IOP Conference Series: Materials Science and Engineering

PAPER • OPEN ACCESS

Influence of synthetically generated inclusions on the stress accumulation and concentration in X65 pipeline steel

To cite this article: N. Fehlemann *et al* 2021 *IOP Conf. Ser.: Mater. Sci. Eng.* **1157** 012056

View the article online for updates and enhancements.

Influence of synthetically generated inclusions on the stress accumulation and concentration in X65 pipeline steel

N. Fehlemann, Y. Sparrer, F. Pütz, M. Könemann and S. Münstermann

Integrity of Materials and Structures, Steel Institute, RWTH Aachen University, Intzestraße 1, 52072

E-mail: {niklas.fehlemann, yannik.sparrer, felix.puetz, markus.koenemann, sebastian.muenstermann}@iehk.rwth-aachen.de

Abstract. The use of a simulative approach with representative volume elements (RVE's) is particularly well suited to investigate the influence of different microstructural parameters on the damage behavior of a material. In order to statistically analyze the individual components of the microstructure (e.g. geometric structure of grains and inclusions), well-known distribution functions such as logNormal/Gamma are normally used, but these do not take into account the interdependencies between the different parameters. However, newer approaches like machine learning techniques can only describe one phase of a single material at a time. Therefore, in this study, we extended an existing Wasserstein Generative Adversarial Network (WGAN) to a Conditional Wasserstein GAN with Gradient Penalty (CWGAN-GP), with which it is possible to process multiple materials/phases simultaneously. Training this algorithm on different steels and associated inclusions showed that a single trained network can generate synthetic microstructure for all different phases and materials with very high quality. A newly implemented evaluation method using the regularized Wasserstein-distance confirmed the excellent agreement of the real data with the synthetic data for all phases/materials. As a use case for our algorithm, the influence of different inclusions on the stress accumulation and concentration of X65 pipeline steel was investigated to find initiation sites for damage in the material. These investigations showed a pronounced correlation between stress concentration and inclusion parameters, thus confirming the usefulness of the CWGAN-GP as an input-generator for RVE's.

1. Introduction

With more and more demanding tasks, the microstructure and properties of modern steel grades are becoming exceedingly complex. These materials show intricate interactions between their microstructure and the damage mechanisms. [1] For example, steel grades like modern pipeline steels are strongly influenced by the microstructure; in particular, the damage behavior depends highly on the non-metallic inclusions in the matrix. [2] One important tool to study microstructural behavior is the statistically representative volume element (RVE). [3] A simulative approach using these elements is particularly suitable for the analysis of the microstructure, because in this way the individual components of the microstructure can be better separated from each other and analyzed individually. A RVE is a volume element that represents the microstructure on a statistically vaild base, while it must be small enough to differ from the macroscopic scale. [4] The process of the RVE-generation is a separate, sophisticated area of research, more details can be found in Henrich et al. [5] In this study, the focus should be on the statistical describtion of the microstructural parameters.

The starting point of every RVE-Simulation is a careful and sophisticated study of the microstructural properties of the desired material. There are several techniques to archieve this goal: In the much more easy 2D-case, the data can be extracted using light optical microscopy (LOM) and scanning electron microscopy (SEM). The 3D-case is more difficult, techniques like serial-sectioning of the microstructure are time consuming and difficult. [6] In most of the cases, the microstructure is analyzed using SEM-techniques, with the help of an EBSD (Electron Backscatter Diffraction) detector. All of the important microstructural properties, such as grain size, aspect ratio, grain slope and so on (e.g. properties of non-metallic inclusions in the microstructure) have to be characterised to build an RVE as accurate as possible. [7] [8] After extracting the grain data, a distribution (most commonly a logNormal or Gamma-distribution) can be fitted to the data. [9] From these distributions, input data can now be sampled for the creation of RVE's . To simplify the generation, the combination of area and aspect ratio can be re-written to get the major and minor axis of the grain.

Several problems remain with the use of this approach: First, the interdependencies between the individual parameters are not taken into account if only univariate distribution functions are used. Figure 1 from a previous study [10] depicts this problem. In the picture, the relations

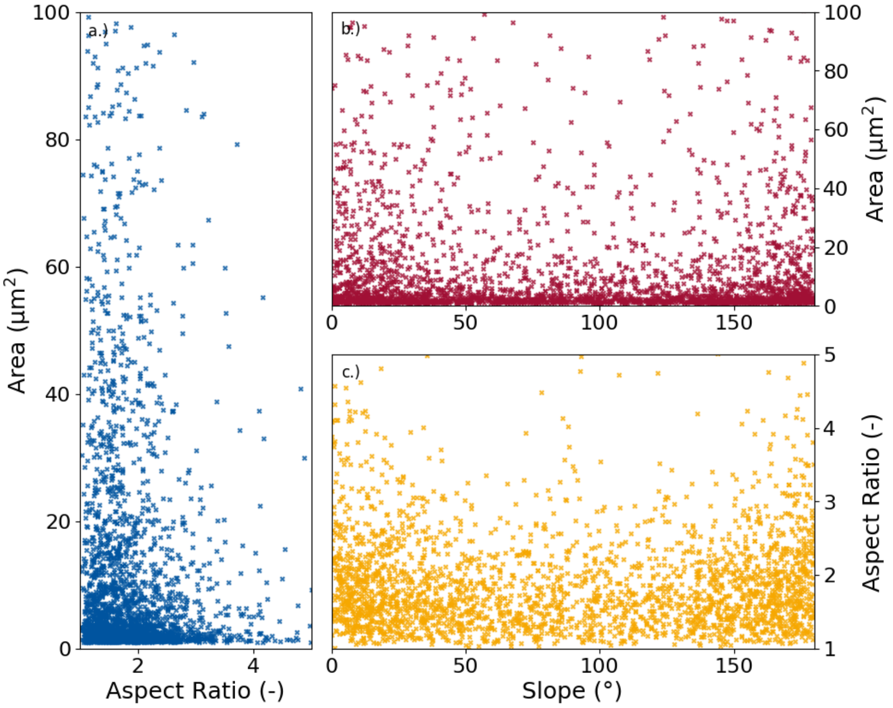


Figure 1: Scatterplots for the relations between the main parameters grain size, aspect ratio and slope [10]

