Parametric Shape Optimization of Stretch Webs in a Progressive Die Process using a Neural Network Surrogate Model

S Athreya¹, A Weinschenk¹, F Steinlehner², D Budnick³, M Worswick³, W. Volk² and S. Huhn¹

¹ Forming Technologies, Hexagon Manufacturing Intelligence, Oakville, Canada
² Chair of Metal Forming and Casting, Technical University of Munich, Garching, Germany
³ University of Waterloo, Waterloo, Canada
* Corresponding Author: srinivas.athreya@hexagon.com

Abstract. Progressive die stamping provides a solution for producing sheet metal parts in large quantities. These parts are connected to carriers by stretch webs. As the part undergoes bending and forming operations, the stretch webs are exposed to translational and rotational deformation. A suitable design of these entities is crucial to avoid failure caused by splits or excessive thinning. A common way to evaluate such designs is to use finite element (FEM) simulation. Since it is not efficient to run FEM based optimization studies for the design optimization and to enable further automation of the stretch web design, this paper is proposing the use of machine learning (ML) technologies. A surrogate model based on an artificial neural network is used as a predictor in the presented study. This neural network is used to optimize the geometric parameters of the stretch web to obtain a quality result. The model is trained using FEM results and the study shows that it was possible to obtain an accurate model with a prediction error of 5%. The trained surrogate model can be used for the optimization study. This approach is computationally inexpensive and can provide very good results.

1. Introduction

Progressive die stamping is commonly used for manufacturing of large quantities of small and medium sized sheet metal parts. It is a fast and efficient process as it reduces the required blank handling steps compared to a conventional line die process. Additionally, it allows a high stroke rate because the formed part is connected to a carrier and can easily be carried from one operation to the next. Based on the coil width, the progressive nesting layout is defined. It consists of one or multiple carriers, blanks, connections, and scrap. These connections are also known as stretch webs. The progressive die process is divided into a 2D process consisting of piercing and cutting of scrap and a 3D process consisting of forming, drawing, restrike, bending and flanging operations followed by trim operations after which the final part is released. This demonstrates that stretch webs undergo multiple operations which result in multi-dimensional translational and rotational motions. They therefore need to fulfil certain requirements to avoid failure during the process. Stretch webs need to connect the part with the carrier, take the draw depth of the part into account by allowing sufficient stretching and need to provide enough stiffness to ensure progression of the carrier. Depending on the required boundary conditions, different
stretch web shapes, for example I-, O-, S-, U-, L-shapes, are used. This presents a challenge for the engineer who needs to have a good understanding of the existing load cases in order to choose a suitable web shape. The goal is to select a suited shape which doesn’t fail during the entire process. The selection of the web shape is mainly experience-driven and was studied in more depth for the first time in [1]. This study can be used as a guideline to help the engineer to determine which shape is suitable for the respective load cases. Each shape has several geometric parameters which influence the strain condition as well. The study conducted in [2] shows the sensitivity of these geometric parameters by comparing resulting force-displacement curves. The selection still has to be made by the engineer and iterations are often required.

FormingSuite [3] is a software developed by Forming Technologies, for sheet metal process design, cost estimation and simulation. It allows the user to set up the entire progressive die process and has a separate workbench for stretch web definition and simulation. The objective of this study is to build and implement an optimization system in FormingSuite that will suggest an optimum design for the stretch web based on the initial design by the user.

2. Approach

The industry has standard parametric shapes that are commonly used for the design of stretch webs. We shall discuss these shapes in detail in Section 3. To achieve the goal, we have developed an optimization solution which will attempt to maximize the quality of the stretch web by choosing the best combination of geometric parameters. The first step in building an optimization model is to develop a metric for quantifying the quality of our stretch web design. The stretch web designs are evaluated on a Finite Element Method (FEM) model and a quality score for the design is obtained. The methodology of obtaining the quality score is explained in Section 4. Our optimization model neglects the effect of other stretch webs in the forming process. We shall also exclude the effect of vibrations in the progressive die process for the sake of simplicity. Since FEM is slow and inefficient for an optimization study in which a large number of iterations need to be evaluated in real time, we shall use a surrogate modelling approach. For generating our surrogate model, we have generated 1646 experiments with our existing FEM solver. A comparison study was completed (see Section 6) and it was decided that an Artificial Neural Network Model (ANN) would be the best model for our surrogate for its simplicity and accuracy. Our optimization will be performed over the ANN model using the gradient ascent algorithm which will be discussed in Section 5.

