



DEVELOPMENT OF A SUPPORT TOOL TO PREDICT MANUFACTURABILITY OF DEEP-DRAWING PARTS

D. Goller, S. Schreyer, B. Alber-Laukant, F. Rieg and W. Volk

Keywords: process analysis, deep drawing, geometric modeling kernel, feasibility analysis

1. Introduction

There are several different approaches to develop new products. A popular example is the product development process (PDP) by Pahl/Beitz [Feldhusen and Grote 2013]. The development engineer always benefits from detailed knowledge from later stages of the PDP, like process knowledge. Most likely this information is not available at these early stages, so it could be transferred from similar cases, which already have completed this stage, or the information can be generated by fast but preferably accurate estimations.

The main goal of the collaborative research centre FORPRO², supported by the Bavarian research foundation (Bayerische Forschungsstiftung - BFS), is the optimization of simulation based development by knowledge transfer. One aspect of before mentioned optimization is the development of a software tool to predict the manufacturability of deep drawing parts.

This can be achieved by process simulation by means of finite element analysis, but getting reliable output is usually extremely time-consuming and requires detailed process knowledge. Especially in early stages of the development process, a fast prediction on basis of non-detailed data is more expedient. There is already an approach to create a fast assumption of the manufacturability by using One-Step-solvers, which are already available in commercial solutions as for instance Autoform-OneStepforCatia [AutoForm 2015] and Altair HyperForm [Altair Engineering Inc. 2015]. These tools compute estimations regarding manufacturability by simplifying the physics of the forming operation. Examples are the assumption of linear coherences or simplifying of contact conditions. The results are generated by calculating from the final deformed configuration to the initial undeformed configuration. The accuracy of this method however varies from case to case [Tekkaya 2000].

The developed method differs from this approach. The idea is to only use the geometry information of the deformed configuration as input for the algorithm to rate the different areas of the part according to the suitability for the manufacturing process. This not only benefits the complexity and calculation effort of the algorithm but also simplifies the usability and shortens the learning time for users. Furthermore with an integrated geometric based analysis, a robust geometry in terms of feasibility can be defined in an early stage of the PDP. Thereby less re-design iterations are necessary in comparison to an exclusive simulation based approach, which results in saving of time (Figure 1). The whole procedure is divided into three processes. First the geometry gets separated into widely independent areas (following referred to as patches), then these areas are rated on the basis of geometric properties. After that the boundaries have to be analysed to determine dependencies among each other.

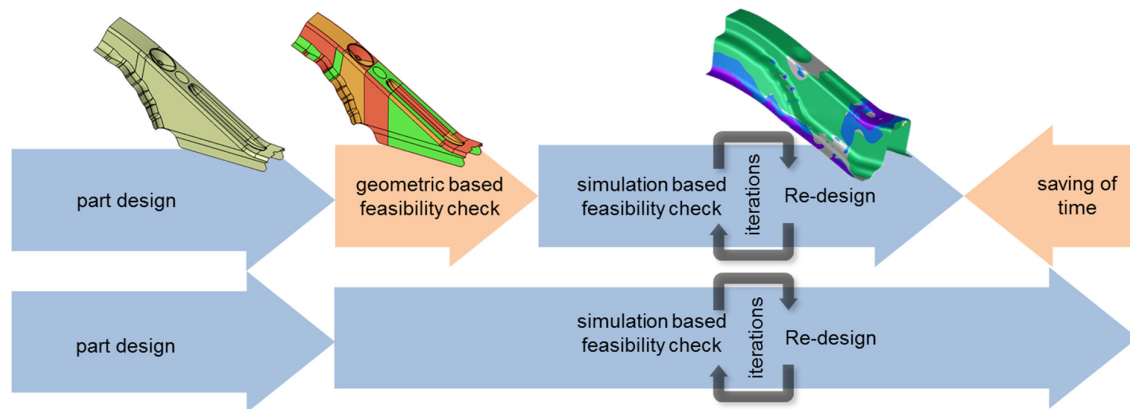


Figure 1. Acceleration of PDP by integration of geometric feasibility check

This paper describes the first part of developing the support tool, which is determining the theoretical approach, specification of the application, selection of the necessary libraries, implementation of the algorithms for separation and rating, implementation of the user interface and verification of the implementation.

