR works is, it initially considers all sets of possible functions. Then it considers training points one by one, after each training point it sets the value of uncertainty at that point to be zero (since we are fully certain of its value), we can also observe from the above image that the uncertainty of the neighborhood points also goes down. The assumption behind reducing uncertainty of neighborhood points is that the required function is continuous. We can infer from this that the biggest reduction in uncertainty happens when we get a data point in whose vicinity the uncertainty is high. Thus the best strategy for reducing the number of trials would be to get only those data points where uncertainty is high in neighborhood points. So by using GPR, we can smartly reduce the number of FEM simulations by taking use of the uncertainty values.

Using VAE-GPR combination, they were able to construct a continuous latent space which can be employed to find the microstructures of required mechanical properties. The advantage of using VAE over other dimensionality reduction methods is that we can obtain reconstructed images using decoder.

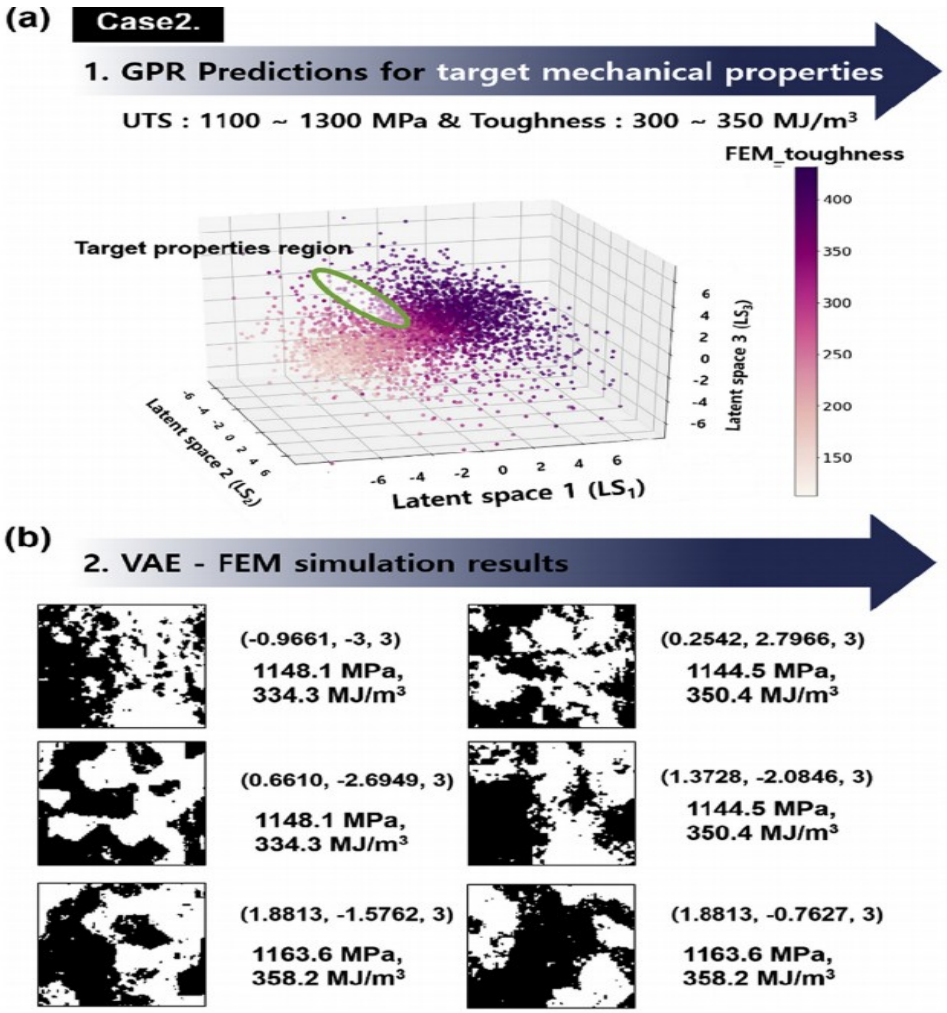


Fig 3. (a) GPR predictions according to target properties (b) Reconstructed images simulated in ABAQUS for verification[1]

Using GPR we can sample microstructures with target mechanical properties from the latent space. The encoding from the latent space can then be reconstructed using the decoder, in the above image the reconstructed images were simulated using FEM to verify the model. We can observe that the FEM results lie within the target mechanical properties.