3. Stretch Webs

![Figure 1. a) Geometric Parameters of the oval web and b) Angle of Connection](image-url)
Parametric stretch web shapes [2] which are supported by the FormingSuite software are the solid web, oval web (Figure 1) and the S web. Some advanced users also use a sketch web tool, which allows the user to define their own free form shape. In this study we will focus on parametric shapes for their simplicity and to align with our objective to assist novice users to design stretch webs faster. In this paper, we have used oval web data to represent our surrogate model. The methodology of generating the surrogate model can be extended to other parametric shapes as well.

4. The Finite Element Model
The FEM stretch web solver, developed by Forming Technologies, is used to evaluate the quality of the stretch web design. The stretch web material is assumed to be isotropic and homogeneous. The FEM solver will evaluate the stresses and strains of the stretch web design using the blank deformation as the input. The major and minor strain is then calculated for each node and the safety zones are evaluated based on the forming limit diagram. The figure 2 shown below is an example of the forming limit diagram obtained from forming suite. The engineering minor strain and the engineering major strain for each node in the model has been plotted on the X and Y axes respectively. The forming safety zones are assigned based on the regions obtained in the forming limit diagram as shown. A more detailed explanation on the forming limit diagram can be found in [4].

![Figure 2. An example of the forming limit diagram obtained from FormingSuite](image)

Once the forming zones are evaluated for each node, if the stretch web design has at least one node that lies in the excessive thinning, shear failure or split safety zones, the quality score is set to 0. The quality score is set to 1 in all other cases. Since the quality score output is binary, we can use a classification model for our surrogate. The current model does not include sheared edge formability and it will be considered in future work.

5. Optimization
Our optimization problem is to maximize the quality score obtained from the FEM solution by choosing the best combination of geometric parameters for the stretch web. Figure 3 illustrates our optimization problem.
Figure 3. The optimization problem

Since we do not have an analytical expression that equates the geometric parameters to the quality score output, we need to consider derivative free optimization techniques. The derivative free optimization techniques can be broadly classified into direct methods and model-based methods [5]. We are going to consider model-based optimization since it won’t be computationally feasible to use our Finite Element solution for an optimization problem.

5.1. Surrogate Model

Surrogate modelling is an optimization technique which is used in cases where the objective function cannot be evaluated easily. Due to the computation cost associated with the finite element model, it is necessary to construct a surrogate model that can replicate the results in real time. The surrogate model can also be used as a predictor which can give immediate feedback on the shape during the design process. Since our quality score is binary, we can use a classification surrogate model. Commonly used classification models are the logistic regression model, support vector classifiers, artificial neural networks, random forests etc. We have completed a comparison study on Logistic Regression, Linear Support Vector Machines [7], Radial basis Function Support Vector Machines [7], Artificial Neural Networks and Random Forests [8]. After doing a comparison study, it was decided that due to the accuracy (see Section 6) and its simplicity, Artificial Neural Networks (ANN) with the architecture, illustrated in Figure 4, would be our surrogate model. The ANN used in our study consists of one hidden layer, and 12 neurons in the hidden layer. The linear activation function is used for the hidden layer and a sigmoidal activation function is used for the final output layer. The sigmoidal activation function is defined in equation 1.

Figure 4. A representation of our surrogate model
The output $y$ of this network is given by equation (2)

$$ S(x) = \frac{e^x}{e^x + 1} $$. \hspace{1cm} (1)

$$ y = S(\sum_{j=1}^{m} (w_j \sum_{i=1}^{n} w_{ij} x_i)) $$ \hspace{1cm} (2)

Here, $w_j$ and $w_{ij}$ are weights associated with the hidden layer and output layer respectively. The surrogate model was trained by extracting the data from the FEM model described in Section 5. The inputs used for generating the surrogate model are width, web width, oval width, hole width, angle, boundary, angle, displacement boundary conditions, area of the stretch web and material properties. Material properties include yield stress, Young’s modulus, strain hardening, Lankford’s coefficient, thickness, and Poisson’s ratio. These properties are used to plot the forming limit diagram (Figure 2). We decided to exclude Poisson’s ratio from our model after considering the feature dependency plot shown in Figure 5. While Poisson’s ratio is essential for plotting the forming limit diagram, its value shows limited variability for most metallic materials used in the industry.