2. Theoretical approach

There are three main quality criteria in the PDP of sheet metal parts. The first one is size accuracy, which is mainly affected by the spring back of the deep drawing part in the demoulding phase of the process. Secondly there is the possibility of surface defects, for instance die marking lines. The last defect is insufficient functional properties, for example as a result of excessive thinning [Hoffmann et al. 2012]. But there is one failure, which strongly depends on the geometry of the part: wrinkling of the 2nd order [Wang et al. 2000]. Due to the fact, that the aim of the method described in this paper is to evaluate the feasibility of sheet metal parts only based on geometric information of the part, only the failure of wrinkling is taken into account. Even if the other defect types are not evaluated with the described method, time consumption in the PDP is expected due to the effect that failures interact. By minimising the wrinkling tendency by a more robust part geometry, the process window is enlarged and less re-design cycles are necessary to define a feasible solution. This becomes particularly obvious in the case of the interaction of 2nd order wrinkles and splits. A higher restraining force in the flange reduce wrinkling tendency in the side wall, but also causes a higher thinning and risk of splits in the whole deep drawing geometry. Conversely, a more robust part geometry enables lower restraining forces and so even a lower risk of splits.

As mentioned before, wrinkling is affected by the geometrical design of the sheet metal part. Wrinkling is caused by compressive stresses, whereas the stress location and magnitude mainly corresponds to the gradient of unfolding length and therefore the geometry of the part. A high sheet thickness and high local curvature as further geometric features counteract the wrinkling tendency [Hutchinson and Neale 1985]. Beside these geometric factors, also material properties like the young's modulus and process values like the binder force affect the wrinkling initiation, which is simplified in this evaluation method due to fuzzy data in this early design stage of the sheet metal part [Birkert et al. 2013]. As the aim of the method is not to substitute the more precise feasibility evaluation achieved by the finite element simulation but to create a better basis for further method planing, this simplification is tolerable.

Research efforts to investigate wrinkling in sheet metal forming can be subdivided in analytical, experimental and numerical approaches. Analytical analyses enable to determine the influence of various parameters on wrinkling in a negligible computational time, but are limited to simple geometries and boundary conditions, which makes it difficult to apply it to 3-D sheet metal forming [Wang and Cao 2000]. Numerical simulations overcome this limitation, but the sensitivity of the model parameters has to be considered precisely to generate reliable information, like location and temporal course of compressive stresses. Within the method presented in this paper an experimental approach of Yoshida is taken as basis for the wrinkling evaluation [Yoshida et al. 1981]. Many researchers modified this

specimen and its testing procedure to expand the investigation possibilities, like considering tool surfaces as an impact factor on wrinkling [Cao et al. 2007]. Nevertheless, only flat surfaces are investigated in these experimental analyses. With the adaption to a sheet metal testing machine, available at the chair for forming and casting, also curved surfaces are able to be investigated by changing the punch of the testing machine. The result of this modified yoshida test with a coupled simulation based evaluation is the wrinkling initiation depending on sheet thickness, surface curvature and compressive stress (Figure 2, left). Last-named impact factor can be related to the gradient of unfolding length for simple part geometries and thereby reduced to pure geometric information. Therefore an investigation of reference geometry is necessary. For complex drawing geometries, an OneStep simulation is needed to receive the information of the stress state (Figure 2, right). If the combination of these wrinkling parameters reaches critical values within one patch and wrinkling might occur, it can be visualized to the development engineer by colouring the corresponding patch and depositing information how to prevent wrinkling locally.