2.2. Applied Machine Learning to Predict Stress Hotspots I: Face Centered Cubic Materials, Mangal and Holm (2017)

Microscopic voids formed during nucleation is one of the most common reasons for fracture in materials. These microscopic voids are generally the location of stress hotspots where crack initiation occurs.

Microstructural features determine the location of these voids formed during nucleation. Hence, an understanding of the correlation between microstructural features and location of stress hotspots can help us to design materials in a way to maximize the useful lifetime of a material.

In this research paper, a supervised machine learning model of decision trees is employed to establish the correlation between microstructural features and locations of stress hotspots. The advantages of decision trees is that the heuristics employed in the model can be understood by humans.

The dataset was prepared using DREAM.3D and simulated using the crystal plasticity EVPFFT model with uniaxial tensile deformation. Six different types of textures were considered, and six instantiations of each texture were generated. Hence, 36 microstructures were simulated in total.

Two sets of input features were prepared: 1. Crystallographic descriptors: Texture and euler angles dependent features 2. Geometrical descriptors: Geometry and size of grains

Output was classified as yes or no depending on the Von mises stress at different grains, locations with Von mises stress above 90th percentile were defined as stress hotspots.

Random Forest algorithm was implemented in 2 methods:

1 Partition Model: Different random forest models are created for different

textures.

2 Mixed Model: The data from all types of textures are fed into a single random forest model.

AUC score is one of the most useful metric to judge the performance of a classification model.

This metric is calculated by computing the area under the ROC curve. ROC is plotted using two parameters: 1. False Positive rate (x-axis) 2. True positive rate (y-axis).

The AUC score ranges between 0 and 1. Better score means the model is performing good.

Table 1: Cross validation AUCs for FCC materials for mixed and partition models

Texture kind	**Partition Model AUC (%)**	**Mixed Model AUC (%)**
1	71.27 ± 1.25	72.07
2	73.91 ± 3.69	73.10
3	66.13 ± 2.74	67.83
4	73.61 ± 6.90	79.84
5	74.01 ± 3.25	76.31
6	72.65 ± 3.07	75.01
All	71.93 ± 3.88	74.03 ± 3.72

Fig 4.AUC score of the trained random forest model[2]

We can infer from the above table that the Mixed model performed better than the partition model. AUC score of 74.03% is a pretty good score, considering that the prediction from the machine learning model will be thousands of times faster than FEM.

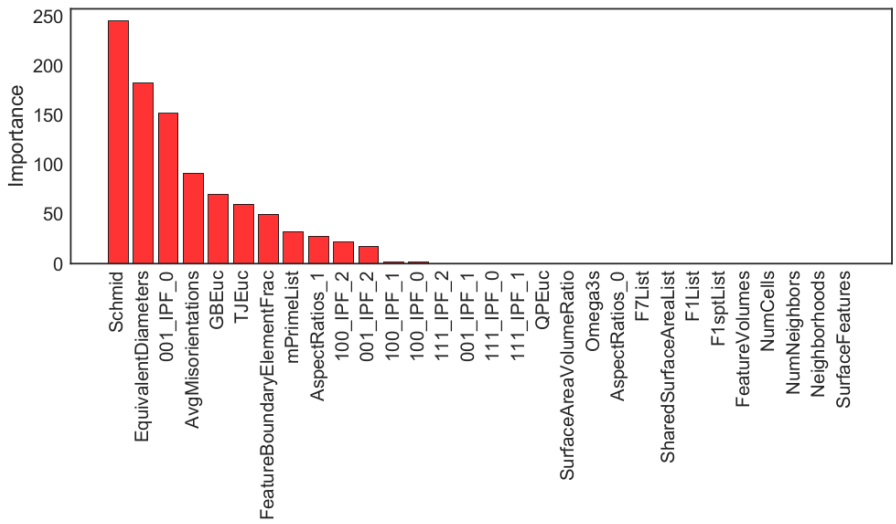


Figure 6: Variable importance in FCC materials: Both texture and geometry derived microstructural features are selected by FeaLect algorithm. The importance scale is arbitrary and set by Fealect.

Fig 5. Feature importance scores calculated using Fealect algorithm[2]

Random forest models (combination of multiple decision trees) are also capable of giving us feature importance scores, which can increase our understanding of the factors most crucial for knowing the locations of stress hotspots. But these scores can have correlation bias, since decision trees algorithms work on greedy algorithm whose main goal is to reduce the length of the decision tree.

