Article Identifier
DOI: 10.22032/dbt.58849
URN: urn:nbn:de:gbv:ilm1-2023isc-139:9

60th ILMENAU SCIENTIFIC COLLOQUIUM
Technische Universität Ilmenau, 4 – 8 September 2023
URN: urn:nbn:de:gbv:ilm1-2023isc:1

Implicit Neural Representations for Deep Drawing and Joining Experiments

Aida Farahani, Payam Atoofi, Julien Vitay, Fred H. Hamker

Lab of Artificial Intelligence, Department of Computer Science, Chemnitz University of Technology

ABSTRACT

A deep understanding of metal deformation processes is essential for producing complex geometries in many industrial applications. Although simulations using Finite Element Methods (FEM) have helped in steering toward that goal, they are particularly time-consuming for large 3D meshes. Searching for the process parameters that lead to the desired shape of a metal part can become extremely expensive in terms of man-hours and computational resources. We investigated how machine learning models, especially deep neural networks, can help in speeding up the design process of deep drawing and joining processes by allowing a fast interpolation of FEM simulations from minutes or hours to seconds. In this study, inspired by implicit representations of 3D objects using neural networks, an implicit approach is used to predict local properties such as the thickness of the metal sheet, its thinning, and plastic strain, using solely the process parameters defining the experiment. We observe that the low number of trainable parameters of the predicting model ensures a generalization to unseen process parameters and ultimately allows for a reliable fast inspection of the processes.

Index Terms – FEM, Deep Neural Networks, Implicit Representations, 3D Meshes.

1. INTRODUCTION

Metal deformation refers to the process of changing the shape or size of a metal sheet through applied forces. It is a critical aspect of various industries, including manufacturing, automotive, aerospace, and construction. Thus, simulation solutions play a vital role in understanding and optimizing metal deformation processes. Current simulation techniques, such as FEM, enable engineers to predict and analyze the behavior of metals under different loading conditions. In addition to its power and widespread use, FEM also has several disadvantages. Firstly, FEM requires expertise and careful implementation due to its complexity and the various steps involved. For parameter tuning during each experiment, new FEM simulations must be performed. Furthermore, FEM becomes computationally demanding for high-dimensional problems and large meshes. Mesh sensitivity is another issue, as inaccurate or poor-quality meshes can lead to incorrect results. The convergence of FEM solutions can be affected by several factors, such as the type of elements used, the nature of the problem (e.g., non-linearities), and the numerical solution methods employed. Ensuring convergence and stability in all scenarios can be non-trivial and may require additional techniques, such as adaptive mesh refinement or stabilization methods.

In recent years, deep neural networks (DNNs) have received significant interest in various engineering fields such as solid mechanics, aiming to address the limitations of traditional methods like FEM [1, 2, 3]. In comparison to FEM approaches, DNNs may offer several

advantages for metal deformation tasks. Firstly, a DNN is capable of learning patterns and relationships from large amounts of data. Neural networks would use the learned statistical underlying information provided through the data in the training phase to predict an outcome for unseen data from the same distribution. By training on extensive datasets containing instances of metal deformation, DNNs can capture complex non-linearities and intricate relationships that may be challenging to model explicitly using FEM. This data-driven approach can lead to more accurate and robust predictions.

Neural networks in general can generalize well to unseen scenarios. Once trained on a dataset, they can interpolate and predict the behavior of metal deformation in situations that were not explicitly present in the training data. This ability to generalize can be particularly advantageous when dealing with novel or complex deformations. They also can be trained to incorporate sensor data, such as force measurements, into the deformation prediction process. This integration allows for real-time monitoring and control of metal deformation tasks, enabling applications such as online process optimization or quality control.

Despite the advantages mentioned above, integrating DNNs with mesh structures, which is the main data type for FEM solutions, can present challenges. DNNs often operate on fixed-size inputs, and maintaining a consistent mesh topology and size can be difficult or even impractical in some cases. To overcome these limitations, we propose a novel approach that combines DNN with implicit representations, eliminating the need for explicit meshes in deformation tasks. FEM simulations typically require discretization of the domain into elements, leading to a high computational cost, especially for complex geometries and large-scale problems. In contrast, once trained, our proposed DNN can provide rapid predictions for new inputs, significantly reducing the computational time, which may not be possible in FEM simulations. The use of implicit representations allows for smaller networks since they do not require the entire mesh information to be fed at once. This approach enables efficient processing of large meshes by operating on implicit functions rather than explicit mesh data. It offers a balance between memory requirements and computational efficiency while handling complex 3D models.

