The International Journal of Advanced Manufacturing Technology (2024) 133:4923–4938
https://doi.org/10.1007/s00170-024-14023-7

ORIGINAL ARTICLE

Prediction of forming accuracy in incremental sheet forming using artificial neural networks on local surface representations

Dennis Möllensiep[1] · Lukas Detering[1] · Philipp Kulessa[1] · Matthias Steinhof[1] · Bernd Kuhlenkötter[1]

Received: 11 March 2024 / Accepted: 12 June 2024 / Published online: 27 June 2024

Abstract

While incremental sheet metal forming offers the potential for producing sheet metal parts in small lot sizes, the relatively low forming accuracy prevents widespread industrial use. For improving the forming accuracy, research institutes are using machine learning techniques to predict the geometric accuracy and modify the toolpath based on the prediction. A critical challenge is it to ensure the generalizability of the prediction model as only a small amount of process data is available to train the model due to the lack of industrial collaborations. This publication presents a highly transferable feature engineering approach where surface representations of the part's geometry around each toolpath point are transferred into a standardized coordinate system. Several artificial neural networks were trained and used for predicting the forming accuracy and modifying the toolpath. During the validation experiments, the forming errors of parts which were independent of the training process were reduced by up to 68.5 %. The framework for computing the surface representations alongside with several pre-trained artificial neural networks is publicity available for download.

Keywords Artificial neural network · Machine learning · Incremental sheet forming · Roboforming

Abbreviations

ANN	Artificial neural network
AutoML	Automatic machine learning
DPIF	Duplex incremental forming
DSIF	Double sided incremental forming
FDB	Full database
FEA	Finite element analysis
ISF	Incremental sheet forming
LPS	Chair of production systems
ML	Machine learning
ROI	Region of interest
SPIF	Single point incremental forming
TCP	Tool center point
TPIF	Two point incremental forming

Nomenclature

i	Point counter
j	Initial value
n	Number of points
P	Prediction
s	Smoothed

✉ Dennis Möllensiep
moellensiep@lps.rub.de

Lukas Detering
detering@lps.rub.de

Philipp Kulessa
kulessa@lps.rub.de

Matthias Steinhof
matthias.steinhof@rub.de

Bernd Kuhlenkötter
kuhlenkoetter@lps.rub.de

[1] Chair of Production Systems, Ruhr-Universität Bochum, Industriestraße 38c, Bochum 44894, Germany

1 Introduction

Forming sheet metal components without the need for part-specific tooling can be done by incremental sheet forming (ISF). It is an ideal process for producing sheet metal components in small batches, such as required in prototyping, because the part is shaped by a hemispherical tool and the driven toolpath, which allows for easy modifications of the created geometry. However, a broad industrial use is hindered by the process's limited geometrical accuracy. To reduce the forming error by predicting it and modifying the toolpath according to the expected error, a lot of research focuses on finite element analysis (FEA) [1]. Despite the extensive

research by numerous research institutes, FEA is currently not capable of simulating the ISF process with sufficient accuracy while also exhibiting the disadvantage of a time-consuming modeling and computing phase. The main cause for the low simulation accuracy is the various process parameters and their complex nonlinear relationships with each other [2]. Recently, researchers have focusing on the application of machine learning (ML) techniques for predicting several process-specific dimensions [3]. These include forming accuracy [4], surface quality [5], tool load [6], forming temperature [7], the pillow effect [8] and the material flow curve [9]. Due to the lack of industrial ISF production lines, the data used for training the ML models has to be gathered by the research institutes themselves. This limits the amount of available data and therefore the transferability of the ML models. To enable generalizability, we present a feature engineering approach where the local geometry around each toolpath point is transformed into a standardized coordinate system and mathematical surface representation. An overview of the individual steps of the approach is given in Fig. 1. Afterward, the surface representations were used to train artificial neural networks (ANN) which exhibit a good prediction performance on independent parts. The prediction of the forming error was used to adjust the toolpath and decrease the forming error accordingly. To enable other researchers, the application of the approach, the trained ANNs, and the framework capable of computing the surface representations alongside various other functionalities is published for download [10].

2 Machine learning in ISF

The recent artificial intelligence hype led to numerous novel applications in almost all scientific fields. In ISF, researchers used ML algorithms for predicting various aspects of the forming process such as the forming accuracy [11]. Although, Nagargoje et al. [3] and Harfoush et al. [12] independently of each other concluded that more process data is a necessity for increasing the prediction accuracy of the ML approaches, while reviewing the state-of-the-art. Most publications used only a handful of geometries with slight variations for establishing a process database because of the experimental expenditure. Therefore, a portion of the authors used a cluster analysis approach to build up a database of 70 individual and diversified forming experiments (see section 3.1) [13] that is the foundation for ML training in this publication. While this is currently the largest ISF database it still only contains a fraction of data compared to datasets of real-world production lines with thousands of parts [14] or the extensively labeled datasets for self-driving vehicles [15]. Therefore, it is crucial to use feature engineering for gathering the maximum amount of information from the limited available process data. While it seems intuitive to rasterize the part geometry two-dimensionally with the height of the grid field as input for the artificial neural network analog to the brightness of the individual pixels when processing pictures, this would cause several problems. To achieve a precise prediction for the whole part, the individual grid fields need to have dimensions of distinctively under a millimeter what would result in a high resolution, even for small parts. In combination with the small amount of data, this would lead to the curse of dimensionality [16]. Therefore, a more efficient mathematical description of the parts with a lower dimensionality than the rasterization is needed. Typically, non-uniform rational B-splines are used in CAD for describing part geometries. These have the disadvantage that their non-linearity results in long computation times and complex implementations when used in ML [17]. One technique used in ISF is the dissection of the part into individual geometry features such as radii or planar surfaces [18]. Behera et al. combined geometry feature detection algorithms with multivariate adaptive regression splines for toolpath optimization [19]. Although, the use of geometry features limits the transferability of the approach as free-form surfaces can not be properly categorized. In their studies, they concluded that the forming error is strongly dependent on the geometry features close by and therefore the local geometry [20]. Forces and torques which are induced by forming are absorbed by the sheet surrounding the tool and have therefore only a neglectable influence on distant sheet areas. Therefore, the amount of available process data could be increased by dividing the part into different fields and rasterizing these afterward. This yields more individual data sets while also decreasing the necessary resolution of the rasterization which counteracts the curse of dimensionality. Khan et al. built up on this idea and introduced local geometry matrices for mathematically describing the local geometry [21]. El Salhi et al. expanded the concept by only considering a few points on the surface and weighting their influence according to the distance [22]. However, using a fixed rasterization still has several disadvantages. The prediction itself is only valid for the center of the grid field. Furthermore, for improving the forming accuracy the CAD file needs to be altered according to the prediction. It is mathematically challenging to modify the part in a way that it is still possible to calculate a valid toolpath. To solve this problem, we present an approach where the local geometry around every toolpath point is transformed into a standardized coordinate system. The description of the local geometry resembles the model input and the forming error of the toolpath points in the normal direction of the model output. Therefore, the toolpath can be directly adjusted while also ensuring a good training performance due to the standardized coordinate system and