Hence, this research paper used Fealect algorithm for determining the feature importance scores. Fealect algorithm computes these scores using L1 regularization. From above we can infer that both crystallographic descriptors and geometric descriptors play an important role in determining the location of stress hotspots.

Thus, from this research paper we saw how using supervised machine learning models we can take use of microstructural features to predict the locations of stress hotspots.

2.3. Machine learning enabled surrogate crystal plasticity model for spatially resolved 3D orientation evolution under uniaxial tension, Pandey and Pokharel (2020)

Understanding material properties at mesoscale through CPFEM requires a lot of computational resources and time, a lot of research is being done to come up with machine learning models that can reduce the resources required for understanding the

mesoscale properties.

In this research paper, they have tried to implement a recurrent neural network that can predict the microstructure evolution under uniaxial tension. Recurrent neural networks are supervised learning models that provide us the capability to model complex sequential data. Recurrent neural networks are a type of artificial neural network that have loops providing them the ability to get feedback from previous states.

In this paper, they prepared the data for strains ranging from 1% to 12% with an increment of one, thus creating a sequential data that can be fed in the RNN. They also tried increments of 2% and 3% to verify the effectiveness of increment size.

For synthetic microstructure generation they relied on the DREAM.3D and EVPFFT model respectively. They generated structures of 16*16*16, 64*64*64 and 128*128*128 voxelized mesh.

The most interesting aspect of this paper, was to prepare the data for 3*3*3 voxel instead of taking full structure of 16*16*16 in one go. The 3*3*3 voxel contained one element at the centre and 26 neighboring elements, thus enabling us to provide information about the local neighborhood. The advantage of this method is the number of training points we have increases drastically, because in a N*N*N voxel we will have (N-2)^3 such 3*3*3 voxels. Here (N-2) is there because we will not consider the elements on edges. For example, if we have a microstructure with 64*64*64 voxels, then we will get 62^3 training points that provide us with immediate neighborhood information.

For organising the training data a 3*3*3*3 data structure was created to represent the 3*3*3 structure and the corresponding euler angles of each voxel.

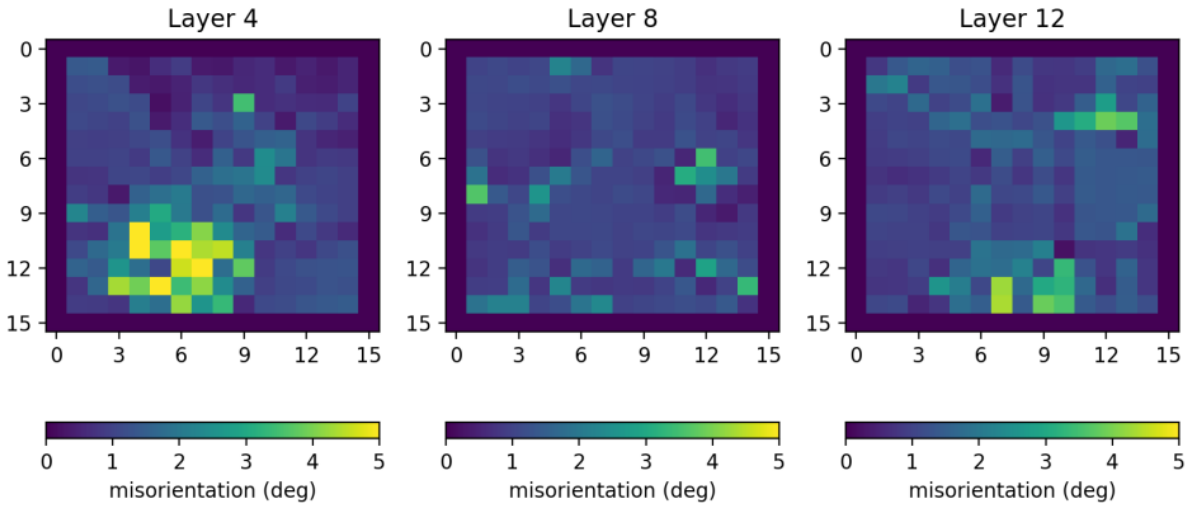
Training data format for a series of time steps for each voxel at a given position j in the RVE.

X	**Y**
X_0^j	V_1^j
X_1^j	V_2^j
X_2^j	V_3^j
$\vdots$	$\vdots$
X_{11}^j	V_{12}^j

Fig 6. Training data format fed into LSTM[3]

So for input we had euler angles of 3*3*3 voxels at (i)% strain rate and the corresponding output was euler angle of the centre element at (i+1)% strain rate.
The next stage was the training of LSTM model, different models were trained with variations in increment of strain rate and the number of data points.
The metric for gauging the performance of the model was the disorientation angle between the LSTM predictions and EVPFFT(ground truth). Basically, what disorientation angle gives is how different two sets of euler angles are.