between the grain parameters area, aspect ratio and slope are shown using three separate plots (The material is a dual-phase steel DP800). With these plots, a few dependencies can be identified: Bigger grains show comparable smaller aspect ratios (left picture). These more elongated grains with smaller AR's also tend to be more often horizontally orientated. This is depicted in the lower-right picture. In the top right picture, it can also be seen that larger grains tend to show orientations around 0 or 180°. These observations show that interdependencies are present and confirm the problem described in the previous section. Furthermore, distribution functions must be generated for all required parameters (e.g. for all grain parameters of the matrix and the inclusions). This is a very time-consuming and error-prone process.

In this work, we present an improved approach to generate more realistic artificial microstructure, which is also less error-prone and time-consuming. For this purpose, techniques from the field of machine learning are used. We want to show that our approach can map large parts of the microstructure at once. The use case for our system is the investigation of the influence of inclusions on the stress concentration and accumulation and therefore damage behavior of X65 pipeline steel. Based on these investigations it can be shown that an accurate characterization of the different components of a microstructure is extremely important.

2. Machine Learning algorithms

For problems that deal with the patterns in complex data, machine learning approaches are very suitable. If the dataset becomes larger and the interdependencies become even more complex, deep learning methods and the use of neural networks can provide very good results. [11] A neural network is an algorithm that replicates the functionality of the human brain. These nets consist of an input layer and an output layer with a number of hidden layer between input and output. In a standard feed-forward-network all of the neurons are connected with each other (Which is also called fully-connected layer). [11] The learning-process of this kind of net is called backpropagation: By using a loss-function, the error is backpropagated through the network and the weights are updated using the gradient of the loss. [12]

One of the best approaches to reproduce a given distribution and generate synthetical data with machine learning is to use a so called generative adversarial networks , introduced by Goodfellow et al. 2014. [13] In a study carried out by Pütz et al., a Wasserstein Generative Adversarial Network (WGAN) was used to generate synthetical microstructure which can map the interdependencies of the different grain parameters. [10] Training these kind of network on the ferrite grains of a DP800 dual phase steel showed impressive results.

However, this algorithm is only capable of processing one set of input data (e.g. one material or phase) at a time. To process various materials/phases at the same time, to enhance the results and the stability and to lower the training time, the WGAN was further developed into a *Conditional WGAN with Gradient Penalty* (CWGAN-GP) in this study. The characteristics of this system are described below.

The CWGAN-GP is a developement of the normal WGAN, which itself is a developement of the original generative adversarial network. [14] All GAN's consist of two neural networks

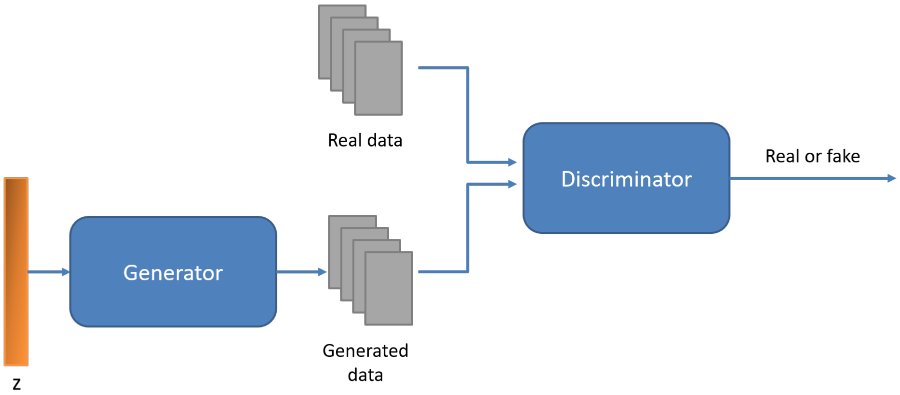


Figure 2: Schematic representation of a Generative Adversarial Network.

wired against each other playing a minmax-game. A GAN is schematically depicted in figure 2: The first network is the so called generator (G). G gets a noise vector sampled from a normal distribution as an input and generates fake data which is passed to the discriminator (D). D takes fake and real data samples as an input and tries to distinguish between the real and the fake data. The aim of the training process is that the distribution p_G becomes equals to the real distribution p_{data}. To archieve this, the generator tries to create samples of fake data for which the discriminator is unable to distinguish them from the real data. In contrast, the discriminator tries to distinguish as well as possible between real and generated data. There are various different forms of GAN's which aim to improve the quality of the synthetically generated data and to stabilize the training process. [15] [16]

One successful approach is the aforementioned WGAN or its close relative, the WGAN-GP. The main difference between the original formulation of the algorithm is that the WGAN uses a different loss-function, namely the Wasserstein-distance, a distance metric in a multi-dimensional space. [17] The Wasserstein-distance is also called Earth Mover's Distance, because an easy illustration of these concept is to imagine two pile of earth. In this case, the Wasserstein-distance is the minimal amount of earth that needs to be transferred to bring both piles in the same shape. Equation 1 shows the mathematical expression of the Wasserstein-distance as used in the WGAN-GP.

$$W(p_{data}, p_G) = \underset{||f||_L \leq 1}{sup} \; E_{x \sim p_{data}}[f(x)] - E_{x \sim p_G}[f(x)] \tag{1}$$

Here, W is the distance, E is the mean value, p_{data} the distribution of the data and p_G the generator's distribution. However, this distance is only valid for functions that fulfill the condition of *Lipschitz-continuity* (an aggravated form of normal continuity, marked with $||f||_L \leq 1$ in eq. 1). To ensure these continuity in the normal WGAN, the weights of the discriminator are clamped after each update step. However, this is not a good technique to ensure the Lipschitz-continuity, as argued in Arjovsky et al. [14] Considering this testimony, Guljarani et al. [18] proposed a new technique called gradient penalty. Because the norm of each 1-Lipschitz-continuous function is 1, a penalty term is added to the Wasserstein-Distance. These penalty-equation enforces Lipschitz-continuity and enable the WGAN-GP to map more complex representations of data. [18] With the mentioned *conditional*-modification, the algorithm becomes able to distinguish between different classes of data (e.g. different materials/inclusions). With this additional functionality, the algorithm can be used to classify and generate synthetic data from various materials and inclusions at the same time. For the sake of simplicity, the different sets of input-data are called *labels* in the ongoing text.