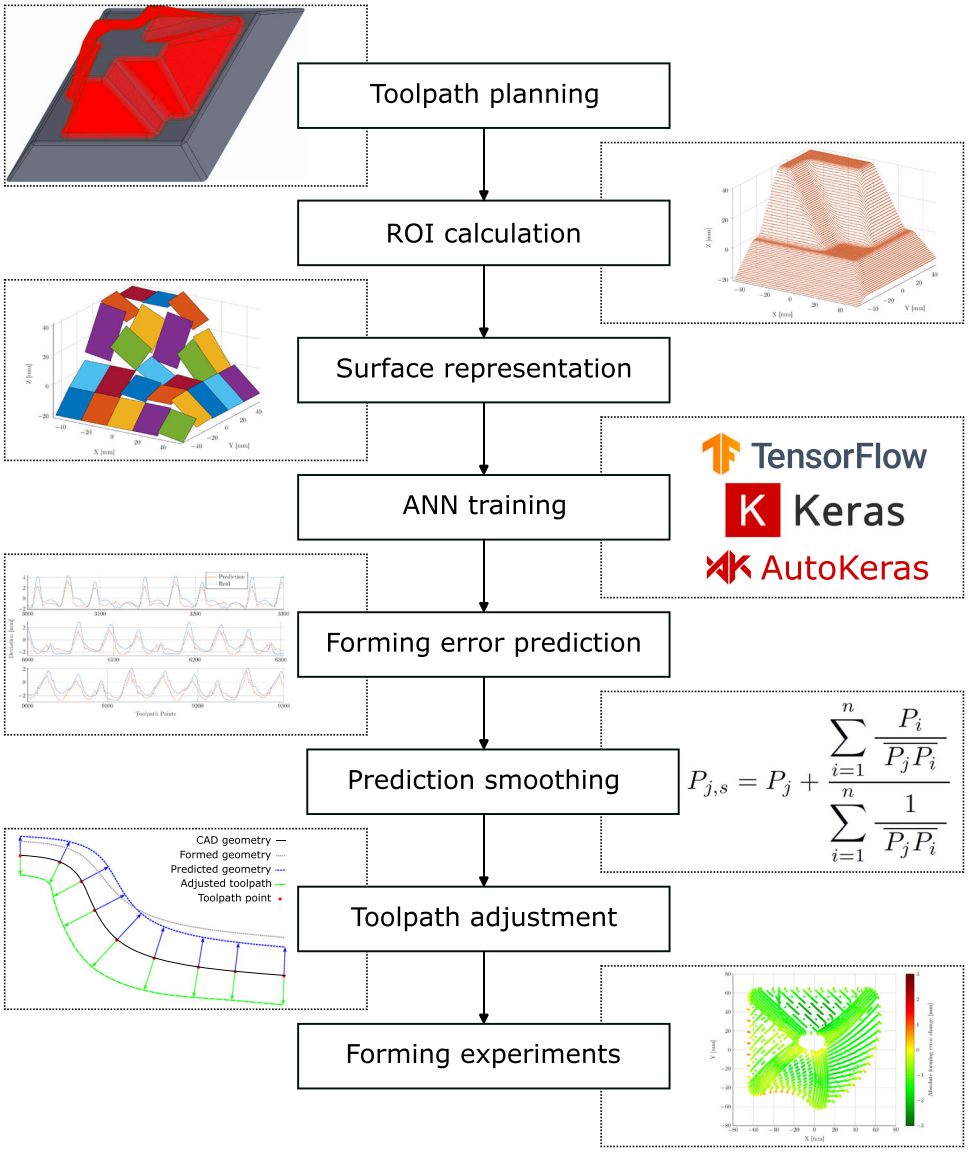

Fig. 1 Flowchart of the individual steps of the feature engineering approach where local surface representations of the geometry surrounding every toolpath point are used to train artificial neural networks

therefore great transferability. To improve the prediction of the trained ANNs, several different local surface representation were implemented.

3 Forming setup

Numerous variations of the ISF process were developed over time. These can be categorized by the kind and quantity of tools, the machinery, toolpath strategies, and auxiliary components like heated elements and unique clamping arrangements. A roboforming system was used to conduct the studies (see Fig. 2). Roboforming is part of duplex incremental forming (DPIF), also referred to as double-sided incremental forming (DSIF) [24]. In DSIF, the sheet is actively formed

Fig. 2 Forming setup at the chair of production systems (LPS). The forming experiments were carried out by two KUKA KR600 robots, each driven by KRC4 controller (background)[23]

by a tool that is usually hemispherical, with assistance from a second tool on the sheet's reverse. In the roboforming process, these tools are driven by industrial robots. Therefore, the large workspace, broad distribution, great kinematic flexibility, and comparatively cheap investment costs of industrial robots are all advantageous to the process. Conversely, industrial robots' open kinematic chains result in low system stiffness and deflections as a result of forming forces, necessitating stiffness compensation during the process [23]. The two distinct tools being used enable DSIF-specific process parameters. The tool center point of the supporting tool on the backside can be rotated around the forming tool to a specific degree, known as the supporting angle, while the forming tool's position is fixed during the forming process based alongside the pre-calculated toolpath. Together with the second distinct roboforming process parameter, the supporting force, the supporting angle demonstrates its benefits. On the reverse side of the sheet, the supporting tool applies a set supporting force in a specific direction using force control. This results in superimposed stress, which positively affects material flow, enhances the forming accuracy, and increases the maximum formable wall angle [25].

3.1 Process database

The training of the ANNs was carried out with the DB4ISF database which was created with the same forming setup [13]. Störkle et al. laid the foundation for establishing this database [26]. They carried out 35 individual forming experiments. In all the experiments, they used the same base geometry (see Fig. 3 G6). The geometry exhibits a wide range of convex and concave radii to cover a lot of process states and enable a generalization of the approach. Furthermore, the geometry was extruded with different wall angles ranging from 30 to 60° in 5° steps. Additionally, to the part's geometry, the general process parameters (step depth, supporting force, supporting angle) were varied in the experiments. Latin hypercube sampling was used to conduct the experimental design [26]. Even though the database consisted of 35 forming experiments with varying parameters and 526,734 toolpath points in total, the data was still too similar for a good generalization. Möllensiep et al. extended the database by forming five new parts which consist of various different geometry features (see Fig 3 G1–G5) [36]. These were systematically derived with a cluster analysis approach where gaps respectively missing geometry features in the initial database were detected. The corresponding geometry features in the gaps were automatically generated and assembled to the new part geometries. Furthermore, the process parameters were varied during the 35 new forming experiments, similar to the initial experimental design. During the forming experiments, three parts cracked and were removed from the database. In total, DB4ISF consists of 67 successful forming experiments with 982,806 toolpath points which were used for training the ANNs. The data of the validation experiments of this publication was added to the database afterward.

4 Surface representations

To enable a direct modification of the toolpath according to the prediction, the part's geometry surrounding every toolpath point is used as input for the ANNs. As only the near environment of the toolpath point has a direct influence on its accuracy, a region of interest (ROI) is specified in the first

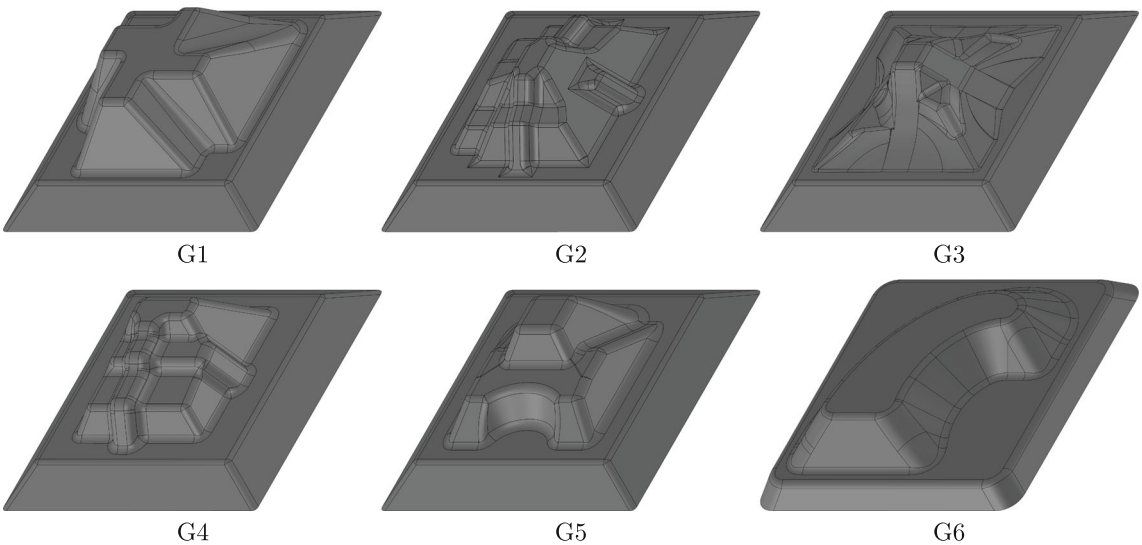

Fig. 3 Overview of the part geometries that were used for establishing a database. The geometries **G1–G5** were conducted by a cluster analysis approach based on the initial geometry **G6**

step. Every portion of the geometry outside of a square with a definable length around the toolpath point is cut off. Afterward, the part geometry is transformed into a standardized coordinate system as the part orientation would have an influence on the prediction results while being inconsequential for the forming process. The center of the coordinate system is the toolpath point. The coordinate system's Y-axis points toward the next toolpath point while the Z-axis maintains its orientation in the main forming direction. The X-axis orientation is set accordingly. After transforming the local part geometry, the local surface representations can be computed. As it is not possible to estimate their prediction performance prior to the training process, their required amount of variables should be limited to around 100 as the best surface representation has to be determined by computation-heavy trial-and-error. Furthermore, the number of variables and the resolution of the representation should be scalable to further improve the prediction results and be able to adjust to new part sizes. We've developed five different surface representations: compensation planes, intersection curves, point series, point cloud rings, and a curvefit grid. They are computed based on an stl-file of the part. For most of the local surface representation, the same pre-processing steps are required. The mesh of the stl is refined where the parameter *subsample distance* specifies the maximum distance between two connected nodes. This is required for intersection operations with the edges as they could be outside the ROI without refinement. Furthermore, some of the surface representations require a grid with equally spaced nodes along the XY-plane. Here, the parameter *grid stepsize* specifies the distance between the grid lines.