![Figure 5. Feature Dependency](image)

The quality score obtained from the FEM model is used as the output. The input and output relationship were extracted from the FEM model for 1646 examples using random sampling over a uniform distribution. For machine learning models to work effectively, it is necessary to center the input data around 0 to avoid the numerically larger inputs dominating the learning model. The data scaling is described using the equation (3). Here $x_i$ is the input, $\mu$ is the mean of the input feature and $\sigma^2$ is the standard deviation of the input feature.

$$ x_i = \frac{x_i - \mu}{\sigma^2} $$ \hspace{1cm} (3)

The input/output data is used to train the ANN model so that it can reproduce the FEM results. Using the ANN model, we were able to predict the FEM results with an error rate of 5%.
5.2. Gradient Ascent Optimization
The trained ANN surrogate model is differentiable, and we can optimize the input parameters using the gradient ascent optimization method. While different approaches have been proposed by [9] and [10], we have decided to go with the gradient ascent optimization to find the local extrema since we do not aim to make huge deviations from the existing design. Unlike the standard gradient descent method which is used to optimize the weights by minimizing the loss function, our objective is to maximize the quality score by modifying the input data. To obtain the updated inputs, we partially differentiate the output \( y \) with respect to the input data. \( w_{b0} \) and \( w_{b1} \) are the bias neurons associated with the input and hidden layer respectively.

\[
\frac{\partial y}{\partial x_i} = S \left( \sum_{j=1}^{m} w_j \left( \sum_{i=1}^{n} w_{ij} x_i + w_{b0} \right) + w_{b1} \right) \\
\left[ 1 - S \left( \sum_{j=1}^{m} w_j \left( \sum_{i=1}^{n} w_{ij} x_i + w_{b0} \right) + w_{b1} \right) \right] \sum_{j=1}^{m} w_j w_{ij}
\]

Equation (4) simplifies to the following equation.

\[
\frac{\partial y}{\partial x_i} = y(1-y) \sum_{j=1}^{m} w_j w_{ij}
\]

(5)

The inputs are then iteratively updated using the following equation.

\[
x_i = x_i + \alpha \frac{\partial y}{\partial x_i}
\]

(6)

Here \( \alpha \) is the learning rate which controls the rate of convergence of the solution.

6. Evaluation and Results
For generating the surrogate model, we obtained experimental data from 29 different stretch web geometries. Material properties were assigned randomly from the material database defined in FormingSuite. Geometric properties were setup with uniform distribution with lower and upper bounds that varied to avoid intersection with the previous blank in the progressive die.

| Table 1. Geometric parameter bounds for generating experiments |
|---------------------------------------------------------------|
| Width (mm) | Web Width (mm) | Hole Width (mm) | Oval Width (mm) | Angle |
| Lower Bound | 2 | 18 | 2 | 2 | -30° |
| Upper Bound | 25 | 50 | 15 | 8 | 43° |

To choose the surrogate model we considered accuracy and simplicity of implementation as our two major evaluation criteria. We have compared the performance of linear support vector classifier (Linear SVC), radial basis function support vector classifier (RBF SVC), artificial neural networks (ANN) and random forests on our data using SKLearn [11]. The accuracy of the learning models is also subject to various other factors such as the initial state of the models and the optimization of hyper-parameters. We have not explored these factors in detail since that is beyond the scope of our study. Our motivation here is to understand the best fit for our model using default settings and then optimize the chosen model to improve its accuracy. The results of the surrogate model accuracy are tabulated in Table (2). These
results were generated by randomly shuffling our dataset and choosing 80% of the data for training the model and 20% of the data for testing the model. The accuracy is calculated by Equation (7).

\[
\text{Accuracy} = \frac{\text{True Predictions}}{\text{True Predictions} + \text{False Predictions}}
\]  

(7)