The results obtained from the model were very close to the ground truth (EVPFFT). The disorientation angles between LSTM and EVPFFT predictions lied below 5 degrees and the mean of disorientation angles was around 2 degrees.



c) The disorientation map between the ground truth and LSTM-CP Model I predicted from 0% to 1% strain for the different layers (layer 4, 8 and 12) of the 3D microstructure structure shown in Fig.2.

Fig. 7 Disorientation angles being used as a metric for comparison between EVPFFT and LSTM-CP results[3]

The assumption in this paper is that the voxel is affected only by interactions from its immediate neighborhood. Also, once the model is trained the model can be used for any microstructure size.
Although, disorientation angle is quite high at some points, that can be solved if we use a combination of LSTM and EVPFFT. For instance, we can use LSTM in initial

increments and then continue with EVPFFT to fine tune. Thus providing us with both accuracy and less use of computational resources.

In this paper, only the evolution of euler angles is calculated. The future work can include the evolution of stresses.

2.4. Microstructural Materials Design via deep adversarial learning Methodology, Yang and Li (2019)

In this paper another deep generative model, GANs were used for content generation with the combination of GPR. GANs are based on game theory, it trains two models namely, Generator and discriminator. The motive of the generator is to generate images as close to the real image, and the discriminator's motive is to get better at identifying the difference between real images and the ones produced by the generator.

The advantage in GANs is that it doesn't have to deal with any explicit probability density estimation, whereas VAE works on the assumption that data lies on a normal distribution. Hence, VAE produces blurry images when compared with GANs. But GANs come with some difficulties as well, it's quite hard to train GANs.

The objective of this paper was to find the material suitable for cell design, physics based simulation RCWA was used to determine absorption index of a given microstructure. But as with CPFEM, RCWA is also a very expensive simulation to run. Therefore, GAN was combined with GP Hedge bayesian optimisation to reduce the number of trials for absorption. The result was that they found the microstructure with optical absorption of 0.74, as compared to the training data average of 0.65.[4]

2.5. Prediction of Mechanical Properties by Artificial Neural Networks to Characterize the Plastic Behavior of Aluminum Alloys, Merayo and Prieto(2020)

In metal forming, formability of a metal depends upon several factors. They are generally studied through models of flow curves, supplemented with FEM and empirical relationships. But the drawback with empirical relationships is that they are specific for some alloy composition and don't generalize for different compositions.

So this paper tried to find yield strength and ultimate tensile strength of an alloy using artificial neural networks, the inputs given to the model were brinell hardness, alloy chemical composition and tempers. Open source dataset of wrought alloys was used for training. The resultant model gave very close results when compared to empirical relationships, with the error of about 2%. But considering that ANNs gave us more generality in terms of alloy composition, ANNs can be employed in more such industrial applications to find mechanical properties. [5]

2.6 Predicting microstructure-dependent mechanical properties in additively manufactured metals with machine- and deep-learning methods, Heriott and Spear(2020)

In this paper, three models namely ridge regression, XGBoost and CNNs were employed to predict the mechanical properties of additively manufactured steel. Ridge regression and Xgboost model required a lot of preprocessing to extract the features out of microstructure images, whereas in CNNs we can directly feed the images.

In results, we saw that CNNs in which images were given alongside texture orientation performed the best, whereas CNNs in which images and grainIDs were given performed poorly even in comparison with ridge regression and Xgboost.

Thus, from this research paper we can conclude that CNNs given microstructure images and texture information can be used for the design of additively manufactured materials.[6]

2.7. Towards the prediction of intergranular fatigue crack initiation in metals due to hydrogen, Aman Arora(2020)

Hydrogen has many applications in different industries, however hydrogen penetration into material surface causes a reduction in the ductility thereby reducing the useful lifetime of a material. Around eighty percent of the useful lifetime of a material goes into the crack initiation process. Hence if we can delay the onset of crack initiation, the useful lifetime of a material would go up significantly.