4.1 Compensation planes

The idea behind the compensation planes surface representation is it to approximate the surface with several individual planes. To do so, the ROI is divided into a grid and a plane is fitted for every grid field (see Fig. 4, all following displayed surface representations are computed for the same ROI). Every patch is specified by its center and normal vector. For the center, only the Z-value is considered as the position is fixed in the middle of the patch. The *number of patches* can be specified with a parameter. One problem arises during the computation of all surface representations. At the base of the part, the ROI could extend over the edge of the part. Therefore, the surface representations can not be calculated in these areas. However, it is mandatory to provide all input variables to the ANN. To be able to compute these, the edges of the part are extended with a constant height if this is the case. This has the possibility to lead to bad prediction accuracies. The resulting planar surface is identical to unformed areas in the base sheet. These are typically the areas with one of the highest inaccuracies as they are not actively formed and therefore not work-hardened. This is the complete opposite of areas at the edge of the stl-file. Here, the sheet is clamped and does not move at all. To differentiate between these, the prolongation of the part can be translated in negative Z-direction by the parameter *extension offset*. Even though this would be advantageous for all surface representations, only in the case of the compensation planes the representation can be computed without approximation problems.

4.2 Intersection curve

To compute the intersection curve approximation, the XY-, XZ-, and YZ-plane are intersected with the part's geometry (see Fig 5a). As they pass through the origin and therefore the toolpath point, they cover the most important area and neglect the far-off corners of the ROI. The intersection curve consists of the intersection points of the planes with the stl-

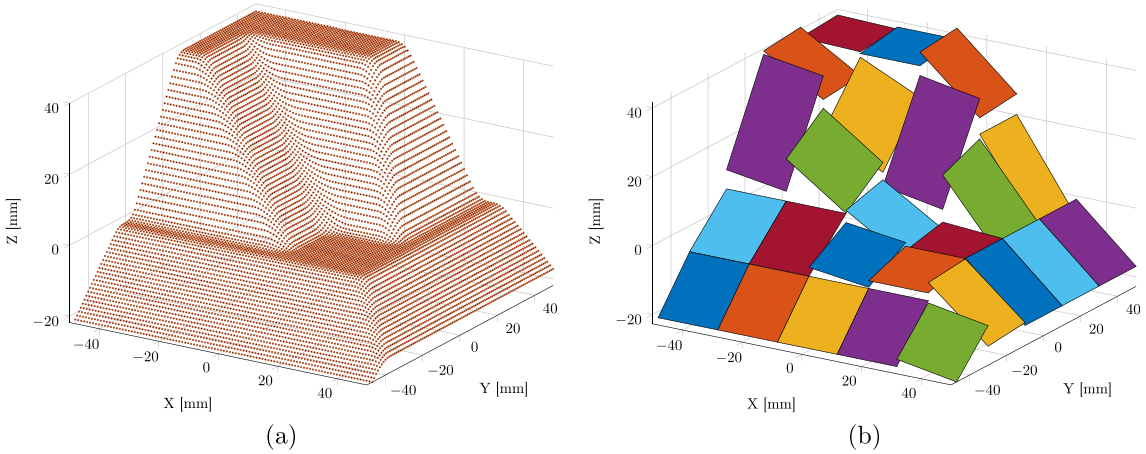

Fig. 4 Depiction of the ROI of one toolpath point with all toolpath points inside it (**a**) and the corresponding compensation planes surface representation (**b**)

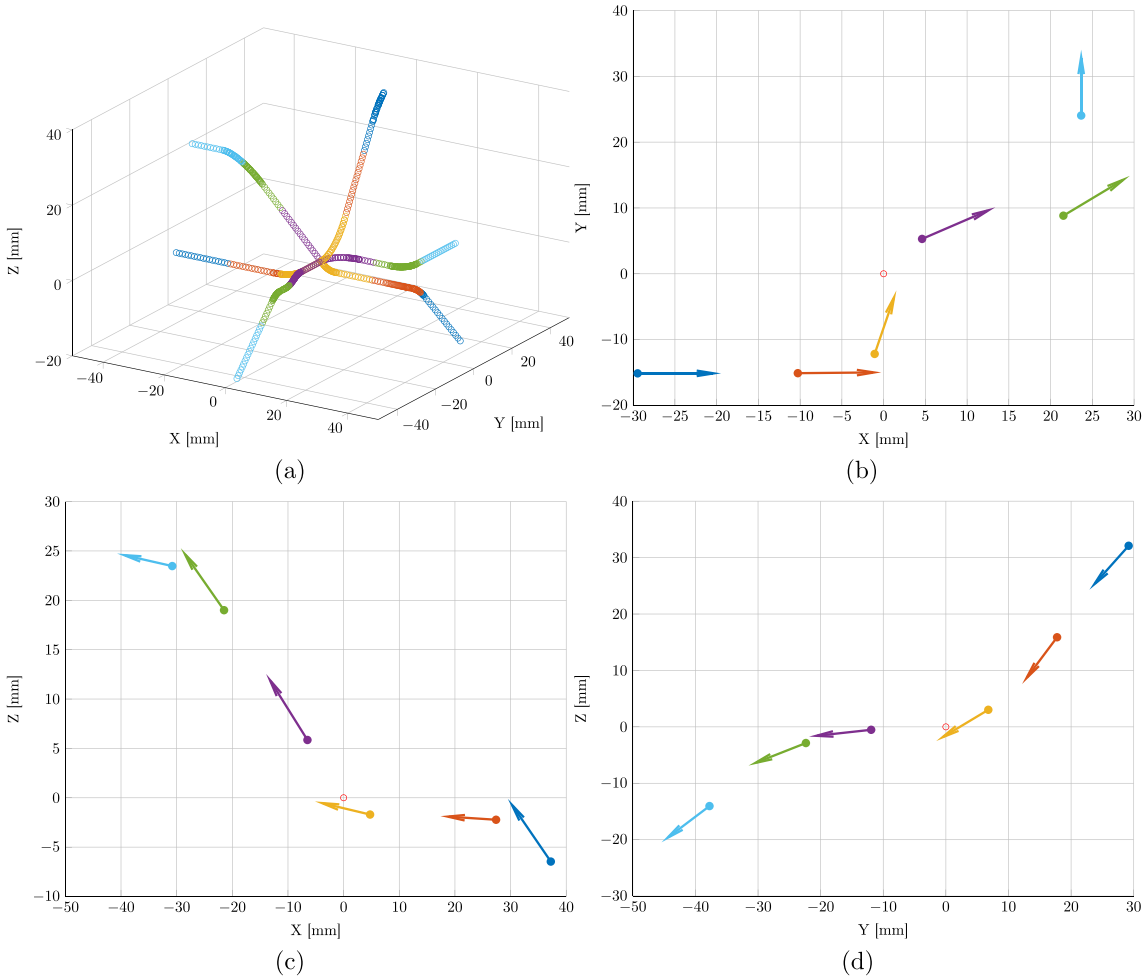

Fig. 5 Intersection curve consisting of the intersection points between the stl-mesh and the basic planes (**a**) and the corresponding division into segments with an arithmetic mean and mean vector (**b–d**)

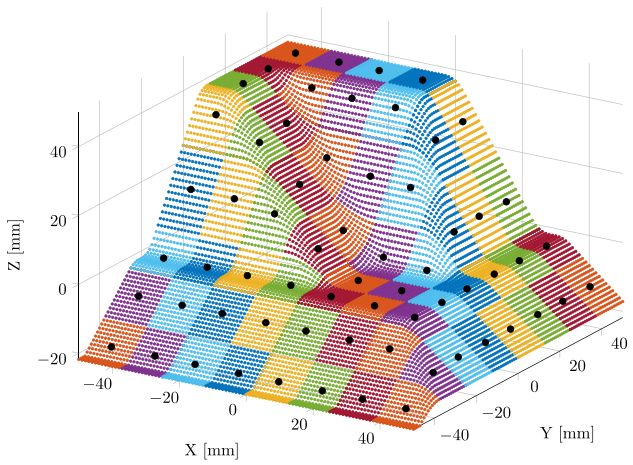

Fig. 6 Rasterization into a grid with the arithmetic mean of the individual grid fields of the point series approximation

mesh. While these intersection curves could be approximated by polynomials, this could lead to the need for more complex ANNs, especially in the case of polynomials of large degree [27]. Therefore, it's approximated in another way. The intersection curve has also the problem, that its length varies depending on the part geometry. As the amount of input variables for the ANN needs to be constant, one can not simply take the coordinates of the intersection points along the curve. Instead, the curve is divided into a specified amount of segments (*point groups*, see color separation in Fig. 5). The segments are described by the arithmetic mean of the individual points in the segment and the mean vector to their next point along the curve (see Fig. 5b–d). As the points are not evenly spaced along the curve, this could distort the approximation. Therefore, *intersection distance* specifies the maximum distance between two points. If the distance between two points is too large, new points are interpolated.