3. Training of the CWGAN-GP

To demonstrate that the new CWGAN-GP, unlike the old system, can handle multiple labels in one training process, additional materials were trained in addition to the X65: A dual-phase DP800 steel was selected for this purpose. Also, the data of the non-metallic inclusions in this DP800 were added. Before the training, input data had to be collected. Therefore, EBSD's were performed on all three steels to extract the aforementioned parameters, grain size, aspect ratio and slope. For the DP800, EBSD images were taken from all three directions, namely transversal x sheet normal (TDxSN), rolling x sheet normal (RDxSN) and rolling x transversal direction (RDxTD). This-data was processed using the MATLAB-extension MTEX to extract the grain parameters grain area, aspect ratio and slope. [19]

A slightly different procedure was applied to the DP800 inclusions: Light-optical microscopy images were taken from the three directions. Fiji [20] was used to extracted the inclusions from the light optical images also in the form of ellipsis with grain size, aspect ratio and slope. All the data was saved as a single .csv-file to be loaded into the CWGAN-GP. Summing up all seven

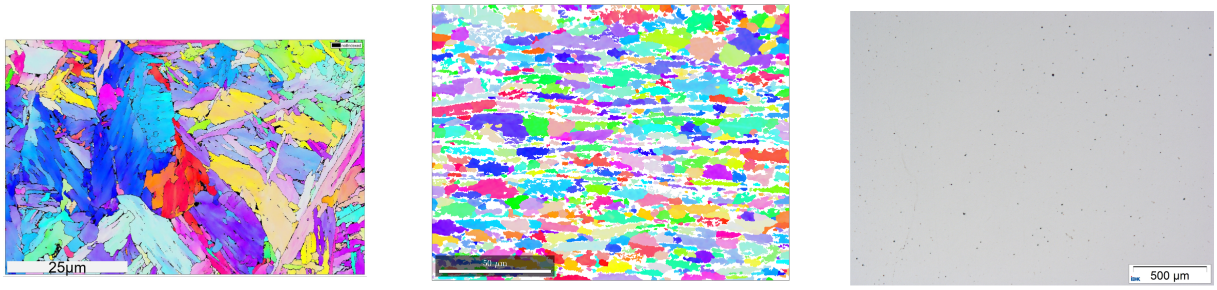


Figure 3: EBSD's for X65 (left), DP800 (right) and LOM for inclusions in DP800 (right) DP800-EBSD in the middle taken from [10]

different labels, there were approximately 15,000 points of training data available. The CWGAN-GP was implemented using Python 3.7 and the PyTorch-package. [21] A modular design was choosen making it easier to implement changes of the structure. The CUDA-technique and the cudNN-library were used to run the training on a GPU and to speed up the training significantly. The complete training process is illustrated in figure 4. As mentioned before, there is only *one* training process for all inputs but each label of data is evaluated *separately*. After the training process, the best state of the generator (saved as a serialized Python-object) can be used to generate synthetic microstructure whenever needed.

The CWGAN-GP algorithm requires so-called hyperparameters to be fully functional. Because the GAN-algorithm tends to be sensitive and unstable with respect to the choosen parameters, the choice of this parameters must be handled carefully. In this study, two simple feedforward-networks (FFN) were used as generator and discriminator, since the input is tabular . Both networks have two hidden layers between input and output layer which are 128 neurons big (also called width of the network). The size of the input/output layer slightly differs for G and D: For D, input is of size four, because there are four features and the output is one single neuron. For G, the input layer has the size of the noise vector z, which is 128 for the present system. The output layer has a size of four, because the output is synthetically generated data, which has four features. To deal with the different data, all data is scaled to the interval [-1,1]. To force the synthetic data to the same interval as the real data, a hyperbolic tangent (tanh) is used after the output layer of G. The activation function differs for both networks. For G, a tanh was found to work best, while a rectified linear unit (ReLU) shows best results for the discriminator. To optimize both FFN's, a RMSprop-Optimizer was used, as proposed in Arjovsky et al.. The batch size was set to 512, while the learning rate is 0.00005 for both nets. Two other important parameters are the dropout and the penalty parameter λ. Dropout is a commonly used technique to avoid overfitting by randomly disabling a certain percentage of neurons during each epoch. However, in this case it was found that 0% dropout works best with these architecture. λ was set to 0.1. The whole architecture was trained for 300,000 iterations of the generator, while a result was written out every 1000 iterations. After the training, the saved results were evaluated for every label and the best one is chosen for further usage.