In Section 2, details about data acquisition are provided and the rationale for using implicit representations over explicit ones is presented. The section ends with the main structure of the models and the input/outputs of the network. In Section 3, the training process of each model is provided, and the results are presented.

2. METHODOLOGY

2.1 Deep Drawing and Joining Processes

Deep drawing is a metal-forming process that transforms flat sheets into three-dimensional shapes. In this process, blank metal sheets are placed over a die and punched to give them a desired shape. Material is stretched and bent to achieve the desired shape, resulting in deformed parts. The joining process refers to the act of attaching different parts to form a unified structure; and its goal is to create a strong and reliable connection between the components, ensuring structural integrity and functionality. Joining operations can result in incomplete or improper fusion of the components. This can result in gaps or voids between the joined surfaces, commonly referred to as “pockets”. Pockets can negatively impact the structural integrity, strength, and overall quality of the joined assembly. Therefore, minimizing or eliminating pockets is crucial for ensuring a reliable and robust joining process. Figure 1 shows a schematic illustration of both the deep drawing and joining processes. To minimize the packet size, the

resulting shapes are compared with the target shapes from the dataset in a supervised manner to reduce the error.

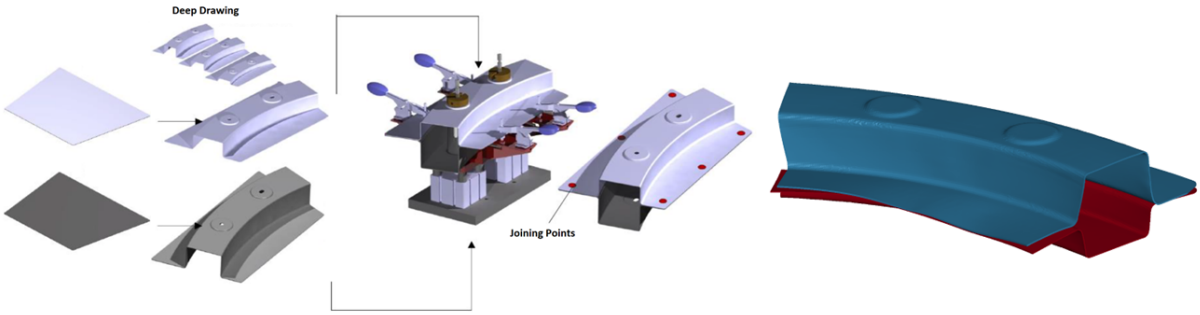

Figure 1. Schematic illustration of the deep drawing and joining processes. A metal sheet with specific features, such as thickness, alloy type, etc., will undergo a deep drawing process, with particular operation parameters, e.g., blank-holder force. Two of such formed objects are then used for the joining process to create the desired final object. A CAD model of the joining process is shown on the right. The ideal sequence of processes is one where the resulting object does not have wrinkles or pockets when joined (notice the contact area of the flat surfaces on the top and bottom objects are not completely aligned). It is worth noting that both sheet features and the operation parameters together are referred to as process parameters.

2.2 Data Acquisition / Preparation

During the BMBF project ML@Karoprod (01IS18055), we aimed to approximate the deep drawing and joining processes with functions, e.g., with neural networks, such that one could optimize the quality of the joining process to minimize the pocket size. Our partners at Fraunhofer IWU Dresden performed 725 successful simulations of deep drawing experiments, as well as 321 joining experiments. These simulations ranging from deep drawing to clamping and joining have been the fundamental aspect of mechanical engineering when it comes to optimization and parameter space exploration. Previous studies showed the importance of methodically exploring the design parameters to obtain a faster and more accurate production line [4]. However, the rigorous search and constant input from the expert renders such approaches laborious. Hence, finding an approximation of the process, a model with fast inference time, which regardless of the existence of its derivative as a function, can help in numerically solving an optimization problem.

The deep drawing simulations start from a blank metal sheet, where the following different process parameters can be varied: the sheet thickness (from 0.99 to 1.48 mm), the drawing depth (30, 50, or 70 mm), the drawing gap (1.6 or 2.4 mm), the blank-holder force (from 10 to 500 kN), the insertion position (-5 to +5 mm) and the material properties (RP0, from 133 to 293). Each simulation resulted in a triangular mesh with over 20,000 nodes, including for each mesh face a thickness value that allows computing the thinning and plastic strain. The joining experiments combined two of the 725 deep-drawn simulations, the upper part always having a drawing depth of 50 mm, the lower part either 30, 50, or 70 mm. The additional process parameters of the joining simulations are the Z-position of the four clamps and a clinch ID for each part. Thus, the process parameters in joining can be expressed by the process parameters of the two deep-drawn parts and the additional information of the joining operation itself.