An understanding of the factors affecting crack initiations can help us to design materials to increase the useful lifetime of a material. Intergranular boundaries are

most vulnerable to crack initiation due to the accumulation of hydrogen in those sites.
An experimental framework was followed for finding the relation between microstructural features and crack initiation locations. Insitu low cycle fatigue loading was done under scanning electron microscope for investigating crack initiations in both hydrogen charged and uncharged specimens.
Then the factor of $\Delta E/\Delta m$ was investigated for each grain, where ΔE represents the difference of elastic modulus in loading direction and Δm is the maximum difference within neighboring grains.
It was concluded that in hydrogen charged specimen, crack initiation occurred at grains where $\Delta E/\Delta m$ was higher.
Therefore this shows that we can predict crack initiation locations if we have the prior knowledge about microstructural features. Moreover, this task can be made much easier using machine learning wherein the feature processing would be done by the algorithm itself.[7]

2.8. PRISMS-Fatigue computational framework for fatigue analysis in polycrystalline metals and alloys

Several crystal plasticity subroutines or stand alone codes have been developed that can be used with commercial softwares like ABAQUS, but the problem with them is the reproducibility of results and changing those subroutines is very inefficient and poses problems in verification and validation. The alternative to this is an open-source framework which has proper documentation that can be used to model complex fatigue problems, without the need for reimplementing basic routines. Hence, PRISMS- Fatigue can be used as a data generation tool to simulate granular microstructural features under fatigue loading. Another advantage of using PRISM-fatigue over ABAQUS is the reduction in time required for simulating, as PRISMS-fatigue uses some form of modified Newton raphson wherein sparse stiffness matrix in only computed at start of each increment rather than each iteration of an increment,. This modified Newton Raphson works well in mildly nonlinear problems.
The parameters developed to compare microstructures while modelling fatigue deformation are Fatigue Indicator parameters (FIPs). FIPs serve as a measurable

parameter which can help us to gauge the driving force of crack initiation at granular level.

PRISMS-Fatigue consists of two packages : 1. PRISMS Plasticity and deal-II library built upon C++, which acts as a Crystal Plasticity-FEM tool; 2. Python code to compute FIPs over each grain, basically acting as a post processing tool.

Another aspect of any FEM tool is the capability to apply desired boundary conditions, fully periodic boundary conditions will be applied, to represent the simulation of bulk specimens. Przybyla[8] has explained the constraints needed to apply periodic boundary conditions. Linear constraints on opposing faces, vertices and edges are specified. Additionally some constraints are applied to eliminate rigid body motion. [9]

2.9. Inferences from literature review

1. Machine learning with their innate ability to model complex features were able to correlate microstructural features with mechanical properties like ultimate tensile strength. As per the author's knowledge, no major research has been done to predict fatigue crack initiations from machine learning.
2. Aman Arora and Mahajan (2020), was able to demonstrate that microstructural features can be used to predict fatigue crack initiations by taking use of an experimental framework.
3. An open source framework: PRISM-Fatigue can be used for the data preparation which will be used to train the machine learning algorithm.

Chapter 3

Methodology

Generating data through experiments is a very tedious task, considering the size of data points needed for a machine learning model. Hence, a computational approach would be followed for generating the data. Synthetic microstructures will be simulated using open source tools DREAM.3D and PRISMS-Plasticity.