To evaluate the training and find the best version of the generator evaluation, a special form of the Wasserstein-distance called regularized Wasserstein-distance (also called Sinkhorn-distance) is used. [22], [23] By calculating these distance between the real data and a batch of generated data after each iteration, a fast and easy to implement validation and evaluation technique is established. The sinkhorn-distance is shown in eq. 2

$$S_\epsilon = \underset{\pi \subset \Pi(\mu,\nu)}{min} \langle \pi C \rangle - \epsilon * H(\pi) \tag{2}$$

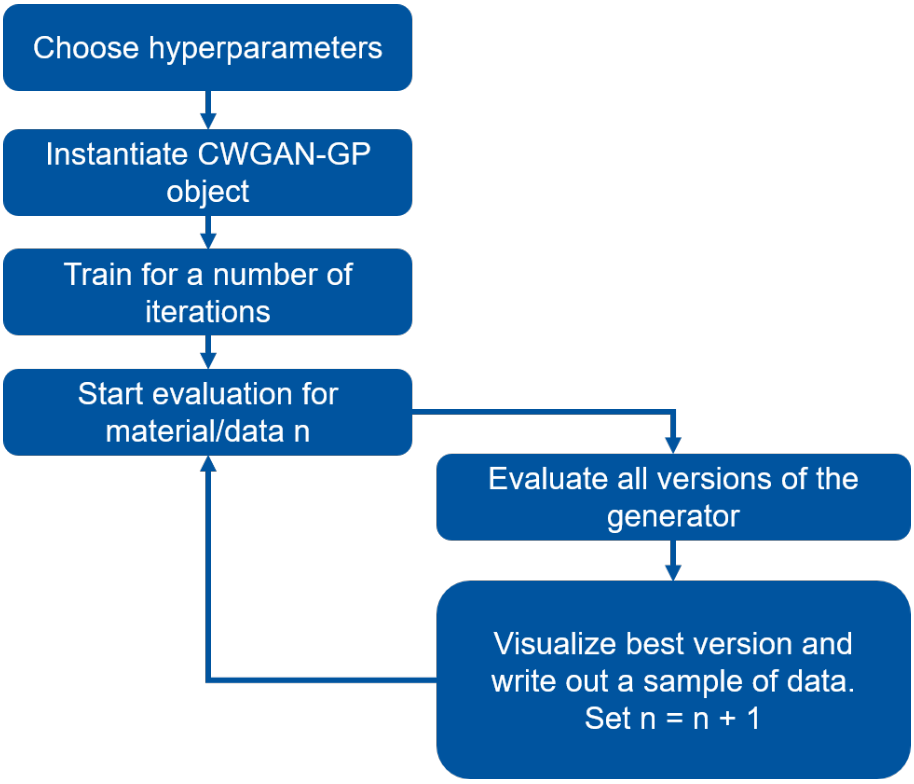

Choose hyperparameters
Instantiate CWGAN-GP object
Train for a number of iterations
Start evaluation for material/data n
Evaluate all versions of the generator
Visualize best version and write out a sample of data. Set n = n + 1

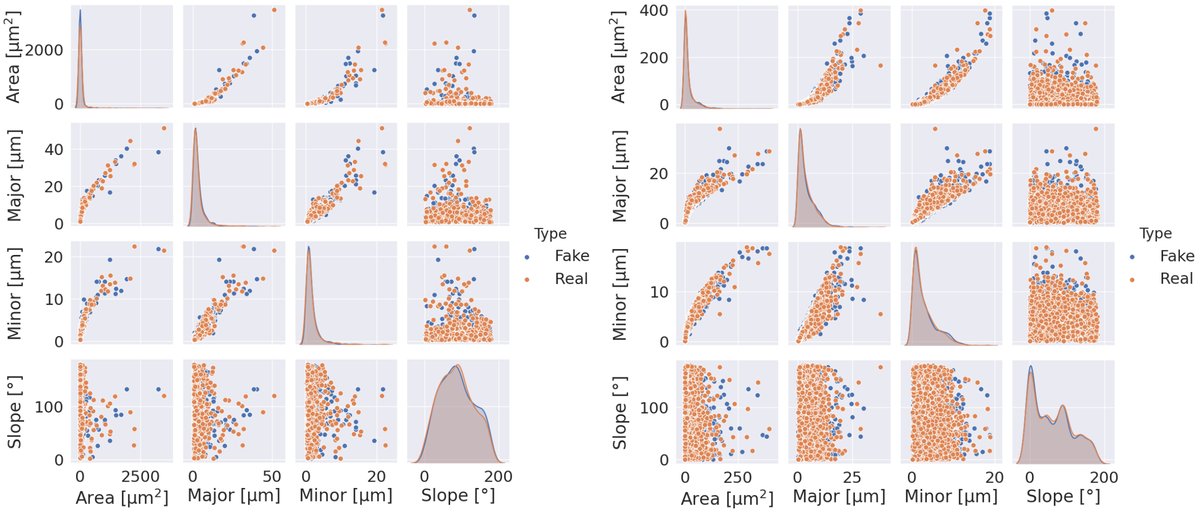
Figure 4: Workflow of the CWGAN-GP training and validating process

mu/nu are the two compared distributions, π is a so-called transport-plan to transform mu into nu (with Π as the set of all plans), C is the cost matrix and $H(\pi)$ is the entropy (information theory) of a transport plan. ϵ is a blurring parameter (here: 0.05). The use of methods like kde-comparisons or cluster analysis is therefore omitted (See Pütz et al. [10] for a more detailed description of the different evaluation methods).

4. CWGAN-GP Results

The training process explained in the previous section was used to show that multiple parts of the microstructure can be represented with a CWGAN-GP.