The simulation generates meshes representing the surface of the object, which would pose difficulties in terms of computational cost as well as possible inconsistencies in terms of geometry. However, given the discrete nature of a mesh, if a continuous representation of the surface is provided by interpolation, one could develop a model on such a domain directly.

UV mapping, initially proposed for texture mapping [5] to bijectively map 3D points onto a 2D surface, has been used in our setting to represent the surface with its additional signal information: thickness after deformation, thinning, and plastic strain. Figure 2 depicts the UV mapping from the 3D surface to 2D space for each variable. In both sets of experiments, a custom UV mapper, developed by our partner Scale GmbH[1], has been employed.

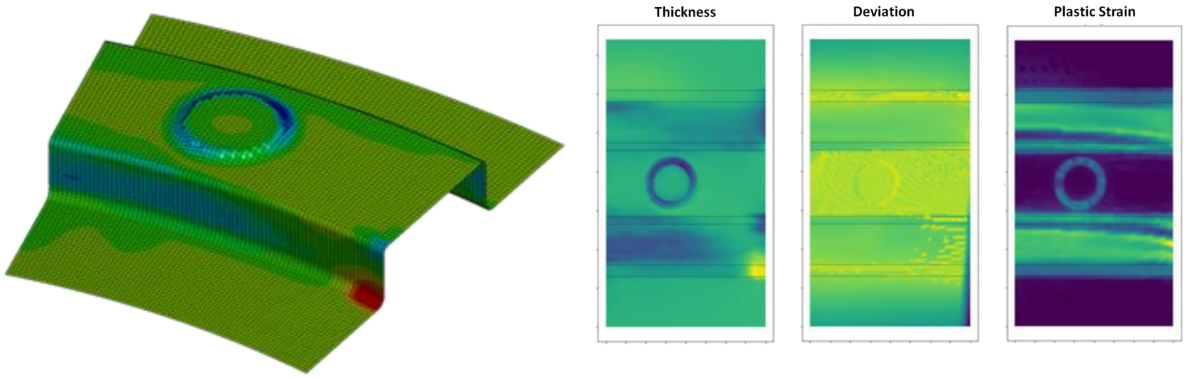

Figure 2. An illustration of UV mapping: 3D surface geometry of a reference mesh (the object on the left) is mapped onto a 2D space, where each mapping shows the distribution of the corresponding encoded signal, e.g., thickness, deviation from the desired shape (displacement), and plastic strain.

2.3 3D Implicit Representations

Implicit representations, such as Signed Distance Functions (SDF), are mathematical techniques used to represent geometric shapes without explicitly defining their boundaries or surfaces. SDF assigns a signed distance value to every point in space, indicating the distance to the nearest point on the surface of the object. This representation allows for a compact and flexible description of complex shapes. Implicit representations are particularly useful in computer graphics, computer-aided design, and simulations, as they enable efficient rendering, collision detection, and deformation calculations. They also provide advantages such as easy blending and interpolation between shapes and robust handling of topological changes.

The authors of DeepSDF [6] presented a deep network that embeds the shape of an object with implicit representation. The input to DeepSDF is a 3D point in Cartesian space, and the output is the signed distance from that point to the surface of the object. By training on a dataset of 3D shapes, DeepSDF can generate accurate and continuous representations of complex objects, allowing for various applications such as shape reconstruction, shape completion, and 3D shape generation.

By extending the DeepSDF idea and adding a condition vector to the network, we were able to predict the deformation of a metal cuboid influenced by a point force [7]. The applied force is added to the network input as well as XYZ coordinates of points, to predict the signed distance values of the deformed shape. By changing the input parameters (position of the force vector and its magnitude), the network could predict the deformed shape in a few seconds, which is much faster than FEM simulations on the same mesh. To our knowledge, this was the first time that a neural network could successfully estimate a shape deformation using an implicit representation in 3D space. Unfortunately, this method could not be directly applied to the deep drawing and joining data, as it needs to assume a watertight mesh as input. A watertight mesh is free from gaps, holes, or cracks, ensuring that no fluid or air can pass through it, thereby

[1] https://www.scale.eu/de/aktuelles/forschungsprojekte

dividing the space into inside and outside regions. For our metal shells, we cannot define such a signed distance. We decided to predict instead the attributes such as thickness, thinning, or plastic strain as if they were signed distances.