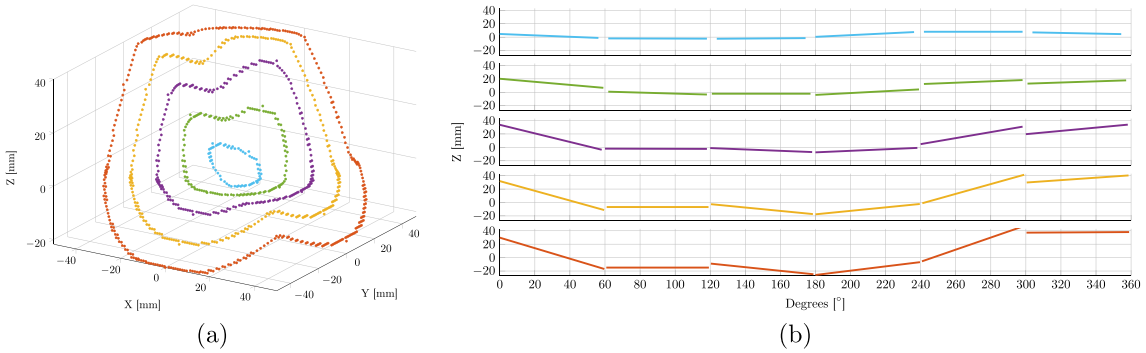

Fig. 7 Depiction of the point cloud rings approximation by intersecting cylinders with the geometry (**a**) and their division into linearly interpolated segments (**b**)

4.3 Point series

The point series approximation is similar to the approach by El Salhi et al. [22]. It is a simplified version of the compensation planes surface representation. The ROI is rasterized into a grid where the *number of patches* can be specified as a parameter as well. On the contrary, only the arithmetic mean of the patch is considered without the normal vector of a fitted plane (see Fig. 6). As the position of the patch and its mid is fixed, only one parameter, the Z-value, is enough to describe the patch. Therefore, a higher amount of patches can be considered compared to the compensation planes approximation without having as much input variables for the ANNs.

4.4 Point cloud rings

The point cloud rings approximation is similar to the intersection curve approximation. Instead of planes, cylinders around the origin with different radii are used for the intersection with the stl-mesh. The result of the intersections is ring-shaped curves composed of individual points (see Fig. 7a). The different cylinders have a parameterizable distance between each other (*ringcurves stepsize*) until they fill out the ROI. The analogy to the intersection curve, the rings are divided into different segments (*number of segments*). Afterward, the individual segments are interpolated linearly according to their degree around the center and their Z-value. The coefficients of the interpolation are used as ANN input.

4.5 Curvefit grid

Lastly, the curvefit grid approximation is also based in intersections. Planes in the XZ- and YZ-plane are intersected with the part geometry (see Fig. 8a). They are evenly spaced according to the *curfefit grid stepsize* parameter. In contrary to the intersection curve approximation, the individual intersection curves can not be approximated in an equal level of detail due to their higher number. Instead, they are approx-

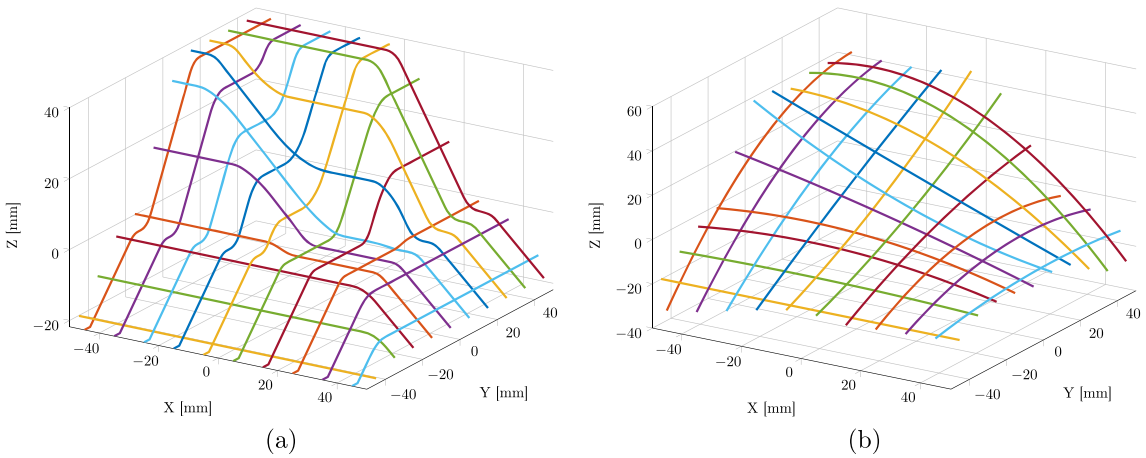

Fig. 8 Evenly spaced intersection curves of the curvefit grid approximation due to intersections between the part and planes in the XZ- and YZ-plane (**a**) and their second-degree polynomial approximation (**b**)

Table 1 Parameter combination of the surface representations that were used as input for the ANNs during the training alongside the general process parameters step depth, sheet thickness, tool radii, support angle, and support force. The forming error of the individual toolpath point was chosen as the output. A subset of 280,070 toolpath points was used

Intersection curve	ROI radius [mm]	Point groups	Intersection distance [mm]			ANN input variables
ANN1	20	4	1			55
ANN2	50	6	0.5			79
ANN3	50	10	1			127
ANN4	50	0.5	10			127
ANN5	80	10	0.5			-
Compensation planes	ROI radius [mm]	Subsample distance [mm]	Grid stepsize [mm]	Number of patches	Extension offset [mm]	ANN input variables
ANN6	20	2	1	16	0	55
ANN7	50	2	1	36	0	115
ANN8	50	1	0.5	25	0	82
ANN9	50	1	0.5	36	0	115
ANN10	50	2	1	36	15	115
ANN11	50	1	0.5	25	15	82
ANN12	80	2	1	25	0	82
Point series	ROI radius [mm]	Subsample distance [mm]	Grid stepsize [mm]	Number of patches		ANN input variables
ANN13	20	2	1	36		43
ANN14	50	2	1	100		107
ANN15	50	1	0.5	64		71
ANN16	50	1	0.5	100		107
ANN17	80	2	1	100		107
Point cloud rings	ROI radius [mm]	Subsample distance [mm]	Grid stepsize [mm]	Ringcurves stepsize [mm]	Number of segments	ANN input variables
ANN18	20	2	1	5	6	55
ANN19	50	2	1	5	6	127
ANN20	50	1	0.5	10	10	107
ANN21	50	1	0.5	5	10	207
ANN22	80	2	1	10	10	167
Curvefit grid	ROI radius [mm]	Subsample distance [mm]	Grid stepsize [mm]	Curvefit grid stepsize [mm]		ANN input variables
ANN23	20	2	1	5		55
ANN24	50	2	1	5		127
ANN25	50	1	0.5	10		67
ANN26	80	2	1	20		55

Table 2 Accuracy of the trained ANNs and the chosen structure

Surface representation	MSE [mm^2]	MAE [mm]	Optimizer	ANN hidden layer structure	Learning rate
ANN1	1.2423	0.8247	Adam	512-32-1024	0.001
ANN2	0.7216	0.6603	Adam weight decay	256-1024-512	0.001
ANN3	0.6815	0.6405	Adam weight decay	64-256-1024	0.001
ANN4	0.7767	0.6805	Adam weight decay	1024-512-32	0.001
ANN6	1.2455	0.8943	Adam weight decay	128-1024	0.001
ANN7	1.2348	0.8580	Adam weight decay	128-64	0.001
ANN8	1.3219	0.8944	Adam	256-512-32	0.001
ANN9	1.1040	0.8104	Adam weight decay	256-512-512	0.001
ANN10	1.2087	0.8245	Stochastic gradient decent	512-128-dropout (0.25 %)	0.01
ANN11	1.4545	0.9551	Adam weight decay	1024-64	0.0001
ANN12	1.3006	0.9052	Adam	64-1024-256	0.001
ANN13	1.1686	0.8495	Adam	32-256-32	0.001
ANN14	1.4960	0.9368	Adam	128-1024-32	0.001
ANN15	1.5420	0.9794	Adam weight decay	128-1024-32	0.001
ANN16	1.2863	0.8905	Adam	512-256-128	0.0001
ANN17	1.8955	1.1421	Adam weight decay	1024-64-32	0.001
ANN18	1.5196	0.9821	Adam weight decay	512-1024	0.001
ANN19	1.3007	0.9044	Adam weight decay	1024-256-32	0.001
ANN20	1.4284	0.9595	Adam	512-256-1024	0.001
ANN21	1.5325	0.9009	Adam weight decay	512-512-512 (batch normalization)	0.001
ANN22	1.7611	1.0313	Adam weight decay	32-1024-256	0.001
ANN23	1.4335	0.9575	Adam	1024-1024-256	0.0001
ANN24	0.9234	0.7484	Adam weight decay	32-1024-128	0.001
ANN25	1.0904	0.8256	Adam weight decay	512-1024	0.001
ANN26	2.1810	1.2566	Adam weight decay	512-16	0.001

imated by a polynomial of the second degree (see Fig. 8b). Therefore, every intersection curve can be described by the three polynomial coefficients.