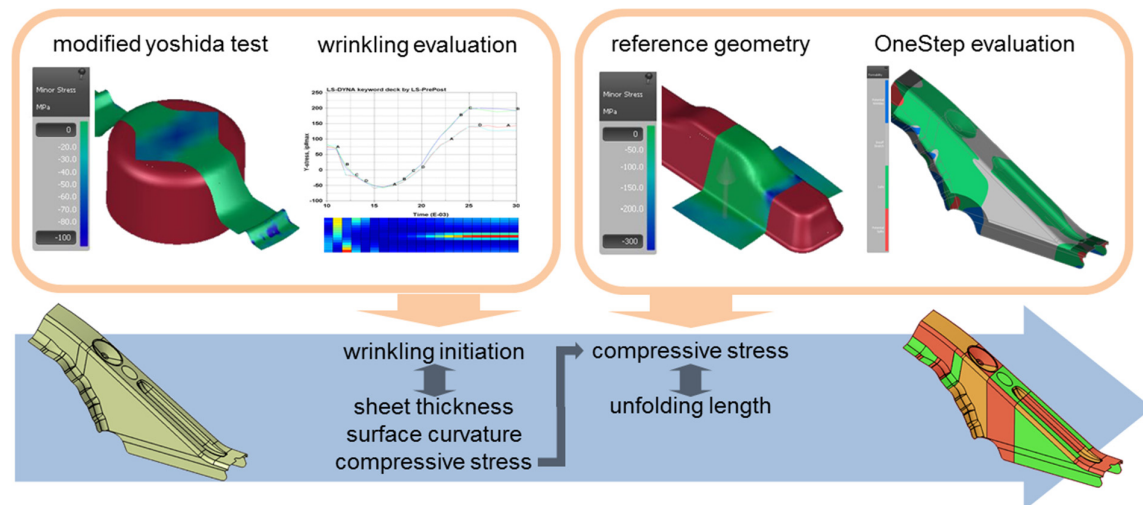


Figure 2. Concept of geometric based feasibility analysis of sheet metal parts

The second main task to evolve the method of supporting the development engineer in an early phase of the PDP with feasibility information is to determine these geometric parameters of a specific sheet metal part and connect the generated process information with the geometry of the part. In order to achieve that task, a software tool has to be developed.

3. Development of software tool

The method is based on simple and reliable algorithms derived from theoretical considerations (see part two), but the implementation is a bit of a challenge, as lots of choices have to be made. A graphic file format has to be chosen to import the geometry, a graphics kernel can be used to provide the underlying model structures and useful functions and last but not least there is need of a widget set, to provide the user with a graphical user interface to receive inputs and provide the user with outputs. The output has to be optimised to lead to fast conclusions without the need of the user to capture, filter and analyse complex information masses. Nevertheless, too many third party libraries increase the number of dependencies which often lead to increased effort to set up the system.

It was defined that the theoretical approach should be implemented in a neutral solution. This means there should not be proprietary software with fee-based license policy included or used in the tool.

3.1 Choosing file format and necessary tools

There are several common neutral geometry file formats with different advantages for both, parametric and tessellated geometry data. Examples are STEP (Standard for the Exchange of Product model data), IGES (Initial Graphics Exchange Specification), JT (Jupiter Tessellation) and ACIS. This set of options

was weighted and rated based on criteria like parameter support, documentation detail, standardisation and support in graphic kernel interfaces. The selected format is STEP, among other things because of its magnitude of storable information, including parameters, its standardisation and widespread distribution [Goller 2014].

The geometric modelling kernel used in this project is OpenCASCADE. It is very extensive and is used as backbone of a lot of commercial and free products like finite element analysis systems (FEA) like Salome [OPEN CASCADE SAS 2015] and applications for computer aided design (CAD) like FreeCAD [Freecadweb 2015].

3.2 Implementation

First step of the process is the import of the geometry information. This is provided in form of a parametric model read from a STEP file. The individual entities of the structure, described by mathematical terms like cartesian point, vector, circle or cylindrical surface [ISO 2013] are converted to entities provided by OpenCASCADE, meaning the internal data format and stored in memory. After input, the geometry model is displayed in the GUI and can be inspected and manipulated by the user. Following a plane is created (henceforth referred to as incremental plane), which is defined by a random point contained in the model and a direction (normal vector of incremental plane) defined by the user. This direction should represent the main geometry direction and has to be perpendicular to the deep drawing direction. It is either provided by a vector, entered component-by-component, or by the normal vector of another plane (contained in the model), which is chosen by the user. Either case leads to a plane, which can be intersected with the model to retrieve the intersection curve. For lacking process knowledge this direction can be estimated in future versions. After intersection, the plane gets translated in normal direction with a constant increment and intersected with the model again. This will be repeated until the length of the resulting curve is zero. Then the edge of the model is reached and the plane is set to the original position to get the intersection curves of the other part with the very same procedure (see Figure 3). The gathered intersection length values plotted over the part length in normal direction of the incremental plane, compose the intersection curve. Differentiation of this curve implies the intersection gradient, the foundation of further considerations. Figure 4 shows qualitatively the intersection curve and intersection gradient of an exemplary part.