2.4 Models

In this section, the architecture of the neural networks learning implicit representations for deep drawing and joining processes is presented in more detail. An interesting aspect of the model is its low dimensionality in terms of having as few trainable parameters as possible, combined with the fully connected nature of each layer, which is especially useful when there is not an abundance of processing units such as GPUs. We show that such an architecture nevertheless has the expressive power to accurately approximate the simulation. Furthermore, despite the inherent difference between the deep drawing and joining processes, it is shown that our approach can simply model both processes, by changing the process parameters. Despite the similarity of the general structure of the designed models, the inputs/outputs are different.

Instead of treating each FEM simulation as a unique sample for training an explicit model, the implicit approach instead relies on learning to predict relevant quantities at each node of each simulation. Not only does this increase the number of training samples, but it also reduces the number of inputs, leading to very small neural networks that are less likely to overfit and are very fast to use. In our framework, for each process, two fully connected neural networks are used, each taking as an input the process parameters of a simulation, as well as the UV coordinates of a single node. The output of the first neural network predicts the XYZ position of each node, while the second predicts the corresponding thickness, thinning and plastic strain.

The rationale behind this approach depicted in Figure 3 is that the UV coordinates can be sampled uniformly between 0 and 1, while the XYZ coordinates would be specific to each experiment. As it was mentioned earlier, the sole difference between the models for the deep drawing process and the models for the joining process is the dimension of the process parameters that are given as part of the input. In the case of the deep drawing process, the input is the six process parameters of the simulation plus the UV coordinates. For a set of unseen process parameters at test time, one only needs to uniformly sample the UV space to recreate the geometry of the deep drawn part and its physical properties. The 725 deep drawing experiments are split into training and test sets, with 75 experiments used for validation.

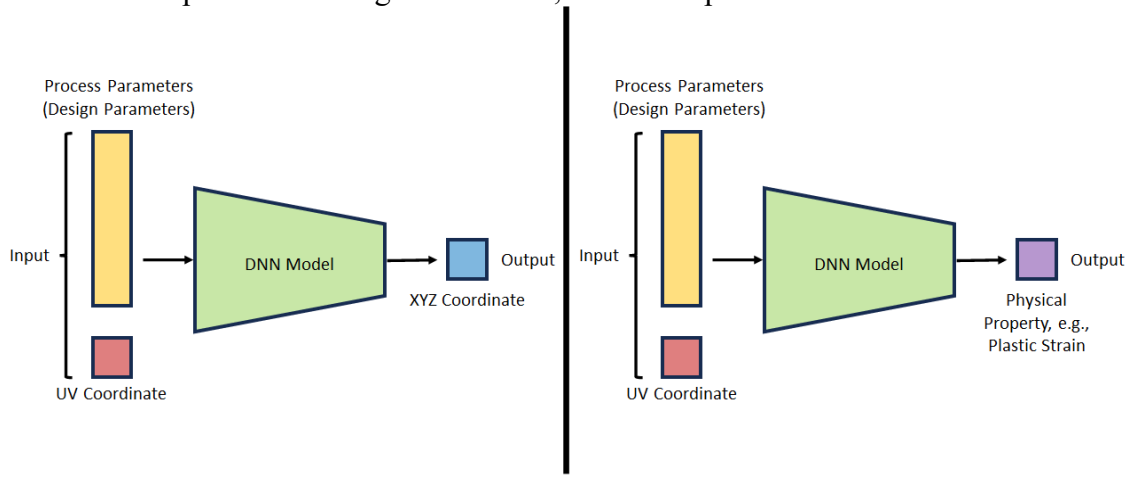

Figure 3. A schematic illustration of the models, and their inputs and outputs. Both models are given the process parameters (design parameters), e.g., blank-holder force, sheet thickness, etc., and a coordinate on the UV domain as input vector. The model on the left outputs XYZ coordinates of the geometry of the object after having undergone deformation, as a result of deep drawing, or after the joining process, corresponding to the UV coordinate given the reference mesh. The model on the right outputs the physical property of the position on the surface (XYZ position), such as thickness, thinning, and plastic strain.

Note that the only difference between the two models used in the deep drawing prediction vs the two models in the joining process prediction is the number of process parameters passed as part of the input.

