Parameterized data handling for forming tool tryout: reverse engineering, data consolidation and springback compensation

L Maier*, C Hartmann, W Volk
Chair of Metal Forming and Casting, Technical University of Munich, Walther-Meißner-Straße 4, 85748 Garching, Germany
*lorenz.maier@utg.de

Abstract. The forming tool design process generates large amounts of data up to the first falling parts. On the one hand, simulation results, geometric measurements and design models originate from different software tools, which leads to a non-consolidated set of data inventory. On the other hand, the total data volume is hard to handle economically. A stable and user-friendly data structure for overarching tool tryout is missing. Often several experience-based iterations are necessary to derive the tool’s working surfaces, which is both time- and resource consuming and even may lead to postponed start of production. Meanwhile, early-generated data does not involve into the manual optimization process. Therefore, in this paper a parameterized data handling methodology is introduced, which enables systematic reverse engineering, data consolidation, and springback compensation. Each generated dataset during tryout is traced back to a mathematical description of geometry using so called control points (B-Spline model). Through, the parameterized description, the different data formats interact straightforward and need minimum storage. The developed concept is demonstrated for the springback analysis of a forming component using design models and numerical data.

1. Introduction
In the automotive sector, sheet metal forming is one of the most important and widespread manufacturing processes. This process is especially indispensable in the production of body components such as door panels, engine hoods or side impact protection strips [1,2]. For these type of products, geometric tolerances are tight and tooling is expensive. Consequently, during the tryout process, tools have to be redesigned so that this targeted geometry is achieved within defined tolerances [3,4]. Therefore, an essential factor is the compensation of springback, which occurs after plastic deformation due to the simultaneous buildup of elastic energy. It is especially important when advanced high-strength steels (AHSS) are considered [5]. Since these materials offer a soaring savings potential in terms of weight and costs for their high strengths, a solution to the springback compensation problem is being sought with renewed vigor. Because numerical prediction of compensated tool geometry requires elaborate material models, expensive software and computational capacity, compensation is recurrently done by machining of forming tools. This process, which is customary in the industry, involves employee expertise and often leads to considerable cost and time overruns in the tightly scheduled tryout and production process [6].

To support this iterative knowledge-based process, it is useful to implement a compensation approach for comparing both discrete part data such as finite element meshes (.stl) and unorganized point clouds (.xyz) with continuous CAD files (.igs or .stp). Therefore, reverse engineering (RE) for discrete data is
state of the art [7-9]. This makes it feasible to design discrete data continuously and to transfer them back into a CAD program by using e.g. B-Splines [10].

The aim of this work is to propose a way for data reduction through RE. Afterwards a compensation valuation is provided, which utilizes these continuous data in order to achieve the components target geometry. A control point-based compensation (CPC) approach is chosen to represent the component geometry through B-Splines. The advantage of this procedure compared to discrete data results from easy data-handling and the transferability of this data to CAD programs.

2. State of the art
In order to achieve the goal of this work, the discrete data are first converted into continuous data by RE. Subsequently, the deviations are compensated with CPC and new tool working surfaces are derived from this compensation.

2.1. Reverse engineering
One way of representing surfaces and curves are the B-Splines. Due to their continuity and use in commercial CAD programs, they are widely used in surface reconstruction and the resulting RE [8]. B-Splines are a linear combination of piecewise polynomial basis functions. This gives them a high degree of flexibility. A B-Spline is defined by a knot vector, control points and the degree of the curve. A knot vector is a non-decreasing vector in the parameter space and written $U = \{u_0, u_1, \ldots, u_{n+p+1}\}$, where $u_i$ is the $i$-th knot, $p$ is the polynomial degree and $n$ the number of basis functions [10]. The $i$-th basis function of degree $p$, written as $N_{i,p}(u)$, is defined recursively as follows: beginning with the basis function

$$N_{i,0}(u) = \begin{cases} 1 & \text{if } u_i \leq u < u_{i+1} \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

for $p = 0$ and followed for $p > 0$ the basis functions are defined by the Cox-de Boor recursion:

$$N_{i,p}(u) = \frac{u - u_i}{u_{i+p} - u_i} N_{i,p-1}(u) + \frac{u_{i+p+1} - u}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(u) \quad (2)$$

In the $d$-dimension, a B-Spline curve in the dimension $\mathbb{R}^d$ with the control points $P \in \mathbb{R}^d$ is defined as follows

$$C(u) = \sum_{i=1}^{n} P_i N_{i,p}(u) \quad (3)$$

A piecewise linear interpolation between the control points bounds the control polygon. Continuous curves and surfaces can be represented by this description. In the case of a B-Spline surface, both an additional knot vector and a Cox-de Boor recursion in the direction of this knot vector are introduced. Due to the widespread description of these surfaces in CAD programs it is therefore suitable as a RE approach [12].