Table 2. Prediction Accuracy for Different Classifiers

|                | Logistic Regression | Linear SVC | RBF SVC | Random Forests | Neural Networks |
|----------------|---------------------|------------|---------|----------------|----------------|
| **Shuffle 1**  | 87.5%               | 88.0%      | 91.8%   | 92.5%          | 92.9%          |
| **Shuffle 2**  | 88.7%               | 90.9%      | 92.5%   | 93.9%          | 94.2%          |
| **Shuffle 3**  | 88.3%               | 90.9%      | 93.8%   | 94.6%          | 95.0%          |
| **Shuffle 4**  | 87.6%               | 89.2%      | 95.0%   | 95.9%          | 95.0%          |
| **Shuffle 5**  | 89.6%               | 90.4%      | 95.9%   | 96.3%          | 96.7%          |
| **Mean Accuracy** | 88.3%   | 89.9%      | 93.8%   | 94.6%          | 94.8%          |

The results from Table 2 show that non-linear classifiers give us better results. Random forests and ANN surrogate models gave us the best results and the difference in the error rate can be associated with randomness associated with such a study. Table 3 shows the confusion matrix on the validation set used to validate our ANN model. The confusion matrix is a better indicator of the performance of a classifier model. It also provides a better illustration on positive and negatively predicted values. We used 221 examples to validate our surrogate model. From Table 3 we can observe that 2 configurations were misclassified as positive and 7 examples were misclassified as negative.

Table 3. Confusion Matrix plotted on the Validation Set for the ANN surrogate model

|                | Predicted Negative | Predicted Positive |
|----------------|--------------------|--------------------|
| **True Negative** | 155                | 7                  |
| **True Positive**   | 2                  | 57                 |

The implementation of the surrogate model in FormingSuite is done using MLPack [12] due to its compatibility with C++.

7. Concluding Remarks
We successfully generated the surrogate model using artificial neural networks. The computation time for evaluating the results improved significantly from 35-40 secs for the FEM model to under 2 secs for the surrogate model. The accuracy of our model averaged to 94.8% over all the datasets. The accuracy
of other nonlinear learning models was also > 92%, which indicates that the chosen features accurately represent our learning model. The optimization model using gradient ascent is also presented in this study. While there is some work to be done, especially with the modelling of geometric constraints in the optimization model, we believe that this optimization model has the potential to provide fast and reliable results. Our surrogate model can also be used as an indicator which can guide the user to build safe stretch webs by reducing the time interval for manual iterations. This study uses the oval web as an example, but our future work will also include other parametric shapes.

8. Acknowledgements
Authors acknowledge financial support of the National Research Council of Canada Industrial Research Assistance Program (NRC IRAP), the German Federal Ministry of Education and Research (BMBF) and the German Aerospace Center (DLR).

9. References
[1] Böttcher O, Gruber M, Raupach M and Volk W 2016 Anbindungsoptimierung von Folgeverbundbauteilen EFB-Forschungsbericht Nr. 449
[2] Steinlehner F, Weinschenk A, Kolb S, Laumann S and Volk W 2022 Numerical description of the physical properties of stretch web connectors in Progressive Die Stamping Numisheet 2022 (under review)
[3] Forming Technologies FormingSuite 2021.0.1 [computer software] Available at: https://www.forming.com/support/software-downloads/formingsuite
[4] Keeler S 1978 Forming Limit Criteria - Sheets Advances in Deformation Processing ed. J. Burke and V Weiss (Springer) chapter 4 pp 127-157
[5] Rios L and Sahinidis N 2013 Derivative-free optimization: a review of algorithms and comparison of software implementations J Glob Optim 56 1247-1293
[6] Myers R, Montgomery D and Anderson-Cook C 2016 Response Surface Methodology: Process and Product Optimization Using Designed Experiments (Hoboken: Wiley – Probability and Statistics)
[7] Cortes C and Vladimir Vapnik 1995 Support-Vector Networks Machine Learning 20 273-297
[8] Ho T 1995 Random Decision Forests Proceedings of 3rd International Conference on Document Analysis and Recognition pp 278-282
[9] Anjum M, Tasadduq I and Al-Sultan K 1995 Response surface methodology: A neural network approach European Journal of Operational Research 101 65-73
[10] Balkin S, Lin D 1999 A neural network approach to response surface methodology Communications in Statistics - Theory and Methods 29 2215-2227
[11] Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Passos A, Cournapeau D, Brucher M, Perrot M, Duchesnay E 2011 Scikit-learn: Machine Learning in Python Journal of Machine Learning Research 2825-2830
[12] Curtin R , Edel M , Lozhnikov M, Mentekidis Y , Ghaisas S and Zhang S 2018 mlpack 3: a fast, flexible machine learning library Journal of Open Source Software 3 726