For the clamping and joining process, there are 18 process parameters involved, 12 of which are the deep drawing process parameters of the two parts, six parameters for each. Of the remaining six parameters, four denote the Z-position of the four clamps and two indicate the clinch ID of each part. For the testing phase of the model, 32 experiments are kept out of the 321 simulations for validation. The upper and lower parts have their own UV coordinate systems. Two networks are trained as previously to predict the XYZ positions of each node of the two parts, as well as the physical properties such as thickness, thinning, and plastic strain.

The optimal architecture of the neural networks is found using Bayesian hyperparameter optimization [8], varying the number of layers, number of neurons in each layer, the learning rate, and regularization level, while using the test MSE as the objective to minimize.

3. RESULTS

3.1 Deep drawing

Given a reasonable search space for the hyperparameters of the neural network, the following hyperparameters have been shown to outperform in terms of MSE on the validation set, and as such are selected for inference: a neural network with five hidden layers (each with 256 neurons), LeakyRelu as activation function, the Adam regularizer, and a learning rate of 0.001. The same architecture was used for the two networks predicting the XYZ coordinates and the physical properties, respectively. The network is trained for 100 epochs and batch size 32768. Both training and validation losses are below 10^-5 at the end of training.

Figure 4 shows the output of the neural network for an unseen experiment of the validation set. The top row shows the predicted thickness, thinning, and plastic strain values in the UV space, while the bottom row maps these values onto the XYZ coordinates predicted by the first network. The residual quadratic error for all predicted quantities in the validation set is extremely low (below 1e-5), showing that the networks have learned to generalize the physical properties of the deep drawing experiments to unseen process parameters.

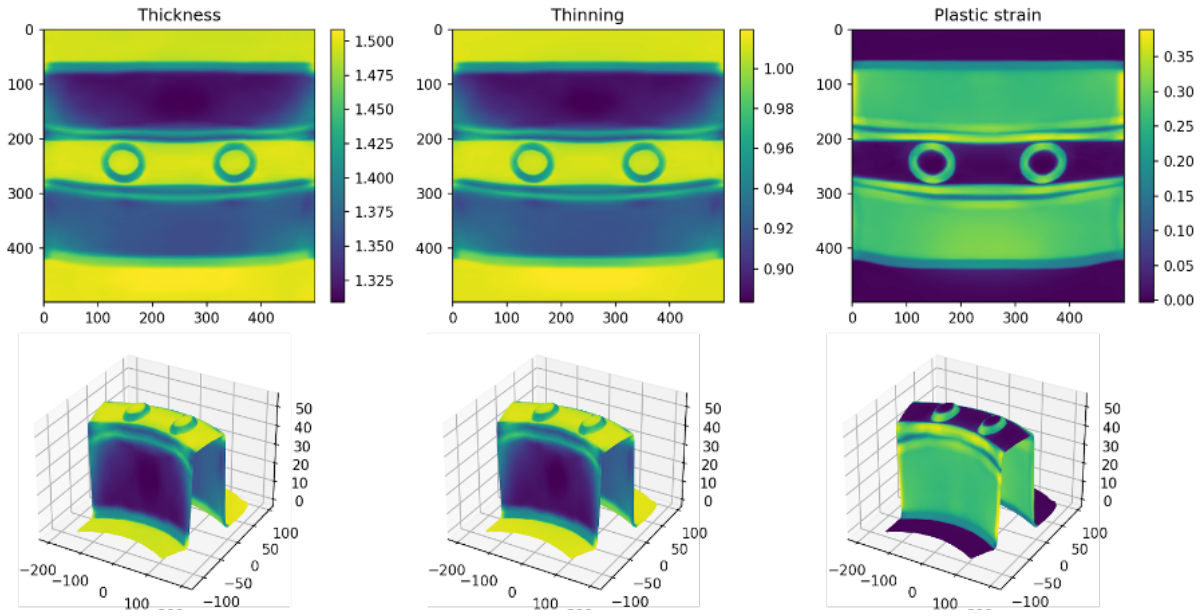

Figure 4. 2D prediction of thickness, thinning, and plastic strain for test experiment #883 (thickness 1.48, blank-holder force 20, drawing gap 2.4, insertion position 0, drawing depth 30, RP0 235.160326). Above: predicted output value in the UV space. Below: mapped onto the predicted XYZ-coordinates.