2.2. Springback compensation
There are several approaches to compensate for the deviation between the first falling components and the desired target geometry [3,4,7]. The principal procedure of this iterative process is shown in Figure 1. For example, the displacement adjustment method (DA) by Gan and Wagoner has proven to be effective [13]. The component is measured and the deviation between the actual and target geometry in punch travel direction is observed. In the next step, this deviation is taken into account by a reducing factor (further mentioned as damping factor $\alpha$) in the opposite direction, which results in a new working surface. Repeating this procedure yields an iterative compensation process. Another displacement-driven method utilizes the normal direction for measuring the deviation [14]. Here, the arc length of the cross section is changed and it is therefore only suitable for compensation to a limited extent [15]. To circumvent this error, a third approach was proposed. The so-called material point tracking uses defined mesh points and considers the occurring displacement of these points during the springback [5].
A method that compensates the deviation independently of material points is the displacement-compatible spring-forward method [6]. Here, the compatible stresses causing deviations are estimated and then applied to the original nominal geometry. After stress relaxation, the result is an updated geometry of whose surfaces represent the compensated working surfaces.

3. Approach

The following section describes the approach adopted in this work to enable RE and to perform deviation compensation according to the resulting control point-based system.

3.1. Reverse engineering and data consolidation

As a RE approach the representation of a 2D geometry by a B-Spline is chosen. This simplifies the presentability without loss of generality. For this purpose, the target geometry is loaded as a .stp file and tessellated by a tolerance of $1 \times 10^{-4}$ mm. In the next step, the component top and bottom are separated from each other automatically. Subsequently, corresponding to the plane stress condition in sheet forming, only the component top is considered. After successful separation, the tessellated points are sorted pursuant to the arc length by a traveling salesman problem algorithm. The resulting path is subdivided homogeneously, the arc length in the cross section to be examined is discretized by $j$ ($j \gg n$) points and afterwards represented by an approximation approach using a B-Spline according to Piegl [11]. This procedure generates the B-Spline with an associated knot vector $U_{\text{target}} = \{u_1, u_2, \ldots, u_{n+p+1}\}$, the polynomial degree $p$ and $n$ control points $P = \{P_1, P_2, \ldots, P_n\}$ as the target geometry according to equations (1), (2) and (3). Figure 2 shows the target B-Spline with $n = 35$ control points and a polynomial degree of $p = 3$ in blue color.

Now the current geometry from simulation or measurement, the discrete data, is used to perform a B-Spline approximation. The .stl data is also discretized and refined by $j$ points after separating and sorting them. To ensure comparability of the curves, the knot vector $U_{\text{target}}$ of the target geometry acts as a constraint and stays constant in the approximation of the current geometry. The implementation of this constraint is essential for the consolidated representation and subsequent compensation. Therefore, the approximation algorithm by Piegl was adjusted. The knot vector $U$ as well as the polynomial degree $p$ and the number $n$ of control points from the target geometry are equal, which ensures comparability of the B-Splines. Hence, the only difference between actual and target B-Spline is the control points.
position. This RE simplifies the transferability of B-Splines to CAD programs and consequently the development of a tool rework strategy.

3.2. Springback compensation

The basic procedure for this work is as shown in Figure 1. The goal of it is to equate the control point positions of the current geometry after springback with the target geometry control points. To achieve this, the tool working surfaces have to be changed. Similar to the DA, the displacement is considered, to be more precise in this case the shift of the control points. The resulting vector between target control point $P_i$ and current geometry control point $P_i'$ is inverted and applied via an attenuation factor. As common, to avoid overshoot, a damping factor $\alpha$ is introduced. This shortens the vector length without changing direction. In Figure 2 the shift for one control point ($P_3$) is shown schematically, although the procedure is applied to all control points simultaneously. This process creates a new B-Spline curve that can be returned into the simulation as a tool working surface in the iterative design process from Figure 1. This may be continued with until the component tolerances are satisfied.

4. Results

The proposed approach is to be investigated by simulation. For this purpose, a test geometry shown in Figure 2 has been developed that allows compensation without undercuts. Therefore, both radii $r = 6$ mm, which are common in forming technology, and surface transitions constructed via B-Splines are incorporated into the target geometry. The component was designed with the CAD program CATIA V5-6 (release 2014). The geometry shown is extruded by 30 mm in x-direction.