5 Automatic machine learning

The surface representations are afterward used as the input variables during the training process to predict the forming error in the normal direction of every toolpath point. As it is unclear what representation is a suitable input, the training process is standardized by using automatic machine learning (AutoML). Software-wise, TensorFlow [28], Keras [29] and AutoKeras [30] are used because their ANNs can be imported easily into the pre-existing pathplanning programs for roboforming written in MATLAB. Due to the excessive computation times of several hours for one surface representation of one toolpath, only a fraction of the DB4ISF data is used in the first step. Three experiments of G1-G5 each and four of G6 are used, resulting in 19 experiments and 280,070 toolpath points (E02, E04, E06, E09, E11, E13, E16, E18, E20, E23, E25, E27, E30, E32, E34, E38, E48, E58, E68). Alongside the surface representation,

Table 3 Prediction accuracy of the former three best ANNs, now trained with the full database excluding the majority of part G6 to prevent overrepresentation

Representation	MSE [mm^2]	MAE [mm]	ANN structure	Optimizer	Learning rate
Intersection curve (ANN3FDB)	0.6695	0.6302	1024-1024-1024	Adam weight decay	0.0001
Compensation planes (ANN9FDB)	1.0772	0.8018	512-1024-512	Adam weight decay	0.001
Curvefit grid (ANN24FDB)	0.9327	0.7635	1024-1024-128	Adam weight decay	0.0001

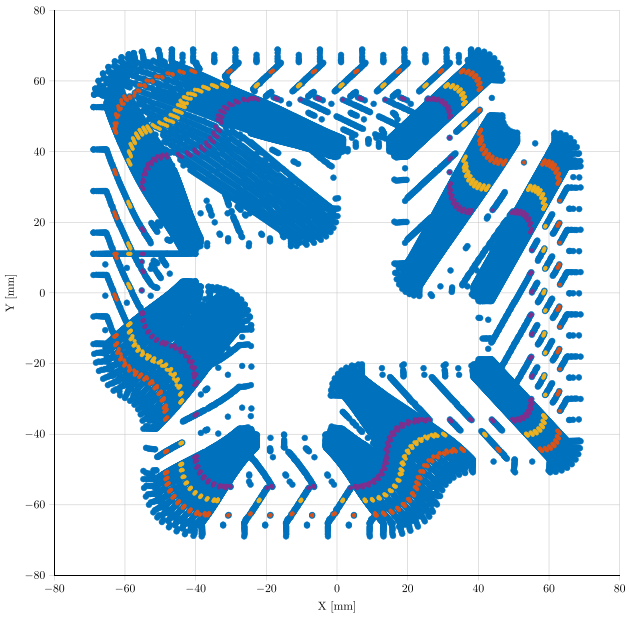

Fig. 9 Depiction of the two circulations each at three different spots of Fig. 10, starting from the outside

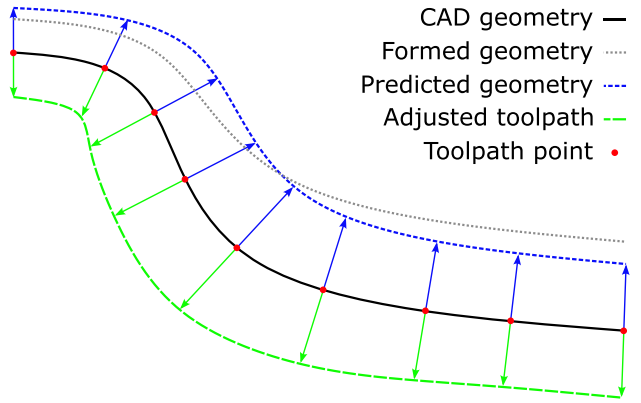

Fig. 11 Principle of the toolpath adjustment in the opposite direction of the prediction of the ANN

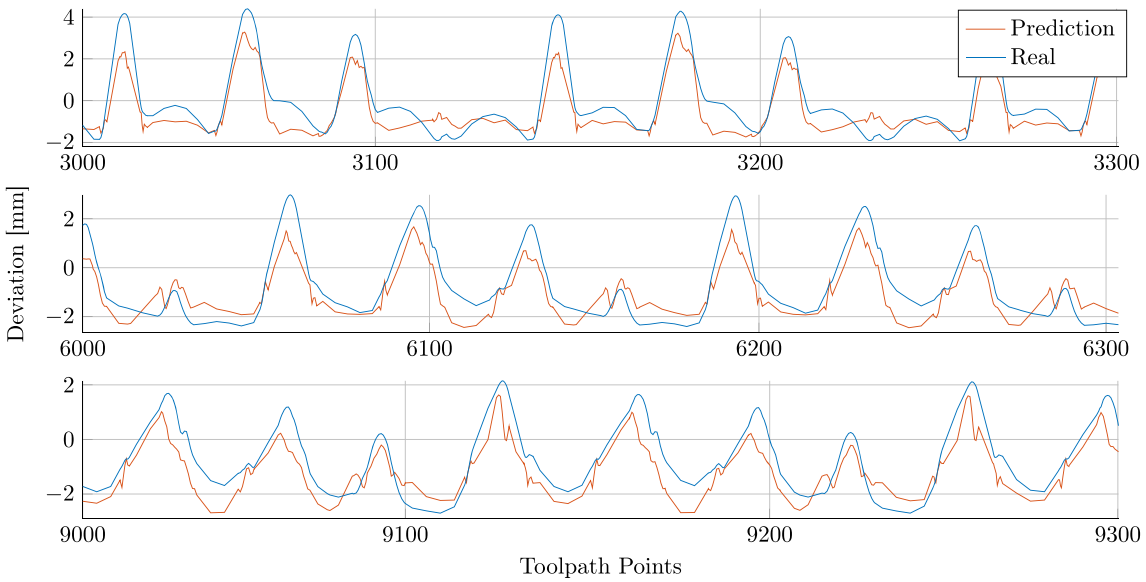

Fig. 10 Comparison between the prediction of ANN3FDB and the measured forming error of experiment E02 of part G1 which was used as part of the test data

Table 4 Influence of the smoothing parameters on the accuracy of the prediction of ANN3FDB on E02

Distance [mm]	Iterations	MSE before [mm^2]	MSE after [mm^2]	Variance before [mm^2]	Variance after [mm^2]
5	1	0.8079	0.7984	0.1005	0.0636
5	2	0.8079	0.8027	0.1005	0.0514
8	1	0.8079	0.8056	0.1005	0.1196
8	2	0.8079	0.8253	0.1005	0.0951

the following general process parameters are also used as input variables: step depth, sheet thickness, tool radii, support angle, and support force. However, sheet thickness and tool radii were identical in all experiments and neglected during the preprocessing. To find the best surface representation, the training is executed according to an experimental design (see Table 1). The parameterizable variables of the representations and common variables like the *ROI radius* are varied in fixed steps to make them reusable for other representations and reduce computation times. We've aimed to train five different ANNs for every surface representation. However, the representations for ANN5 could not be fully computed for the whole database due to algorithmic problems with the high ROI and the intersection. The compensation planes were trained two additional times to cover the influence of the *extension offset* (ANN10, ANN11). In the case of the curvefit grid representation, one highly detailed parameter combination's computation was aborted as it would have taken weeks to finish. As the data of the individual forming experiments is to some degree similar due to the incremental nature of the process, no standard train-validation-test-split was used. Instead, the data of G1 (E02, E04, E06) was used as test data, and 20 % of the remaining data was used for validation. All data was normalized to exhibit a mean of 0 and a variance of 1. Tensorflow was used for final training and AutoKeras for hyperparameter tuning. Stochastic gradient descent [31], Adam [32] and Adam with weight decay [33] could be chosen as optimizer. All ANNs had the same base structure consisting of input for structured data, structured data block, and a regression head. The structured data block consisted of variation layers and neurons with optional dropout- or batch-normalization-layers. The regression head used the mean squared error (MSE) and was fed with the forming error in the normal direction of the toolpath point. While the learning rate could be varied during the training, the number of epochs and trials was set to 20, and the activation function to ReLu. When looking at the results (see Table 2), it can be observed that for most surface representations the MSE and mean absolute error (MAE) are between 0.9 and 1.6 mm/mm^2. The prediction is best for ANN2 and ANN3 and worst for ANN16, ANN22, and ANN26. With regard to the region of interest, it can be seen that a value of 50 mm for the radius provides the best result for most representations. Only for the compensation planes are the predictions similar for all ROIs and for the point series representation the prediction for the ROI radius of 20 mm exhibits the best results. In most cases, varying the other parameters only yields small differences. The best results are shown for the intersection curve approximation. With a mean square error of 0.6815 mm^2, the prediction for ANN3 is the most accurate. ANN2 also delivers a very good result, while ANN4 is somewhat less accurate. The prediction is worst for ANN1. The use of an ROI radius of 20 mm therefore appears to produce less accurate results. The predictions for the compensation planes are the third best on average. All are in the same range and achieve an MSE that is approximately 0.5 mm^2 worse than for the intersection curve approximation. ANN9 has the best result with an MSE of 1.10396 mm^2. The *extension offset* does not appear to have a positive influence on the prediction and will therefore not be considered in future research. An offset of 15 mm provides similar values for the two representations as without the offset. Similarly, no significant differences can be recognized for a higher or lower resolution of the ROI point cloud for the four representations based on it. The predictions for the point series representation are worse, with only ANN13 achieving a good prediction quality of 1.1685 mm^2 MSE. The point cloud ring representation delivers similar results. The best result is a mean square error of 1.3007 mm^2 for ANN18. The curvefit grid representation shows the best prediction for ANN24 with an MSE of 0.9234 mm^2. The predictions for the representation with an ROI radius of 50 mm are significantly better than the others. As the best results are shown for the intersection curve, the compensation planes and the curvefit grid representations, ANN3, ANN9, and ANN24 are selected for further analyses.