3.2 Joining

For the joining process, the following hyperparameters have been also found to perform best within the search space: a neural network with five hidden layers (each with 256 neurons), LeakyRelu as an activation function, the Adam regularizer, and a learning rate of 0.001. Similarly, the goal was to minimize the mean square error loss on unseen data. The network is trained for 50 epochs. The training and validation losses are both below 10^-5 after training. As for the deep drawing models, the same architecture is used for both models predicting the geometry and the physical properties.

Figure 5 shows the prediction of the two networks for a completely new design (the bottom and top parts were never clamped together in the training data). Similarly, the physical properties of these two parts after the joining process are predicted by the model in a few seconds, which would have otherwise taken at least two orders of magnitude more time in the simulation. In Figure 6, the cross-section of the joined objects is provided. These cross-sections are given by changing only the values in one of the two dimensions of the UV map, while the values on the other dimension remain unaltered. The pockets can be observed in the cross-section.

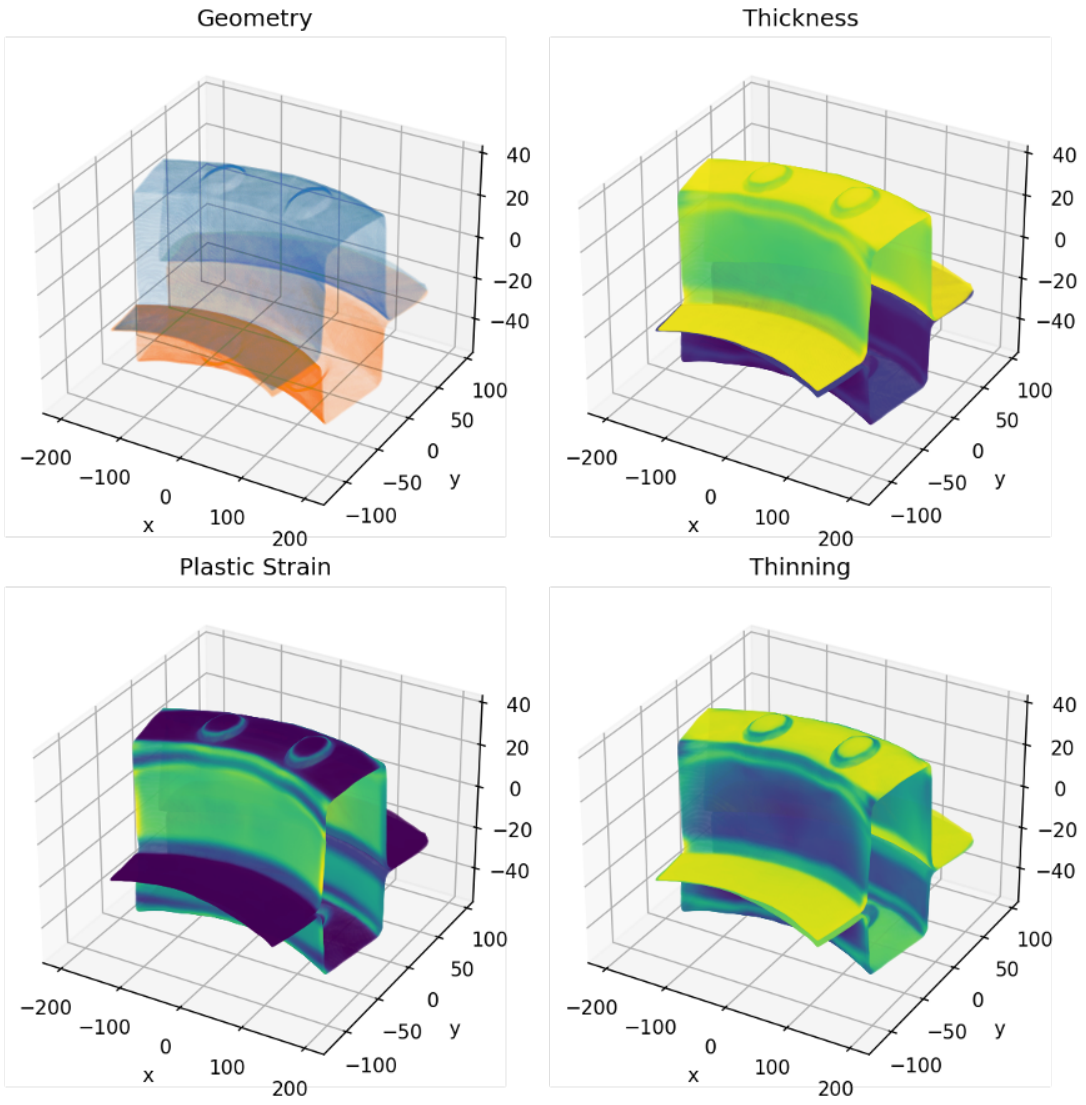

Figure 5. Prediction for a test experiment: clamping position [-3, -3, -5, 3], MID [5, 3], upper part (thickness 1.48, blank-holder force 400, drawing gap 2.4, insertion position -4, drawing depth 50, RP0 235.160326), lower part (thickness 1.01, blank-holder force 410, drawing gap 2.4, insertion position -5, drawing depth 30, RP0 138.226960).