4.1. Reverse engineering

During RE the control point model is computed for each geometry, as described. Initially, control points were added until the geometry was satisfactorily represented by the curve. It was deliberately decided not to add control points only in the area of the radii, since these could lead to geometric complications, e.g. shifts in the wrong coordinate direction in the plane area during the course of compensation. Accordingly, the control polygon was subdivided homogeneously. This led to a target curve with $n = 70$ control points $P$. The curves polynomial degree is $p = 3$. This allowed the curve and the radii to be approximated without oscillations. Increasing the degree $p$ resulted in high deviations in the straight parts of the specimen between the control points.
4.2. Springback compensation

The initial situation for springback compensation is a simulation with AutoForm Forming R8 software. The standard material model of a DP800, available in the software, was utilized. Figure 3 shows the deviation of the non-compensated geometry with springback from the nominal geometry in the false color image. Deviation between actual and nominal geometry for this illustration was calculated using the length of the vector between both parts in the normal direction. Graphics for the deviation analysis were created with the software GOM Inspect. To introduce another error measure the deviation was calculated over more than 4000 points in the z-direction. This results in an average deviation of $\Delta_z = 0.7573$ mm for the initial problem. As the figure shows, both the drawing depth the angle of rebound in the upper part of the component have a high error level.

Based on this deviation, the CPC algorithm described above is applied. First, a damping factor of $\alpha = 0.5$ is examined. The methodology is performed for eight iterations. It was found that the deviation increases sharply at first and falls below the initial problem only from the second iteration on. With the eighth iteration, the total deviation in z-direction could be minimized to $\Delta_z = 0.0748$ mm. Figure 4 shows the deviation in normal direction from the target geometry in the false color image. For better illustration, the limits of the color bars have been redefined.

The influence of the damping factor $\alpha$ on the number of iteration loops and the initial problem was investigated with $\alpha = 0.8$. It was found that, as expected, the initial problem remains and the number of required iterations decreases until the mean deviation falls below the limit of 0.10 mm (cf. Figure 8). Thus, after seven iterations, a deviation in z-direction of $\Delta_z = 0.0705$ mm could be determined. It was noticed that the deviation increased...
The false color image in Figure 5 indicates the deviation after seven CPC iterations from the nominal geometry.

In order to compare these results with an established procedure, the deviation compensation integrated in AutoForm with a compensation factor of 0.8 was carried out. In contrast to the CPC, the deviation is minimized from the first iteration onwards. After four iterations, a deviation of $\Delta z = 0.0877$ mm was achieved. In four further iterations, this result could no longer be improved. On the contrary, the deviation increased again.

The result of iteration four is demonstrated in Figure 6.

In Figure 7, the deviation in the z-direction is evaluated at 16 different points. The target geometry in this area is also shown in the figure. There is no significant disparity between the different compensation strategies in terms of geometric properties. All approaches have approximately the same deviation in the range from $y = 8$ mm to $y = 20$ mm. The same also applies in the lower area of the radius $r = 6$ mm. The figure shows that, regardless of the geometric feature, the new approach achieved compensation as good as the incumbent program. It is worth mentioning that for all three, the transition area in the region of the B-Spline construction ($-10$ mm $< y < 8$ mm) hardly shows any deviation. Nevertheless, the highest point deviation has the proposed solution of AutoForm.

The results of the z-deviation are shown in Figure 8 for each of eight iterations for the three accomplished compensation methods. The already mentioned overshooting problem of the CPC approaches can be clearly seen between initial problem and iteration one. After this initial phenomenon, the deviation for a damping factor of $\alpha = 0.5$ falls almost continuously until we stop the procedure after eight iterations. On this point, the procedure has achieved better deviation compensation globally by again from the fifth to the sixth iteration.
$\Delta_{AF-CPC0.5} = 0.0129$ mm than the compensation method implemented in *AutoForm*. The fluctuations of the CPC with $\alpha = 0.8$ are higher. However, this method already achieves a better global result after seven iterations than the *AutoForm* compensation. The difference between them is $\Delta_{AF-CPC0.8} = 0.0172$ mm.

### 4.3. Data consolidation

A massive advantage of the approach presented is the data saving. Due to the continuous description via the mathematical B-Spline representation, the required storage space is significantly reduced. To describe the target geometry, the knot vector $U$ as well as the control points $P$ in $\mathbb{R}^2$ and the degree were defined. For each further curve, these entries can be reutilized and only the new control points $P$ for the tool geometry and the current geometry have to be stored.

Table 1 lists the memory required for a double precision with eight bytes per number. In comparison, the storage demand for the mesh of the current specimen of the *AutoForm* simulation is shown. Without saving the derived tool geometry, the summed up requirement is already higher by a factor of 148.5 after eight iterations.