5.1 Training with the complete database

To improve the prediction accuracy of the three best ANNs (ANN3, ANN9, ANN24), the training is repeated with a higher amount of data. For G1-G5 all available data is used. As G6 resembles half of the DB4ISF database, only 10 individual experiments are used for training to prevent overrepresentation. For simplicity, this subset is still referred to as a full database (FDB) in the following. All in all, 42 forming experiments with 632,380 toolpath points and therefore input rows are used for training. Identical to before, all experiments of part G1 are used as test data, and 20 % of the remaining data is used for validation. The amount of trials is increased to 50, while all other options remained identical. The accuracy of the newly trained ANNs is presented in Table 3. Their accuracy is only slightly better than that with data of only 19 forming experiments. The best result is again provided by the intersection curve approximation (ANN3FDB) with an MSE of 0.66953 mm^2. Compared to the ANNs calculated with 19 tests, the MSE for the networks is about 0.03 mm^2 better, only for the curvefit grid representation the prediction is 0.01 mm^2 worse. The fact that no significantly better predictions are achieved by the exten-

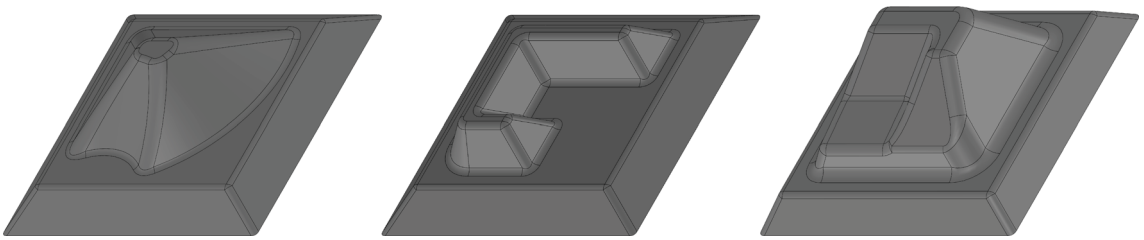

Fig. 12 Parts used in the validation experiments

sion could be due to the fact that although more forming experiments are used for training, these describe the same geometries. The differences between the geometries have the greatest influence on the deviations of the components. Although expanding the training data increases the size of the training table, it does not increase the number of geometric features that are included in the training. Tuning the hyperparameters by hand only yielded neglectable results, although it could be observed that a high learning rate worsened the prediction accuracy by a lot. Nevertheless, the trained ANNs still achieve a really good prediction accuracy. When analyzing the prediction of ANN3FDB, which had the lowest MAE and MSE, for part G1 which was used as test data, it can be seen that the prediction and measured forming error are really close together, exemplarily shown for E02 in Fig. 10. The figure depicts the prediction for two circulations each at three different points of the geometry, starting from the outside (see Fig. 9). As it can be seen, the prediction is really close to the actual trend of the forming error, even at high peaks. Although, there are some small kinks. These resemble a problem if the toolpath should be modified according to the prediction. While there are advanced approaches that use Fourier-transformation [34] or weighting factors [19], it is still common to offset the toolpath points in the direction opposite to the prediction, what was also done during this research (see Fig. 11). This leads to kinks in the toolpath. As the robots are utilizing a stiffness compensation and force control, those kinks would stimulate the system and result in oscillations which would worsen the forming accuracy. Therefore, the modified toolpath needs to be smoothed.

5.2 Path smoothing

To smooth the toolpath, we have applied a weighted three-dimensional Laplacian smoothing [35]. All other toolpath points and their prediction inside a distance around the smoothed toolpath point are considered. Their influence is weighted according to their distance toward the smoothed toolpath point to increase the influence of closer points. Let there be the prediction of the forming error in normal direction P with j denoting the initial prediction of the ANN and s denoting the smoothed one, i numbering the toolpath points inside the smoothed area and n as their total amount, then the prediction can be smoothed according to following equation:

$$P_{j,s} = P_j + \frac{\sum_{i=1}^{n} \frac{P_i}{\overline{P_j P_i}}}{\sum_{i=1}^{n} \frac{1}{\overline{P_j P_i}}} \tag{1}$$

The smoothing can be repeated to increase its effect. However, if the smoothing is too strong, the prediction accuracy could be decreased. When comparing several smoothing options, it can be seen that a large smoothing distance has a negative influence on the prediction accuracy (see Table 4). While the MSE is almost unaffected by the smoothing, the variance decreases especially for a smoothing distance of 5 mm and two smoothing iterations. The variance is especially important for a smooth robot movement. Therefore, these parameters are used to smooth the prediction in the validation experiments. A more detailed analysis of the influ-

Table 5 Prediction accuracy of the three best-performing ANNs for the three validation geometries, formed with standard process parameters

ANN	Formed geomtry	MSE [mm^2]	MAE [mm]
ANN3FDB	Val1	0.7176	0.5571
	Val2	1.9770	1.2542
	Val3	0.8925	0.6889
ANN9FDB	Val1	0.6887	0.5127
	Val2	1.8404	1.1679
	Val3	0.9711	0.7682
ANN24FDB	Val1	0.8527	0.7068
	Val2	1.0871	0.7570
	Val3	1.0919	0.8908

ence of different smoothing parameters should be researched in the future.

6 Validation

To validate the accuracy of the prediction of the ANNs and their potential to increase the forming accuracy by modifying the toolpath, three new parts were designed that have not been part of the training process (see Fig. 12). Val1 is a standard geometry that was used in many validation experiments of the LPS as it features planar surfaces and convex and concave radii. Val2 resembles a part that is difficult to form with ISF. The middle section suffers from large forming errors due to subsequent deformation caused by forces and torques induced by forming the surrounding areas. Lastly, Val3 possesses a plateau that could cause forming errors as spiral toolpath contain only few toolpath points in the area. The three formerly best-performing ANNs (ANN3FDB, ANN9FDB, ANN24FDB) are used to predict the forming accuracy of these three parts utilizing the standard general process parameters that were used in most experimental studies of the LPS (sheet metal: 0.8 mm DC04, step depth: 0.5 mm, tool radii: 4 mm, support force: 250 N, support angle: 0.95). Afterward, the prediction is smoothed and the parts are formed with the aforementioned parameters. Lastly, the toolpath is modified according to the prediction of ANN3FDB. The individual toolpath points are shifted in the opposite direction of the predicted error in the normal direction of the sheet. Afterward, the three validation parts are formed utilizing the standard forming parameters without adjusting the toolpath according to the prediction. Therefore, these can be used as a reference for validating the achieved improvements. When comparing the prediction accuracy of the three ANNs for the validation parts, it can be seen that none of the ANNs clearly outperforms the others. Instead, every ANN is able to predict one part more precise that the others (see Table 5). This indicates that ANN ensembles might be a promising approach to research to combine the advantages of the individual representations and therefore ANNs. Especially as ANN3FDB and ANN9FDB exhibit a similar prediction error, while ANN24FDB clearly outperforms them for geometry Val2. Despite the differences between the three ANNs, ANN3FDB was used to modify the toolpath to improve the forming result as is had the prediction accuracy during the training process. The numeric improvement is shown in Table 6 and the distribution of the forming errors can be seen in Fig. 13. Particularly, the MAE could be reduced substantially for Val1 (68.5 %) and Val3 (59.0 %). Contrarily, the reduction for Val2 is minimal (17.7 %). The reason for this small increase results partly from the worse prediction performance of ANN3FDB. Although the main reason is the middle section of the part, close to the lower end of the geometry (see Fig. 14). Even with a perfect prediction, it would be impossible to increase the forming accuracy by simply shifting the toolpath points in the opposite direction of the prediction. This would bring them on the wrong side of the initial flat sheet so that the tools would have to pull instead of push at the start of the forming process which is impossible with ISF. This also caused a crash of the supporting tool with the initial sheet. Therefore, this was prevented by our toolpath planning software where toolpath points can only be shifted until they reach the initial sheet plane. To fully utilize the prediction capabilities of the presented ANNs, it is mandatory to develop advanced methods of toolpath manipulation. Nevertheless, the validation experiments of the parts Val1 and Val3 demonstrate the already great potential of the ANNs to predict and improve the forming accuracy.