As stated above, the algorithm and the corresponding hyperparameter set were evaluated using the regularized Wasserstein-distance. A distance of exact 0.0 means that the compared point clouds/distributions are exactly equal. So, the closer the distance approaches a value of 0.0, the more similar are both point clouds and the better is the quality of the generator. Because the results are multi-dimensional, the best way to *optically* analyze the results is a so called *pairplot*. In this kind of statistical visualization, all the features are plotted against each other. Thus, this approach is best suited to visualize a multivariate distribution. In figures 5a and 5b, pairplots are shown for the X65 grain distribution (left) and the non-metallic inclusions in DP800 (right). These plots correspond with the EBSD/LOM-measures shown in figure 3. The real data are displayed with orange dots, the synthetically generated data points with blue ones. For both materials, the kernel density estimation for area as well as for the major and minor axis show distributions similar to a logNormal or Gamma-distribution (graphics on the inclined axis). In contrast, the approximation of the slope distribution using KDE's shows much more complex distributions with multiple peaks. Despite the different distributions of the various labels, the CWGAN-GP can mimic the distribution extremely well, since the blue and orange



(a) Best-fit pairplot for the X65 ferrite matrix at the optimal iteration 227,000

(b) Best-fit pairplot for the inclusions in DP800 at the optimal iteration 244,000 (RDxTD)

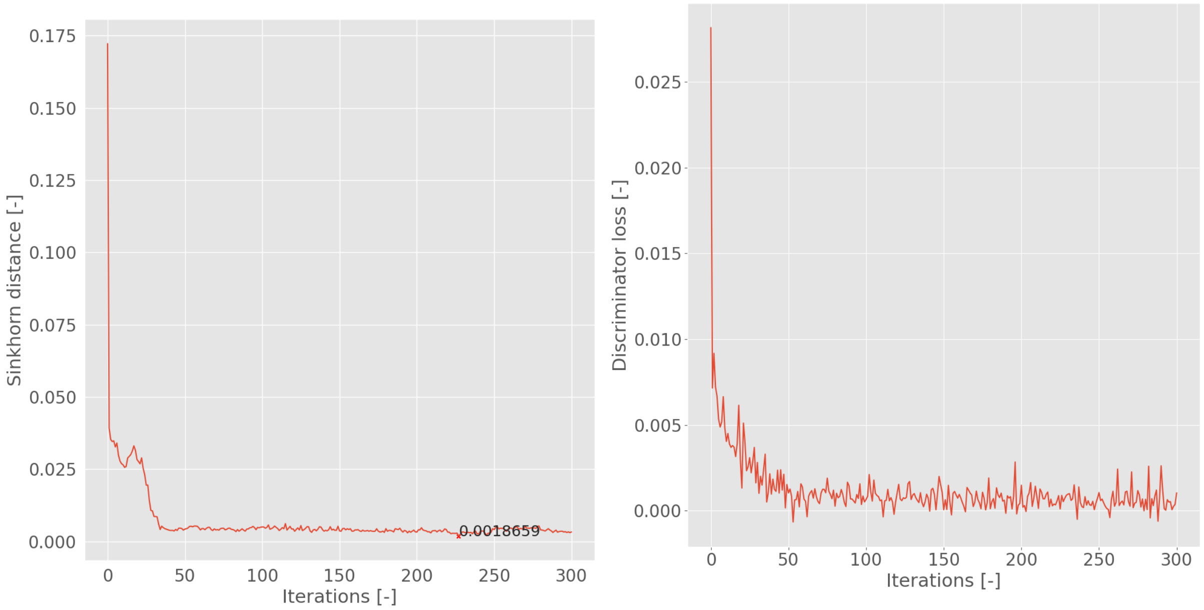
Figure 5: Pairplots for X65 and inclusions in DP800

kde-estimations fit almost perfectly on each other. Also, no outlier can be seen in the respective plots (e.g. slope vs. area) that could indicate problems with the generated distributions. These two data sets were chosen as an example here because it can be shown that the CWGAN-GP can process completely different labels (such as the grain parameters of one steel and the inclusions in another steel) at the same time in high quality.
The plots are made using the generator yielding the lowest Sinkhorn-distance between the real and synthetic data. The values for all seven labels are shown in table 1:

Label	Iteration (in 1000)	Loss
X65	227	0.00187
DP800 (TDxBN)	117	0.00383
DP800 (RDxBN)	49	-0.00058
DP800 (RDxTD)	181	0.00068
Inclusions (TDxBN)	17	0.00006
Inclusions (RDxBN)	207	0.00014
Inclusions (RDxTD)	244	-0.00032
Mean	–	0.00081

Table 1: Regularised Wasserstein-distance for all trained labels

As all of these values are extremely close to zero, it is shown that the CWGAN-GP can create almost perfect synthetic data for six materials/labels with only one training loop and only one hyperparameter configuration. Also, the distance-values confirm the optical results that can be seen in the pairplots.



(a) Regularized Wasserstein-distance for the X65 ferrite matrix (Iterations in 1000)

(b) Evolution of the Wasserstein-loss during the training process - For the whole dataset (Iterations in 1000)

Figure 6: Sinkhorn-distance and Wasserstein-distance

Another important criterion is the stability of the training process. For that, the evolution of the sinkhorn-distance during the training is shown. Note that the value is calculated every 1000 iterations with the 100. iteration shown as the starting point. The slope can be seen in figure 6a. These values are unitless, as it shows the real distance between the point clouds. It can be seen that the curve rapidly declines towards nearly zero and then remains constant with minor fluctuations. Only in the range of about 0 - 30,000 iterations there are a few stronger swings, but after these the curve is very smooth until the end of training. The lowest value during the training process and therefore the best iteration is marked with the blue cross.
A similar slope-behavior is observed for the the discriminators loss in the right picture (figure 6b), which falls from 0.025 towards zero with slightly stronger oscillations then the sinkhorn-distance. Because of the batch-wise loss computation of the discriminator, this curve does not allow to retrieve information about the *quality* of the CWGAN-GP or the generator. Nevertheless, the stability of the training process can be inferred from the curve, since no strong fluctuations occur. The smooth progression of both slopes indicates a high stability of the training process, which also leads to results of a very good quality as stated above.

5. Use case: The influence of inclusions on the stress accumulation and damage behavior in X65 steel

To emphasize the importance of the system described in the previous sections and thus an accurate, holistic description of the microstructure, the following use case was designed as an example:
It is known that the stress concentration in the microstructure of X65 strongly depends on the inclusions. The stress concentration and accumulation is an important indicator of the occurrence of damage, since the accumulation of stress mostly occurs at points of different ductility in the material, at which cracks and pores preferentially form.