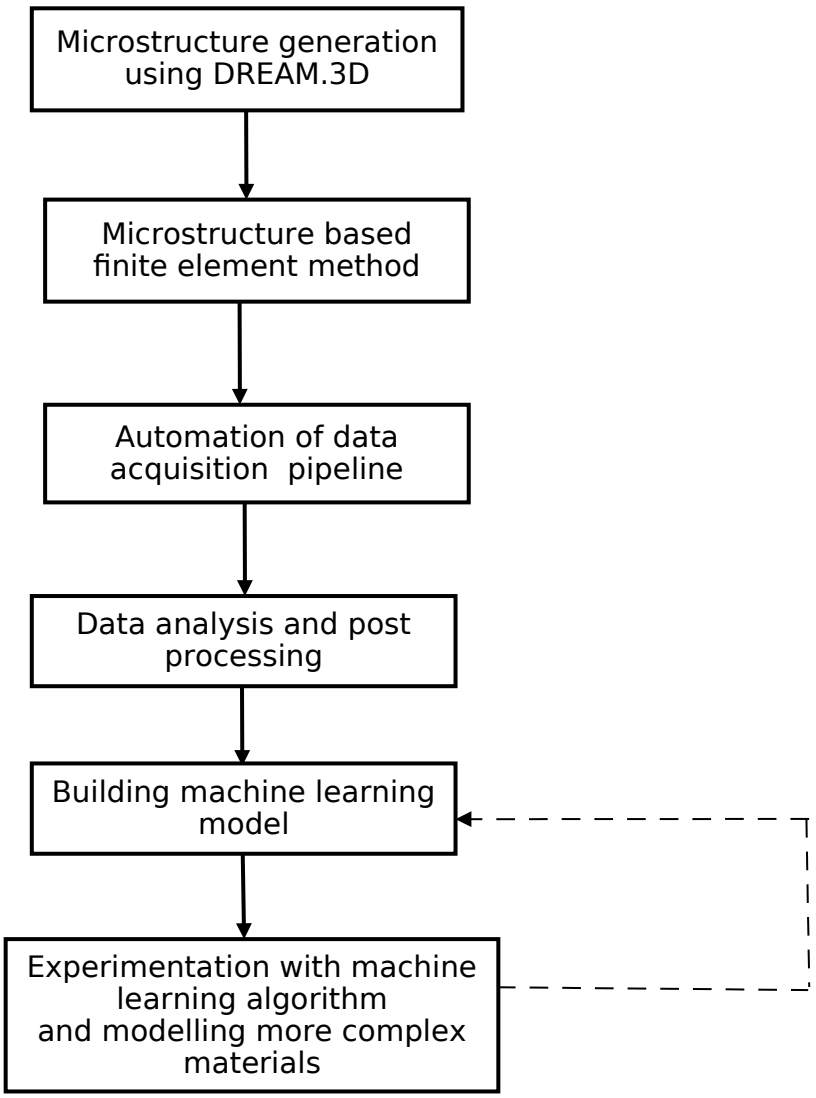


Fig 8. Schematic workflow of the steps involved in building a machine learning model for prediction of intergranular crack initiations

3.1 Synthetic microstructure generation using DREAM.3D

DREAM.3D is an open source tool to analyse and reconstruct microstructure data. It has a host of filters that can be used to process the microstructure data that we get from experiments. It also provides the capability to make artificial random data if we don't have the experimental data. The microstructures produced can also be meshed and simulated on CPFEM softwares like ABAQUS and PRISMS-Plasticity. We can also visualize the generated microstructure in Paraview.

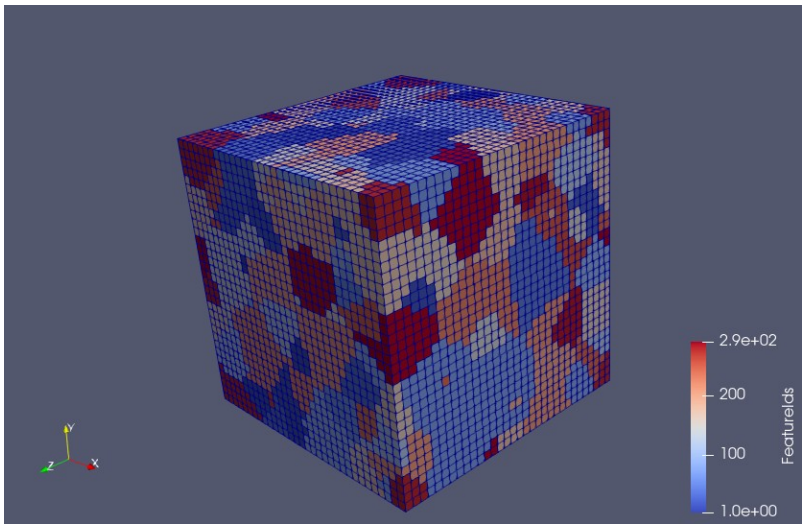


Fig 9. Synthetic Microstructure produced by DREAM.3D

There a lot of filters in DREAM.3D but 2 of them are most important to know:

1. StatsGenerator: We have to input information about number of phases, morphology, grain size, B/A, C/A elongation ratio, orientations distribution, function, misorientation distribution function. Experimental data from ESBD analysis can also be fed into the model
2. Initialize synthetic volume: Dream.3d creates an instantiation and also produces a mesh with this filter. We have to tell the number of voxels in each direction and the resolution (size) of the microstructure.

Other than these 2 filters, all other filters are optional and depend on the application of the user.