Table 6 Forming accuracy of the three validation geometries with standard forming parameters and with a toolpath modified according to the prediction of ANN3FDB

	Val1	Val2	Val3
MAE [mm]	1.4883	1.7839	1.6320
MAE new toolpath [mm]	0.4701	1.4683	0.6693
Maximum absolute error [mm]	3.9460	11.1944	4.9788
Maximum absolute error new toolpath [mm]	1.7311	13.6347	4.5871

7 Conclusion

Incremental sheet forming is a process capable of economically forming sheet metal parts in small batch sizes which distribution is hindered by its low forming accuracy. To increase the accuracy, the authors presented a highly transferable machine-learning approach based on local surface representations of the sheet area surrounding each toolpath point. In forming experiments of sheet metal parts that were not part of the training process, the mean average forming error could be reduced by up to 68.5 % by modifying the toolpath according to the prediction of the formerly trained artificial neural network. To enable it other research institutes to utilize and further develop the approach, the machine learning framework, and all experimental data are published as open-source. The results of the validation experiments indicate that advanced toolpath modifying and smoothing algorithms and ensemble deep learning methods are promis-

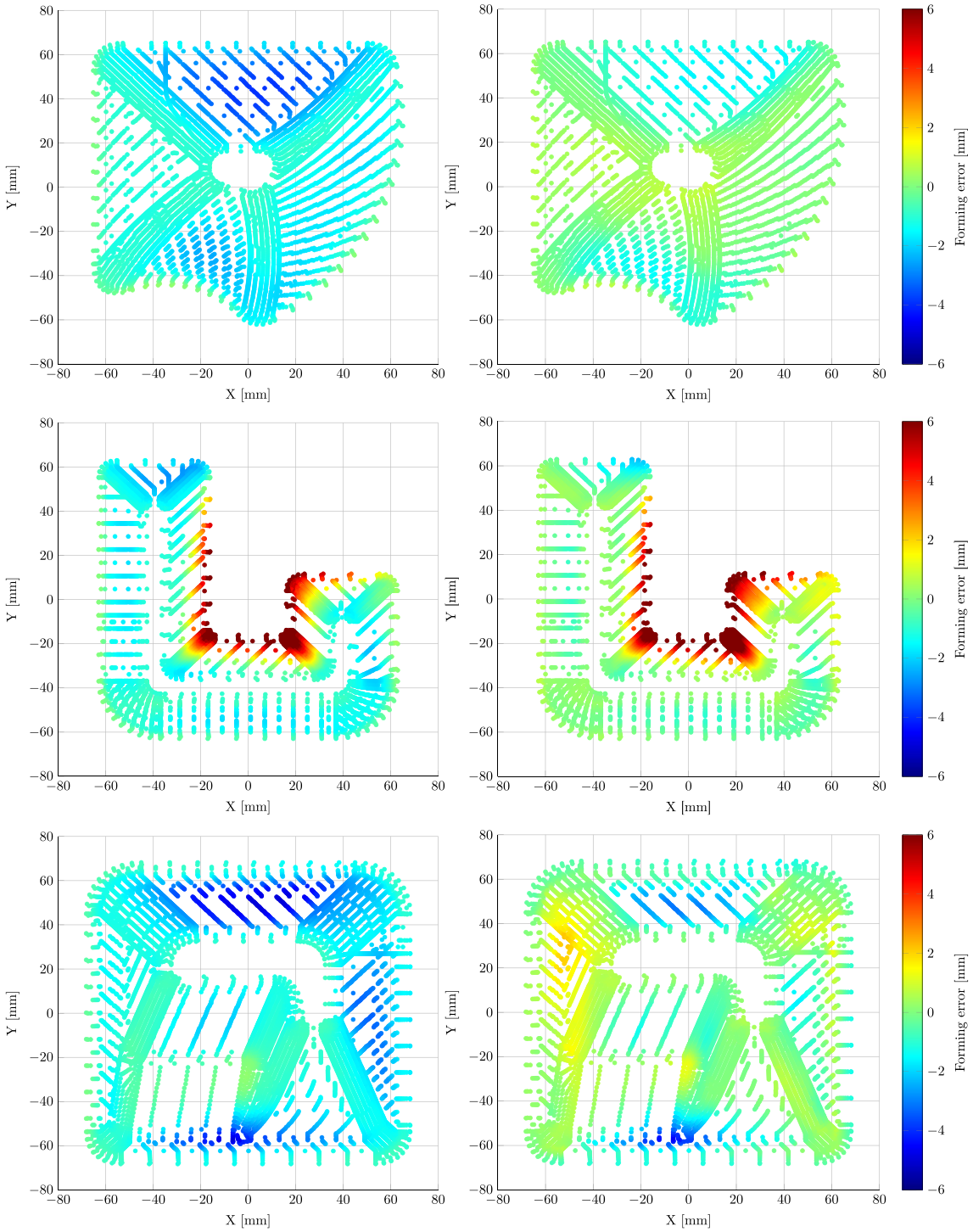

Fig. 13 Distribution of the forming accuracy of the three validation geometries with standard forming parameters (left side) and with a toolpath modified according to the prediction of ANN3FDB (right side)

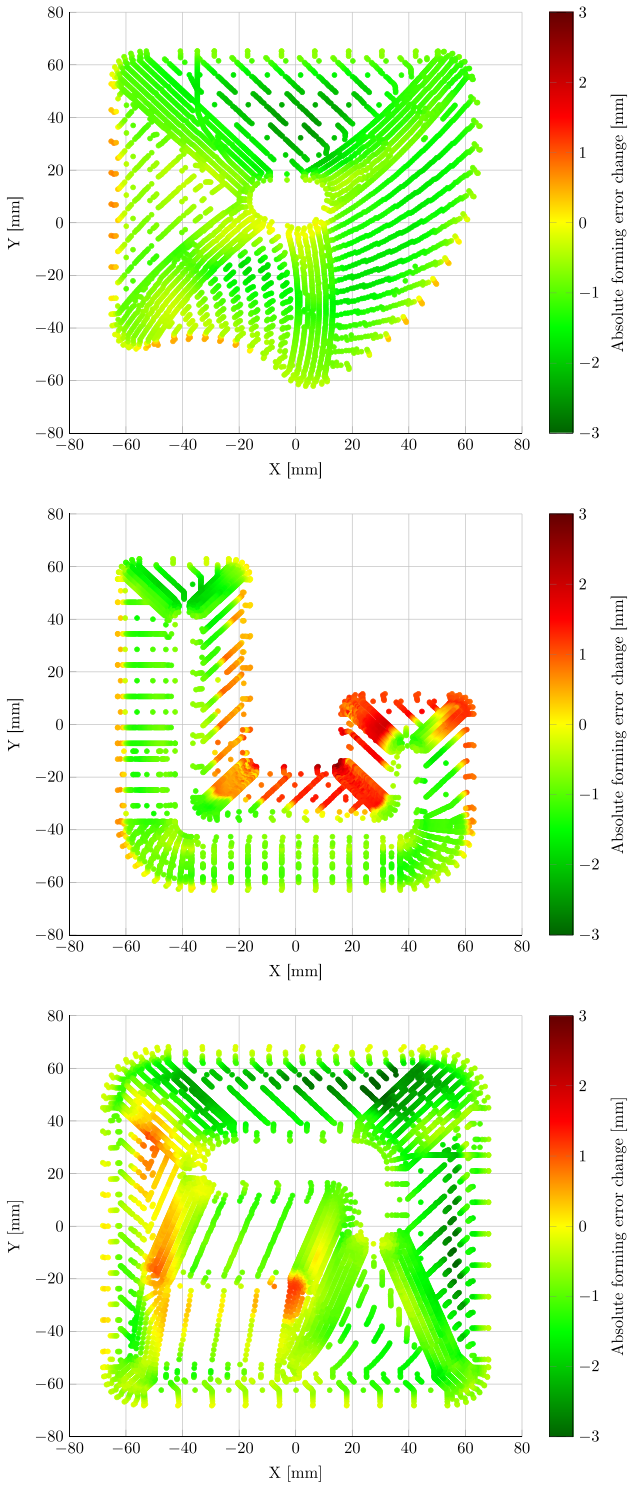

Fig. 14 Change of the absolute forming error of the validation geometries formed with a toolpath modified according to the prediction of ANN3FDB compared to the parts formed with standard parameters

ing approaches to even further increase the prediction and forming accuracy.

Funding Open Access funding enabled and organized by Projekt DEAL. This research work was funded by the German Research Foundation (DFG) within the research project "Prediction and compensation of subsequent deformation in robot-based incremental sheet forming by application of machine learning" (project number 457407945). The authors thank the DFG for promoting and financing the research.

Availability of data and materials The machine learning framework capable of data management, calculation of surface representations and training data tables, prediction of the forming accuracy with several pretrained artificial neural networks, toolpath adjustments and smoothing can be downloaded at [10].
The whole process database used for training and the validation experiments consisting of 76 experiments and their general process data, CAD files, toolpaths, robot programs used for forming, digitization and deviation of every toolpath point in normal direction can be downloaded at [13].

Declarations

Conflict of interest The authors declare no competing interests.