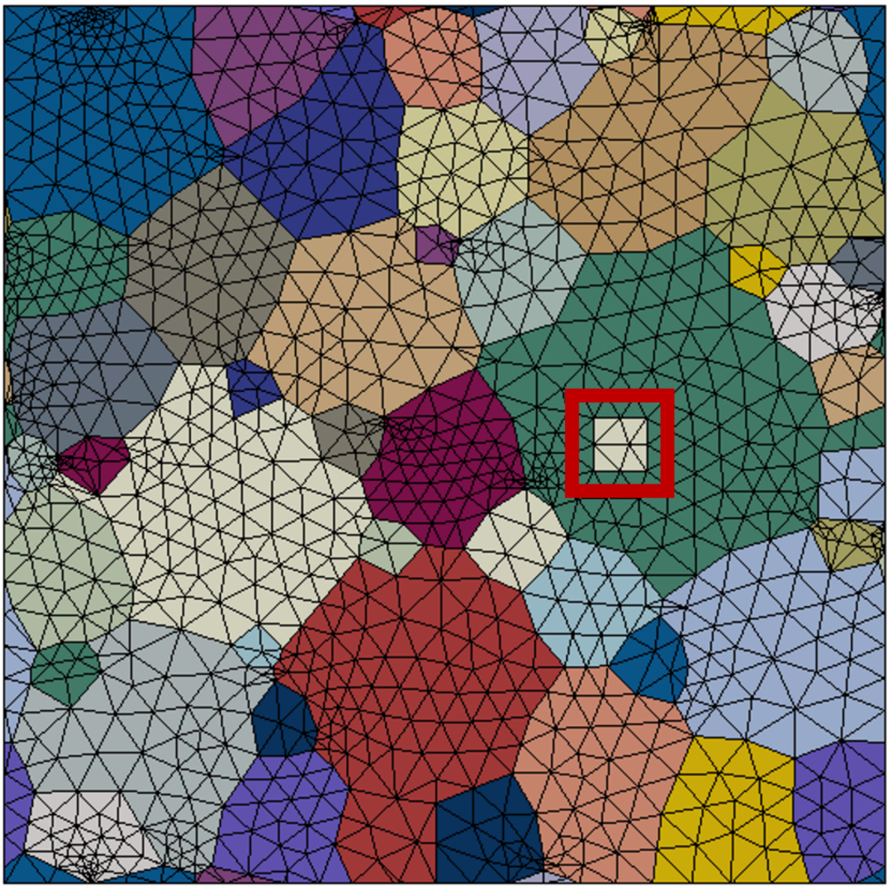
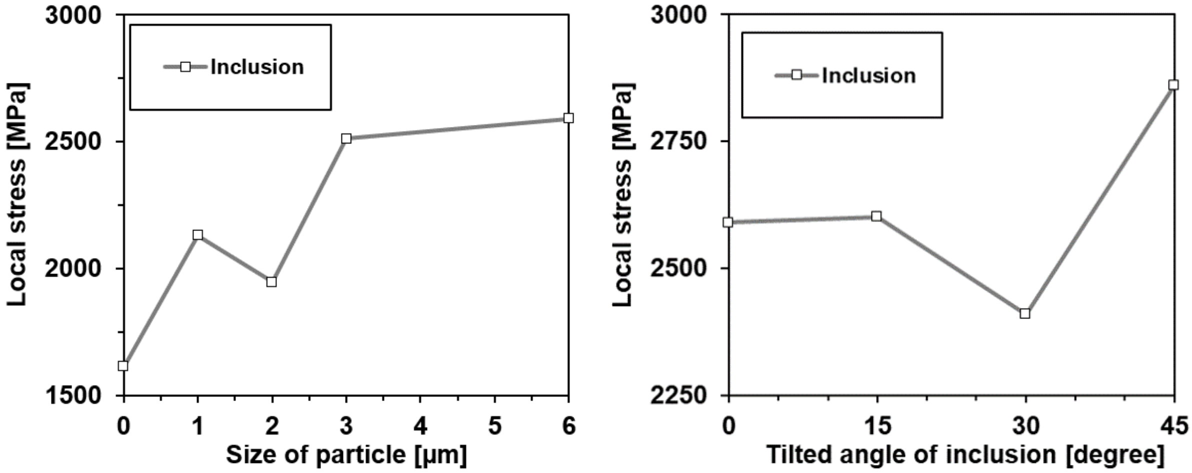


Figure 7: Meshed 2,5-dimensional RVE with inclusion on the right side (red rectangle)

Therefore, in order to show these influences, RVE simulations were performed. For this, a 2,5 dimensional RVE was constructed, as shown in figure 7. This RVE is $100\mu m$ x $100\mu m$ x $1\mu m$ in size and the inclusion is placed $20\mu m$ right from the center of the RVE. Note that the microstructure does not change in the thickness direction. A displacement of $15\mu m$ is applied to the top of the RVE. After the displacement, the stress concentration in the elements around the inclusion is analyzed. To display the inclusion, the mechanical properties of Titannitrid (TiN - table 2) are assigned to the square. A quadratic shape was choosen, because TiN-inclusions tend to have a rectangular shape. The simulations were performed using a crystal-plasticity-model, calibrated using Nanoidentation-tests. For a more elaborate explanation of the calibration scheme see Vajaragupta et al. [3] At first, size and tilted angle of the inclusion and the stress concentration was analyzed. Both slopes are shown in figures 8a and 8b, with the stress concentration w.r.t the size in the left picture and the stress concentration w. r. t. the tilted angle in the right picture. It can be seen, that the size of the inclusion has a major influence

Density	Young's Modulus	Poisson's ratio
5.12 g/cm^3	600 GPa	0,295

Table 2: Elastic properties of TiN



(a) Influence of the inclusion-size on the local stress concentration

(b) Influence of the inclusion-angle on the local stress concentration

Figure 8: Influence of size and tilted angle on the stress concentration

of the stress concentration. The stress value rises from 1600 MPa without a inclusion ("size" of $0\mu m$) to a value of more then 2500 MPa with a $6\mu m$ inclusion. The influence of the tilted angle is not as pronounced as the size. For an angle of 15°, there is no change in the level of stress concentration to the non-tilted inclusion (0°) of size $6\mu m$. By turning the inclusion by 30°, the stress level falls to approximately 2400 MPa. With the maximum tilted angle of 45°, the stress concentration level rises to around 2900 MPa. So, both graphics illustrate the pronounced influence of differently designed inclusions on the stress concentration.