For running CPFEM in PRISMS Plasticity, two input files from DREAM.3D are needed-

GrainID.txt : Information about which voxel contains which grain

Orientation.txt : Rodrigues angles of all the grains

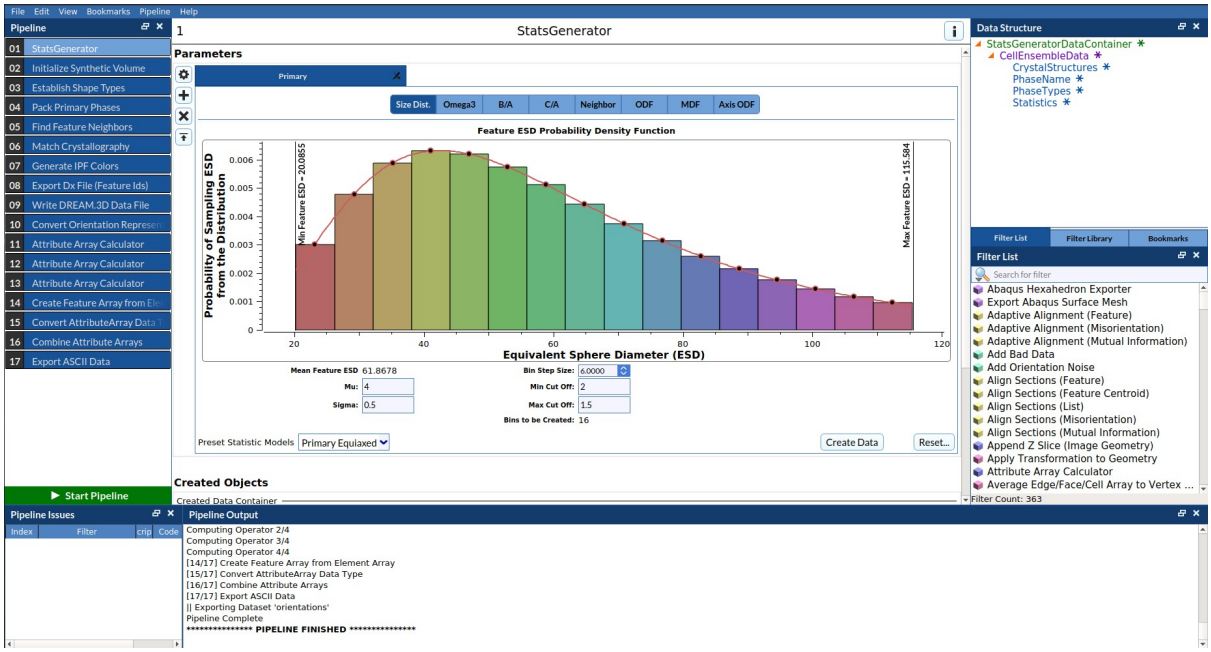


Fig 10. Different filters required in DREAM.3D

3.2 CPFEM Simulations using Open source Software PRISMS-Plasticity

A highly efficient open source CPFEM software: PRISM Plasticity[10] will be used for generating mechanical properties of the synthetic microstructures we produced from DREAM.3D . The element type considered is 3D Hex. The FEM will provide us with stress-strain values, evolution of texture (euler angles), and slip activity.

The input files required for running the simulation in PRISMS-Plasticity are:

1 Latent Hardening ratio.txt: Latent hardening ratio provides us information about the way in which different slip systems interact with each other/

2 Slip Directions.txt and Slip Normals.txt: The orientation of different slip directions and slip planes is defined.

3 BCinfo.txt: Different types of boundary conditions can be applied by defining the velocity gradient tensors or displacements. Furthermore, we can define cyclic, tabular or periodic boundary conditions as well.

4 GrainID.txt and Orientation.txt: Imported from DREAM.3D

5 Prm.prm : This is the most important file, which defines all the parameters for meshing, elasticity constants, order of gaussian quadrature and basis functions.

In Output Files, we get:

1 StressStrain.txt: This gives values of green strain and Cauchy stress at each increment of strain .
2 Visualization files: These files can be post processed to visualize the results in Paraview.
3 QuadratureOutputs.csv: It gives coordinates, rodrigues angles, elastic deformation gradient tensor, Stress tensor, slip activity for every quadrature point at specific increments. The number of increments after which we get this file, is to be defined in prm.prm file.[11]

3.3 Data Processing

Arranging the data in a form which can be fed in a machine learning model is one of the most crucial steps of any data science project. Hence, python data processing libraries like numpy and pandas will be used to process the data processed by CPFEM.

3.4 Building Machine learning model

A supervised learning model would be implemented to find the relationship between microstructural features and mechanical properties. Experimentation would need to be done to come up with the machine learning model that provides us with desired performance.