References

1. Li Y, Chen X, Liu Z, Sun J, Li F, Li J, Zhao G (2017) A review on the recent development of incremental sheet-forming process. Int J Adv Manuf Technol 92(5–8):2439–2462. https://doi.org/10.1007/s00170-017-0251-z
2. Abd Ali R, Chen W, Al-Furjan MSH, Jin X, Wang Z (2019) Experimental investigation and optimal prediction of maximum forming angle and surface roughness of an al/sus bimetal sheet in an incremental forming process using machine learning. Materials 12(24):4150. https://doi.org/10.3390/ma12244150
3. Nagargoje A, Kankar PK, Jain PK, Tandon P (2021) Application of artificial intelligence techniques in incremental forming: a state-of-the-art review. J Intell Manuf 34(3):985–1002. https://doi.org/10.1007/s10845-021-01868-y
4. Low DWW, Chaudhari A, Kumar D, Kumar AS (2022) Convolutional neural networks for prediction of geometrical errors in incremental sheet metal forming. J Intell Manuf 34(5):2373–2386. https://doi.org/10.1007/s10845-022-01932-1

5. Bautista-Monsalve F, García-Sevilla F, Miguel V, Naranjo J, Manjabacas MC (2021) A novel machine-learning-based procedure to determine the surface finish quality of titanium alloy parts obtained by heat assisted single point incremental forming. Metals 11(8):1287. https://doi.org/10.3390/met11081287
6. Kumar P, Singh H (2023) Experimental analysis of tool geometry and tool rotation in SPIF process on aa7075-o alloy using ml and ANN approach. Int J Interact Des Manuf. https://doi.org/10.1007/s12008-023-01535-x
7. Jiang Z, Ehmann KF, Cao J (2022) Prediction of forming temperature in electrically-assisted double-sided incremental forming using a neural network. J Mater Process Technol 302:117486. https://doi.org/10.1016/j.jmatprotec.2021.117486
8. Najm SM, Paniti I (2022) Investigation and machine learning-based prediction of parametric effects of single point incremental forming on pillow effect and wall profile of almn1mg1 aluminum alloy sheets. J Intell Manuf 34(1):331–367. https://doi.org/10.1007/s10845-022-02026-8
9. Pham QT, Le HS, Nguyen AT, Xiao X, Kim Y-S, Nguyen VD, Tran HS, Van Tran X (2022) A machine learning-based methodology for identification of the plastic flow in aluminum sheets during incremental sheet forming processes. Int J Adv Manuf Technol 120(5–6):3559–3584. https://doi.org/10.1007/s00170-022-08698-z
10. Möllensiep D, Detering L, Lademann D, Kulessa P, Steinhof M, Ohm M, Störkle DD, Kuhlenkötter B (2023) ML4ISF: machine learning framework for incremental sheet forming. Zenodo. https://doi.org/10.5281/ZENODO.10036335
11. Carette Y, Duflou JR (2022) Mastering the complexity of incremental forming: geometry-based accuracy prediction using machine learning. In: Dix, M., Kräusel, V. (eds.) The 28th Saxon Conference on Forming Technology SFU and the 7th International Conference on Accuracy in Forming Technology ICAFT. SFU/ICAFT 2022. MDPI, ??? https://doi.org/10.3390/engproc2022026012
12. Harfoush A, Haapala KR, Tabei A (2021) Application of artificial intelligence in incremental sheet metal forming: a review. Procedia Manuf 53:606–617. https://doi.org/10.1016/j.promfg.2021.06.061
13. Möllensiep D, Schäfer J, Altmann P, Störkle DD, Kuhlenkötter B (2023) DB4ISF: An incremental sheet forming database. Zenodo. https://doi.org/10.5281/ZENODO.10000815
14. Syafrudin M, Alfian G, Fitriyani N, Rhee J (2018) Performance analysis of IoT-based sensor, big data processing, and machine learning model for real-time monitoring system in automotive manufacturing. Sensors 18(9):2946 DOIurl-https://doi.org/10.3390/s18092946
15. Cordts M, Omran M, Ramos S, Rehfeld T, Enzweiler M, Benenson R, Franke U, Roth S, Schiele B (2016) The cityscapes dataset for semantic urban scene understanding. arXiv https://doi.org/10.48550/ARXIV.1604.01685
16. Bellman R (1984) Dynamic programming. Princeton Univ. Pr, Princeton, NJ
17. Prasad AD, Balu A, Shah H, Sarkar S, Hegde C, Krishnamurthy A (2021) NURBS-diff: a differentiable programming module for NURBS. arXiv https://doi.org/10.48550/ARXIV.2104.14547
18. Lu B, Chen J, Ou H, Cao J (2013) Feature-based tool path generation approach for incremental sheet forming process. J Mater Process Technol 213(7):1221–1233. https://doi.org/10.1016/j.jmatprotec.2013.01.023
19. Behera AK, Verbert J, Lauwers B, Duflou JR (2013) Tool path compensation strategies for single point incremental sheet forming using multivariate adaptive regression splines. Comput Aided Des 45(3):575–590. https://doi.org/10.1016/j.cad.2012.10.045
20. Verbert J, Behera AK, Lauwers B, Duflou JR (2011) Multivariate adaptive regression splines as a tool to improve the accuracy of parts produced by FSPIF. Key Eng Mater 473:841–846. https://doi.org/10.4028/www.scientific.net/kem.473.841
21. Khan MS, Coenen F, Dixon C, El-Salhi S, Penalva M, Rivero A (2014) An intelligent process model: predicting springback in single point incremental forming. Int J Adv Manuf Technol 76(9–12):2071–2082. https://doi.org/10.1007/s00170-014-6431-1
22. El Salhi S, Coenen F, Dixon C, Khan MS (2015) Predicting "springback" using 3d surface representation techniques: a case study in sheet metal forming. Expert Systems with Applications 42(1):79–93. https://doi.org/10.1016/j.eswa.2014.07.041
23. Möllensiep D, Gorlas T, Kulessa P, Kuhlenkötter B (2021) Real-time stiffness compensation and force control of cooperating robots in robot-based double sided incremental sheet forming. Prod Eng Res Devel 15(5):683–699. https://doi.org/10.1007/s11740-021-01052-4
24. Kreimeier D, Buff B, Magnus C, Smukala V, Zhu J (2011) Robot-based incremental sheet metal forming - increasing the geometrical accuracy of complex parts. Key Eng Mater 473:853–860. https://doi.org/10.4028/www.scientific.net/kem.473.853
25. Meier H, Magnus C, Smukala V (2011) Impact of superimposed pressure on dieless incremental sheet metal forming with two moving tools. CIRP Ann 60(1):327–330. https://doi.org/10.1016/j.cirp.2011.03.134
26. Störkle D, Altmann P, Möllensiep D, Thyssen L, Kuhlenkötter B (2019) Automated parameterization of local support at every toolpath point in robot-based incremental sheet forming. Procedia Manuf 29:67–73. https://doi.org/10.1016/j.promfg.2019.02.107
27. Chong KFE (2020) A closer look at the approximation capabilities of neural networks. arXiv https://doi.org/10.48550/ARXIV.2002.06505
28. Abadi M, Agarwal A, Barham P, Brevdo E, Chen Z, Citro C, Corrado GS, Davis A, Dean J, Devin M, Ghemawat S, Goodfellow I, Harp A, Irving G, Isard M, Jia Y, Jozefowicz R, Kaiser L, Kudlur M, Levenberg J, Mané D, Monga R, Moore S, Murray D, Olah C, Schuster M, Shlens J, Steiner B, Sutskever I, Talwar K, Tucker P, Vanhoucke V, Vasudevan V, Viégas F, Vinyals O, Warden P, Wattenberg M, Wicke M, Yu Y, Zheng X (2015) TensorFlow: large-scale machine learning on heterogeneous systems. Software available from tensorflow.org https://www.tensorflow.org/
29. Chollet F, et al. (2015) Keras. https://keras.io
30. Jin H, Chollet F, Song Q, Hu X (2023) Autokeras: an automl library for deep learning. J Mach Learn Res 24(6):1–6
31. Kiefer J, Wolfowitz J (1952) Stochastic estimation of the maximum of a regression function. Ann Math Stat 23(3):462–466. https://doi.org/10.1214/aoms/1177729392
32. Kingma DP, Ba J (2014) Adam: a method for stochastic optimization. arXiv https://doi.org/10.48550/ARXIV.1412.6980
33. Loshchilov I, Hutter F (2017) Decoupled weight decay regularization. arXiv https://doi.org/10.48550/ARXIV.1711.05101
34. Fu Z, Mo J, Han F, Gong P (2012) Tool path correction algorithm for single-point incremental forming of sheet metal. Int J Adv Manuf Technol 64(9–12):1239–1248. https://doi.org/10.1007/s00170-012-4082-7
35. Herrmann LR (1976) Laplacian-isoparametric grid generation scheme. J Eng Mech Div 102(5):749–756. https://doi.org/10.1061/jmcea3.0002158
36. Möllensiep D, Schäfer J, Pasch F et al (2024) Cluster analysis for systematic database extension to improve machine learning performance in double-sided incremental sheet forming. Int J Adv Manuf Technol. https://doi.org/10.1007/s00170-024-14014-8