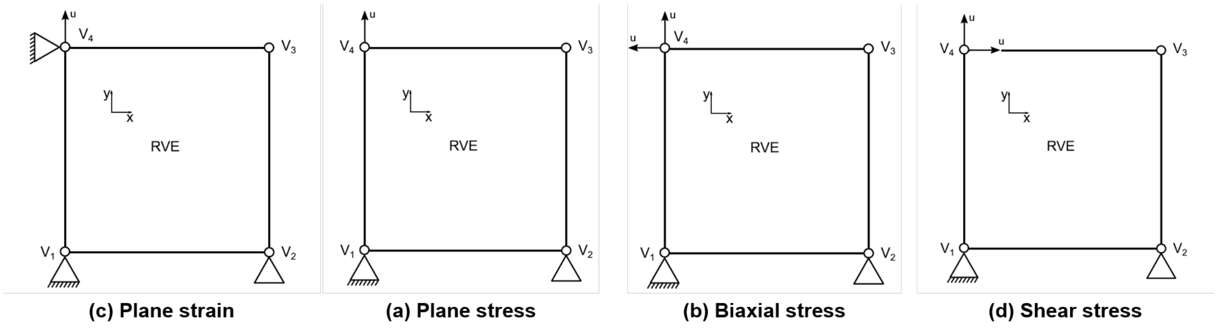


Figure 9: Loading conditions applied to the RVE

All of the aforementioned simulations were performed under plain strain conditions. For further investigations, different stress states were applied. Figure ?? shows the influence of different stress states and loading conditions on the stress concentration of the RVE's. The dashed line shows an RVE with a 45° rotated inclusion, the black line an RVE with a not rotated inclusion. Both inclusions are 6 μm in size. In opposite to the influence of size and angle, the influence of the stress state and loading conditions are less pronounced on the RVE's with inclusions. The local stress concentration is about 2500 MPa for the 0° rotated inclusion and around 2800 MPa for the 45° rotated inclusion and doesn't changes much for the different stress states. Despite

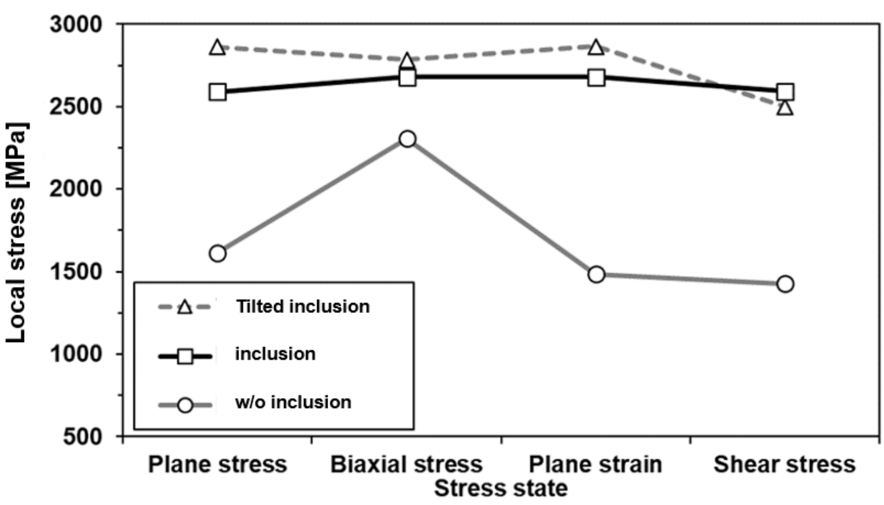


Figure 10: Influence of the loading conditions on the stress concentration

this behavior, the inclusion-less RVE's shows much more dependence of the stress state: For plain strain, the stress concentration is about 1500 MPa, which rises by about 33% to 2250 under biaxial loading conditions.

6. Conclusion

In this study, an improved algorithm for generating almost real synthetic microstructure was presented, which is capable of processing multiple labels of data at the same time. The results in chapter 4 show several advantages from moving from the *vanilla-WGAN* to the *CWGAN-GP*. These are the following:

(i) The algorithm can process multiple data sets at the same time, even if they are from different materials (such as non-metallic inclusions and different steel grades).

(ii) The training time is reduced and the training is stabilized.

(iii) The difficult and time consuming process of finding appropriate hyperparameters is accelerated significantly, since *one* set of parameters fits for *all* labels.

(iv) The regularized Wasserstein-distance was implemented to provide a fast and easy to use evaluation method which also makes the present algorithm easier to use for further investigations.

The use case presented shows the importance of describing the microstructure as accurately as possible statistically, which becomes possible with the CWGAN-GP. The RVE simulations show a pronounced influence of the different parameters (e.g. slope and size) of non-metallic inclusions on the stress accumulation and concentration in the microstructure. This is very important, because a high stress concentration in the affected areas leads to damage initiation and thus determines the overall performance of the material or the component.

In order to investigate stress concentrations and accumulation, as well as the origin of damage now even further and more precisely, the inclusions assumed to be rectangular and defined according to a fixed set of angles and sizes can be replaced by more realistic synthetic inclusions generated by the CWGAN-GP. The statistical representation of the remaining microstructure can also be improved using a CWGAN-GP. Furthermore, the damage can also be analyzed and processed in such a way that it can also be displayed in the CWGAN-GP. This would also allow the generation of RVEs with different damage states. The microstructure of the matrix as well as inclusions and potentially damage parameters could be mapped very accurately with one single CWGAN-GP.