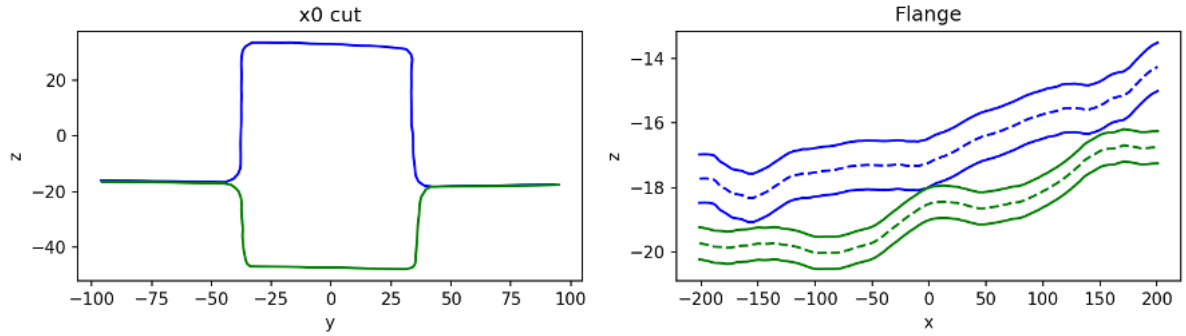

Figure 6. Cross-sections of the object after the joining process are shown. One could, using such a model, set an attainable goal to reduce the pockets' sizes in order for the contact area to have no gaps, i.e. a much faster model for an optimization problem where the optimal values for the process parameters are searched.

Figure 7 shows the exploration and inspection of the process parameters domain. The model can be used to interactively inspect the influence of the process parameters. The code underlying these simulations is freely available in GitHub[2].

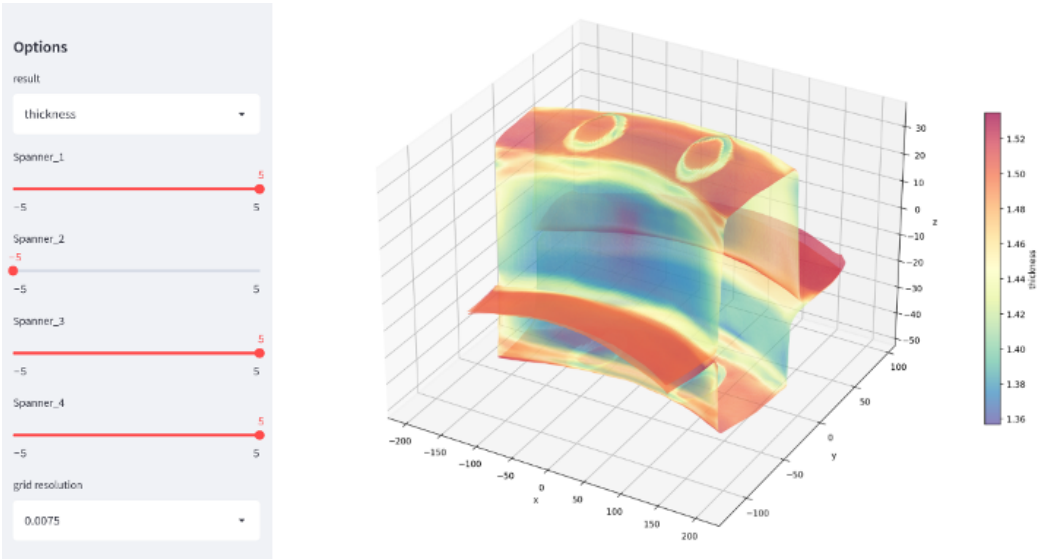

Figure 7. Using the menu on the left, containing the process parameters, as well as the expected information on the output, e.g., distribution of the plastic strain or thickness across the object, one could interactively and significantly faster explore and inspect the impact of a given set of process parameters. After the training stage of the model, see Figure 3, given the UV mapping of a reference mesh, in the inference stage, different points from UV space are sampled and their corresponding XYZ coordinates are predicted by the model, which results in the geometry shown here. For information on the surface, for instance thickness, the corresponding values are predicted for each XYZ position, which is depicted as a heatmap.