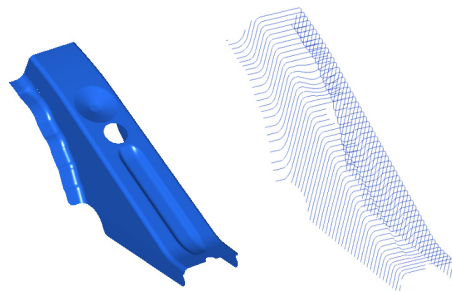


Figure 3. Model and its intersection curves

To specify the patch boundaries, the intersection gradient is analysed. If this curve crosses the edge of a predefined range in positive or negative direction (determined in process shown in Figure 2), a patch boundary is created. The same happens if the intersection gradient is out of the range and re-enters it. During development it is experimented and tested with different absolute boundary values. The final values (either constant or as a function of geometry and process parameters) that have to be used for real geometries are determined by research partners (see chapter 2).

A new patch emerges by dividing an existing patch or face from import geometry into two new independent faces. This is done with another intersection. The incremental plane is translated along its normal vector to its respective position and intersected with concerned faces of the model. The intersection curve of each of these faces is added to it as a new boundary, thus it is split into two. The geometry is updated to maintain the new face configuration for all further steps. This is repeated for every position with over- or understepping intersection gradient.

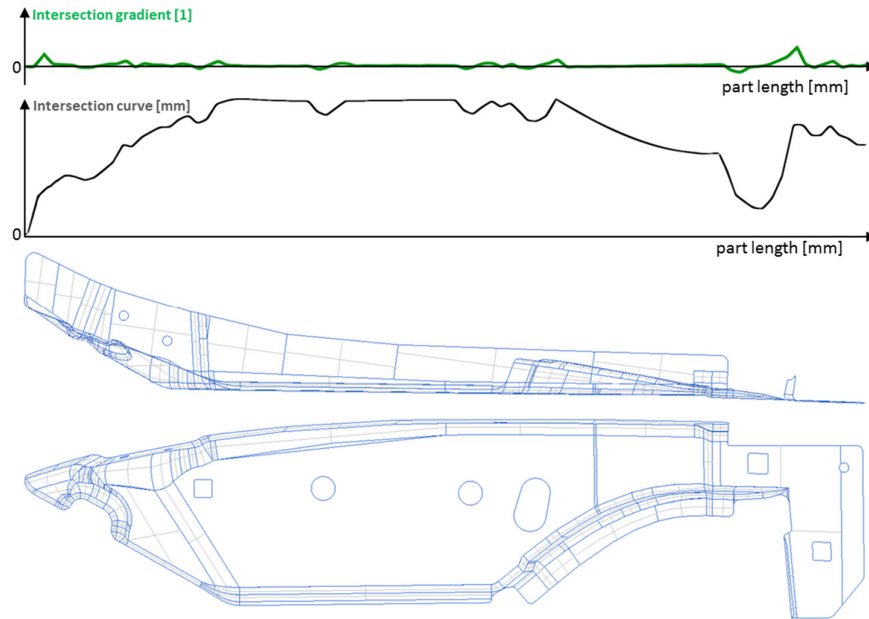


Figure 4. Calculated intersection curve and intersection gradient of random example part

After this procedure the geometry is completely divided into patches, which now can be rated separately. The rating value calculated from mean intersection gradient and other quantities of the patch, like curvature values (for example Gaussian curvature), is connected to the face. On demand this scalar quantity can be displayed as a colour scale over the geometric representation. The normalized colour scale ranges from green (relatively uncritical for manufacturing) over orange to red (relatively critical). The normalized, relative scale can be expanded to an absolute scale, if the boundary values are defined universally, independent from part and process. By means of this rendering, the user is able to get an overview of the manufacturability of the whole part within seconds.

Another example part is shown in Figure 5. On the left, there is a shading view of the concerning part. In the middle, there is a wireframe view of the original faces and on the right, there is a wireframe view of the new faces. All original faces are obtained and some of them (see middle and rear section of the part) are cut parallel to incremental plane.