7. Acknowledgement

This research was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – Projektnummer 278868966 – TRR 188. The authors also gratefully acknowledge the Forschungsvereinigung Stahlanwendung e. V. (FOSTA) and “Arbeitsgemeinschaft industrieller Forschungsvereinigungen e.V” (AiF) for providing financial support for the project “IGF.-Nr. 20112 N/FOSTA P1313”, which formed the basis for the investigations shown in this paper.

The support of this work by the Institute for Physical Metallurgy and Materials Physics (IMM) of RWTH Aachen and Carl Kusche is gratefully acknowledged, who provided the used EBSD pictures.

Simulations were performed with computing resources granted by RWTH Aachen University under project thes0880.

8. References

[1] Pütz F., Shen F., Könemann M., and Münstermann S. 2020 The differences of damage initiation and accumulation of DP steels: a numerical and experimental analysis *International Journal of Fracture* **226** 1-15

[2] Spriestersbach D., Grad P. and Kerscher E. 2014 Influence of different non-metallic inclusion types on the crack initiation in high-strength steels in the VHCF regime *International Journal of Fatigue* **64** 114-120

[3] Vajaragupta N., Wechsuwanmanee P., Lian J., Sharaf M., Münstermann S., Ma A., Hartmaier A. and Bleck W. 2014 The modeling scheme to evaluate the influence of microstructure features on microcrack formation of DP-steel: The artificial microstructure model and its application to predict the strain hardening behavior *Computational Materials Science* **94** 198-213

[4] Gitman I.M., Askes H. and Sluys L.J. 2007 Representative volume: Existence and size determination *Eng. Frac. Mech.* **74** 2518-34

[5] Henrich M., Pütz F. and Münstermann S. 2020 A Novel Approach to Discrete Representative Volume Element Automation and Generation-DRAGen *Materials (Basel, Switzerland)* **13** 1887

[6] Lewis A.C., Bingert, J.F., Rowenhorst D.J., Gupta A., Geltmacher A.B. and Spanos G. 2006 Two- and three-dimensional microstructural characterization of a super-austenitic stainless steel *Materials Science and Engineering: A* **418** 11-18

[7] Groeber M., Ghosh S., Uchic M.D. and Dimiduk D.M. 2008 A framework for automated analysis and simulationof 3D polycrystalline microstructures.Part 1: Statistical characterization *Acta Materialia* **56** 1257-73

[8] Groeber M., Ghosh S., Uchic M.D. and Dimiduk D.M. 2008 A framework for automated analysis and simulationof 3D polycrystalline microstructures.Part 2: Synthetic structure generation *Acta Materialia* **56** 1274-87

[9] Fátima Vaz M. and Fortes M.A. 1988 Grain size distribution: The lognormal and the gamma distribution functions *Scripta Metallurgica* **22** 35-40

[10] Pütz F., Henrich M., Fehlemann N., Roth A. and Münstermann M. 2020 Generating Input Data for Microstructure Modelling: A Deep Learning Approach Using Generative Adversarial Networks *Materials (Basel, Switzerland)* **13** 4236

[11] Goodfellow I., Bengio Y. and Courville A. *Deep Learning Book* 2016 *MIT Press.*

[12] Rumelhart D., Hinton G. and Williams R. 1986 Learning representations by back-propagating errors *Nature* **323** 533-536

[13] Goodfellow I., Pouget-Abadie J., Mirza M., Xu B., Warde-Farley D., Ozair S., Courville A. and Bengio Y. 2014 Generative adversarial nets *NIPS* **2** 2672-80

[14] Arjovsky M., Mirza M. and Bottou L. 2017 Wasserstein GAN *Proceedings of the 34th International Conference on Machine Learning* **70** 214-223

[15] Radford A., Metz L. and Chintala S. 2017 Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks *Preprint 1511.06434*

[16] Srivastava A., Valkov L., Russell C., Gutmann M. and Sutton C. 2017 VEEGAN: Reducing Mode Collapse in GANs using Implicit Variational Learning *Advances in Neural Information Processing Systems* **30** 3308-18

[17] Villani C. *Optimal Transport - Old and New* 2009 *Springer-Verlag* 338

[18] Guljarani I., Ahmed F., Arjovsky M., Dumoulin V. and Courville A. 2017 Improved Training of Wasserstein GANs *Advances in Neural Information Processing Systems* **30**

[19] Bachmann F., Hielscher R. and Schaeben H. Grain detection from 2d and 3d EBSD data—Specification ofthe MTEX algorithm 2011 *Ultramicroscopy* **111** 1720-33

[20] Schindelin J., Arganda-Carreras I., Frise E. et al. 2012 Fiji: an open-source platform for biological-image analysis *Nature methods* **9** 676-82

[21] Paszke A., Gross M., Massa F., Lerer A., Bradbury J., Chanan G., Killeen T., Lin Z., Gimelshein N., Antiga L., Desmaison A., Köpf A., Yang E., DeVito Z., Raison M., Tejani A., Chilamkurthy S., Steiner B., Fang L., Bai J. and Chinatla S. 2019 PyTorch: An Imperative Style, High-Performance Deep Learning Library *Advances in Neural Information Processing Systems* **32** 8024–35

[22] Peyrè G. and Cuturi M. 2019 Computational Optimal Transport *Foundations and Trends in machine learning* **11** 355-607

[23] Feydy J., Séjourné T., Vialard F., Amari S., Trouve A. and Peyré G. 2019 Interpolating between Optimal Transport and MMD using Sinkhorn Divergences *The 22nd International Conference on Artificial Intelligence and Statistics* 2681-90