#### Table 1. Total memory required in kB per iteration for CPC approach and *AutoForm*.stl output.

|        | $i_0$ | $i_1$ | $i_2$ | $i_3$ | $i_4$ | $i_5$ | $i_6$ | $i_7$ | $i_8$ | $\Sigma$ |
|--------|-------|-------|-------|-------|-------|-------|-------|-------|-------|----------|
| CPC    | 3.40  | 2.24  | 2.24  | 2.24  | 2.24  | 2.24  | 2.24  | 2.24  | 2.24  | 21.32    |
| *AutoForm* | 342.0 | 353.0 | 349.0 | 351.0 | 360.0 | 349.0 | 361.0 | 356.0 | 345.0 | 3166.0   |

### 5. Discussion

The results presented show that CPC is useful for achieving a production tool geometry, but also raise some questions about the initial value problem compared to other deviation compensations. In the following section the differences to the approach implemented in *AutoForm* are explained and the advantages of CPC are described.

Figure 4 as well as Figure 5 show that the CPC approach with control points is suitable to compensate for springback. Compared to the initial situation from Figure 3 without compensation, the target geometry was achieved with a tolerance of $\Delta_z < 0.1$ mm for all approaches. The best result was achieved with CPC using a damping factor of $\alpha = 0.8$ within seven iterations. For this method a mean deviation of $\Delta_z = 0.0705$ mm was achieved. Followed by the CPC approach with $\alpha = 0.5$ and $\Delta_z = 0.0748$ mm for eight iterations. The strategy implemented by *AutoForm* requires only four iterations to achieve a difference of $\Delta_z = 0.0877$ mm between target and the best result for the geometry within eight iterations.
Figure 8 shows that the deviation of the CPC initially increased. Only after the second iteration was it reduced again and moved below the initial value. A change in the damping factor, as demonstrated in the plot, has a direct effect on this, but also affects the number of iterations required to achieve the best result in each case. What causes this effect must be investigated further. What is already apparent in Figure 2 becomes visible in Table 1. The memory requirement due to the control point representation decreases significantly compared to the memory required to store the respective meshes. Here, the required volume could be reduced by a factor of 148.5. This makes it feasible to compare several components simultaneously. Particularly when considering unorganized point clouds, this type of representation can massively reduce memory requirements and thus allow larger components to be calculated more quickly.

6. Conclusion
This investigation has shown that the method produces more than acceptable results and reduces the amount of memory required. If further investigations solve the initial problem of the CPC method, it can be assumed that the number of iterations required will also decrease. This and the applicability to measured data show the achievable possibilities by this method. For example, purely measurement series can be used for compensation and the experience-based, industry-related tool tryout process can be supported. And, last but not least, the transferability of the B-Splines to CAD programs simplifies the development of a tool rework strategy.

Acknowledgment
The authors would like to thank the European Research Association for Sheet Metal Processing (Europäische Forschungsgesellschaft für Blechverarbeitung e.V.) for supporting this research by funding the research project IGF 21467N - EFB 10/218.

References
[1] Livatyali H and Ergeldi M M 2006 Journal of Materials Processing Technology 173 14.
[2] Lingbeek R, Huétink J, Ohnimus S, Petzoldt M and Weiher J 2005 Journal of Materials Processing Technology 169 115.
[3] Hartmann C, Eder M, Opritescu D, Maier D, Sautaella M and Volk W 2018 Journal of Materials Processing Technology 261 140.
[4] Hartmann C, Eder M, Opritescu D and Volk W 2017 Procedia Engineering 207 466.
[5] Maier D, Hartmann C, Till M, Büdenbender C, Behrens B A and Volk W 2019 Key Engineering Materials 794 277.
[6] Scandola L, Büdenbender C, Till M, Maier D, Ott M, Behrens B-A and Volk W 2021 International Journal of Material Forming.
[7] Maia A, Ferreira E, Oliveira M C, Menezes L F and Andrade-Campos A 2017 Computational Methods and Production Engineering (Elsevier) pp 51–82.
[8] Bazilevs Y, Calo V M, Cottrell J A, Evans J A, Hughes TJR, Lipton S, Scott M A and Sederberg T W 2010 Computer Methods in Applied Mechanics and Engineering 199 229.
[9] Wang H, Zhou J, Zhao T and Tao Y 2016 The International Journal of Advanced Manufacturing Technology 85 1187.
[10] Dan J and Lancheng W 2006 The International Journal of Advanced Manufacturing Technology 31 574.
[11] Piegl L and Tiller W 1997 The NURBS book, 2nd Edition (Berlin: Springer).
[12] Greco F, Coxx L, Maurin F, Balla V K, Deckers E and Desmet W 2018 Computers & Structures 201 15.
[13] Gan W and Wagoner R H 2004 International Journal of Mechanical Sciences 46 1097.
[14] Mole N, Cafuta G and Štok B 2014 Journal of Materials Processing Technology 214 1673.
[15] Birkert A, Hartmann B and Straub M 2017 Journal of Physics: Conference Series 896 12067.