Chapter 4

Conclusion and Future Work

A new computational approach is discussed for prediction of intergranular crack initiation using machine learning algorithms. This would enable us to design materials without going through the hassle of conducting experiments or running Crystal Plasticity simulations which require use of extensive computational resources.

Several machine learning algorithms like CNNs, RNNs and gaussian processes were dicussed that can be used to correlate microstructural features with mechanical properties.

First step in implementation would involve data preparation by generating synthetic microstructures in DREAM.3D and simulating them in PRISM-Plasticity. Till now, we have generated and simulated few microstructures for getting familiar with the forementioned tools. Second step would be to automate these tools for data acquisition, considering the size of microstructures needed to be simulated. The last phase would involve data processing and building the machine learning model for prediction of intergranular crack initiations.

Experimental research is also being conducted in this direction by my lab colleague Mohit Singh, which will help us in validating the machine learning model.

Tentative Timeline

Work\Month	July 2021	Aug 2021	Sep 2021	Oct 2021	Nov 2021	Dec 2021	Jan 2022	Feb 2022	Mar 2022	April 2022	May 2022
Literature survey	X	X	X	X							
Phase 1: Understanding Crystal Plasticity-FEM software and automating data acquisition pipeline			X	X							
Phase 2 : Data analysis and post processing					X	X					
Phase3: Building the machine learning model						X	X				
Phase 4: Experimentation of machine learning algorithm and complex materials							X	X	X	X	
Phase 5: Writing thesis										X	X

References

[1] Y. Kim *et al.*, "Exploration of optimal microstructure and mechanical properties in continuous microstructure space using a variational autoencoder," *Mater. Des.*, vol. 202, p. 109544, 2021, doi: 10.1016/j.matdes.2021.109544.

[2] A. Mangal and E. A. Holm, "Applied machine learning to predict stress hotspots I: Face centered cubic materials," *Int. J. Plast.*, vol. 111, no. October 2017, pp. 122–134, 2018, doi: 10.1016/j.ijplas.2018.07.013.

[3] A. Pandey and R. Pokharel, "Machine learning enabled surrogate crystal plasticity model for spatially resolved 3D orientation evolution under uniaxial tension," no. 2017, 2020, [Online]. Available: http://arxiv.org/abs/2005.00951.

[4] Z. Yang, X. Li, L. C. Brinson, A. N. Choudhary, W. Chen, and A. Agrawal, "Microstructural materials design via deep adversarial learning methodology," *J. Mech. Des. Trans. ASME*, vol. 140, no. 11, pp. 1–34, 2018, doi: 10.1115/1.4041371.

[5] D. Merayo, A. Rodríguez-Prieto, and A. M. Camacho, "Prediction of mechanical properties by artificial neural networks to characterize the plastic behavior of aluminum alloys," *Materials (Basel).*, vol. 13, no. 22, pp. 1–22, 2020, doi: 10.3390/ma13225227.

[6] C. Herriott and A. D. Spear, "Predicting microstructure-dependent mechanical properties in additively manufactured metals with machine- and deep-learning methods," *Comput. Mater. Sci.*, vol. 175, no. February, p. 109599, 2020, doi: 10.1016/j.commatsci.2020.109599.

[7] A. Arora, H. Singh, and D. K. Mahajan, "Towards the prediction of intergranular fatigue crack initiation in metals due to hydrogen," *Mater. Sci. Eng. A*, vol. 787, no. May, p. 139488, 2020, doi: 10.1016/j.msea.2020.139488.

[8] C. P. Przybyla, "Microstructure-Sensitive Extreme Value Probabilities of Fatigue in Advanced Engineering Alloys," *PhD. thesis,Georgia Inst. Technol.*, no. August, p. 2010, 2010.

[9] M. Yaghoobi, K. S. Stopka, A. Lakshmanan, V. Sundararaghavan, J. E. Allison, and D. L. McDowell, "PRISMS-Fatigue computational framework for fatigue analysis in polycrystalline metals and alloys," *npj Comput. Mater.*, vol. 7, no. 1, 2021, doi: 10.1038/s41524-021-00506-8.

[10] M. Yaghoobi *et al.*, "PRISMS-Plasticity: An open-source crystal plasticity finite element software," *Comput. Mater. Sci.*, vol. 169, no. June, p. 109078, 2019, doi: 10.1016/j.commatsci.2019.109078.

[11] M. Yaghoobi, S. Sundar, and A. Lakshmanan, "PRISMS-Plasticity user manual-V1.2.0."