4. CONCLUSION

In this study, we examined the advantages and limitations of Deep Neural Networks and Finite Element Methods for metal deformation analysis, primarily attributed to their reliance on mesh structures. The main advantage of our proposed approach is the ability to handle irregular and unstructured data for metal deformation tasks and joining process optimization. By using DNNs with implicit representations, it becomes possible to directly learn the underlying relationships

[2] https://github.com/hamkerlab/ML-Karoprod-MeshPredictor

between input data and desired outputs without explicitly relying on a mesh structure. This allows for more flexibility in handling complex geometries and varying mesh densities.

Traditional FEM often requires manual adjustments and refinements of the mesh to account for changes in geometry or material properties. In contrast, DNNs can learn from a dataset of instances and potentially generalize well to new scenarios without explicitly re-meshing. This can save costs, both in terms of time and energy, in the modeling and simulation process, e.g., using such machine learning models, an engineer can quickly investigate the influence of various process parameters without having to run expensive FEM simulations. Our proposed model has a small number of trainable parameters, is capable of generalization, and is significantly faster in the inference phase than FEM simulations.

However, it is important to note that there are also challenges associated with using NNs. One challenge is the need for large and diverse training datasets to capture the wide range of possible deformations accurately. Generating such datasets can be time-consuming and computationally expensive. Additionally, the interpretability of the models may be limited compared to traditional methods like FEM, where the mesh provides a clear representation of the physical domain.

In conclusion, this paper presented a methodology for predicting deformation and thickness resulting from a single applied force on a metal sheet. However, to further enhance the applicability of this research in the future, it is suggested to extend the study to include a collection of different force vectors that are sequentially applied to the metal sheet, allowing for the achievement of desired shapes. By optimizing and fine-tuning the set of process parameters through a series of actions, this approach holds great potential for numerous industrial applications. Furthermore, the independence of our approach from mesh structure enables generalization across a wide range of sizes and structures. To accomplish this, a deep reinforcement learning scheme is proposed as a promising avenue for future investigation.

Acknowledgement

This work has been funded by the Federal Ministry of Education and Research (BMBF) - ML@Karoprod (01IS18055).

REFERENCES

[1] Grossmann, Tamara G., et al. "Can physics-informed neural networks beat the finite element method?." *arXiv preprint arXiv:2302.04107* (2023).

[2] Rezaei, Shahed, et al. "A mixed formulation for physics-informed neural networks as a potential solver for engineering problems in heterogeneous domains: comparison with finite element method." *Computer Methods in Applied Mechanics and Engineering* 401 (2022): 115616.

[3] Vurtur Badarinath, Poojitha, Maria Chierichetti, and Fatemeh Davoudi Kakhki. "A machine learning approach as a surrogate for a finite element analysis: Status of research and application to one dimensional systems." *Sensors* 21.5 (2021): 1654.

[4] Kriechenbauer, Sebastian, and Dirk Landgrebe. "Methode zur Auslegung von Sonderverfahren in der Blechumformung mit hoher Prozessdynamik am Beispiel der Kissen-Stößel-Pulsation." (2016).

[5] Sheffer, Alla, Emil Praun, and Kenneth Rose. "Mesh parameterization methods and their applications." *Foundations and Trends® in Computer Graphics and Vision* 2.2 (2007): 105-171.

[6] Park, Jeong Joon, et al. "Deepsdf: Learning continuous signed distance functions for shape representation." *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019.

[7] Farahani, Aida, Julien Vitay, and Fred H. Hamker. "Deep Neural Networks for Geometric Shape Deformation." *KI 2022: Advances in Artificial Intelligence: 45th German*

[8] Snoek, Jasper, Hugo Larochelle, and Ryan P. Adams. "Practical Bayesian optimization of machine learning algorithms." *Advances in neural information processing systems* 25 (2012).

CONTACTS

Aida Farahani

email: aida.farahani@informatik.tu-chemnitz.de
ORCID: https://orcid.org/0000-0002-1758-0924