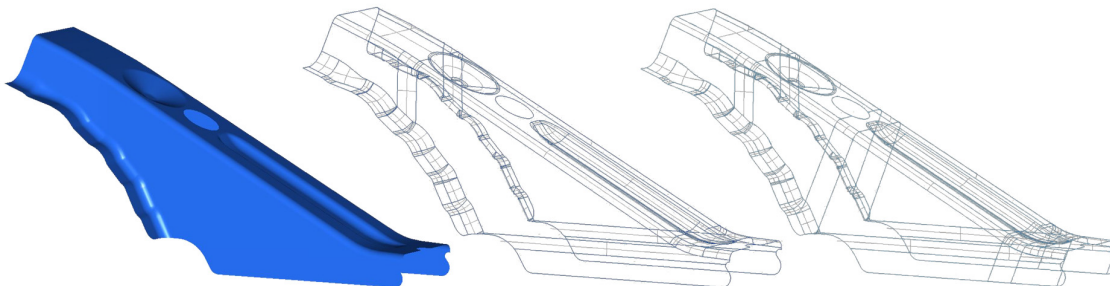


Figure 5. Faces before and after patch generation

3.3 Verification of algorithms

The different calculation routines implemented in this method have to be tested and verified to ensure the functionality. Besides real geometry files of industrial partners test models have been created in order to analyse the behaviour of the algorithm relating to different aspects.

Two main test geometries were created to verify the following aspects.

3.3.1 Intersection position

During development it is necessary to constantly check whether the cutting position for different boundary values is correct. This was achieved by designing a test geometry containing an edge which follows a compounded second order polynomial curve (see Figure 6). The structure is compounded of a polynomial with a positive slope (area I), which causes a linearly increasing intersection gradient and a polynomial with a negative slope (area II), which causes a linearly decreasing intersection gradient. These parts are connected in x_0 and are of equal length (a).

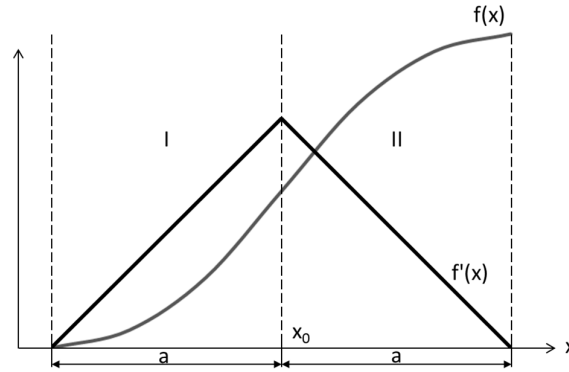


Figure 6. Qualitative profile of edge shape $f(x)$ and its derivation $f'(x)$

The equation of the compounded polynomial, dependant of stretch factor m , the span a and the x -location x_0 is the following:

$$f(x) = \begin{cases} m(x - x_0 + a)^2, & \text{for } x \leq x_0 \\ -m(x - x_0 - a)^2 + 2ma^2, & \text{for } x > x_0 \end{cases} \quad (1)$$

This leads to the differentiation with the constants m , x_0 and a :

$$f'(x) = \begin{cases} 2m(x - x_0 + a), & \text{for } x \leq x_0 \\ -2m(x - x_0 - a), & \text{for } x > x_0 \end{cases} \quad (2)$$

So, if the boundary value is b , the cutting position x as a function of b is:

$$x(b) = \begin{cases} x_0 - a + \frac{b}{2m}, & \text{for } x \leq x_0 \\ x_0 + a - \frac{b}{2m}, & \text{for } x > x_0 \end{cases} \quad (3)$$

The size of the boundary value b , that can be examined, is a function of a and m , because the constraints for minimal x values $x(b) > x_0 - a$ and for maximum x values $x(b) > x_0$ for the first part of the curve (area I). This leads to the range of b dependant on a and m :

$$\text{Minimum: } b = 0 \quad (4)$$

$$\text{Maximum: } b = 2ma \quad (5)$$

With the defined edge it is possible to manually calculate the cutting position very easily to examine the behaviour of the algorithm for different boundary values between 0 and $2ma$. In the resulting shape, the described pattern is mirrored after the end of area II. The equations (1) - (5) can be transferred to the mirrored part, which leads to a total range from $-2ma$ to $+2ma$ that can be examined.

3.3.2 Discretisation

Another test geometry is shown in Figure 7. It is compounded of a straight rectangular part, followed by an increasing part with constant incline. This structure is repeated once. The constantly increasing part causes an intersection gradient of zero, but both junctions cause a non-zero gradient (see Figure 7). This example was selected because for a human it is obvious where the patch boundaries have to be created for a very wide range of boundary values that is in each case at position of the great straight decline. Unlike the first test geometry, this one is not suitable to test different boundary values and their resulting cutting positions, but to analyse the deviation of the actual cutting position from the theoretically exact position.

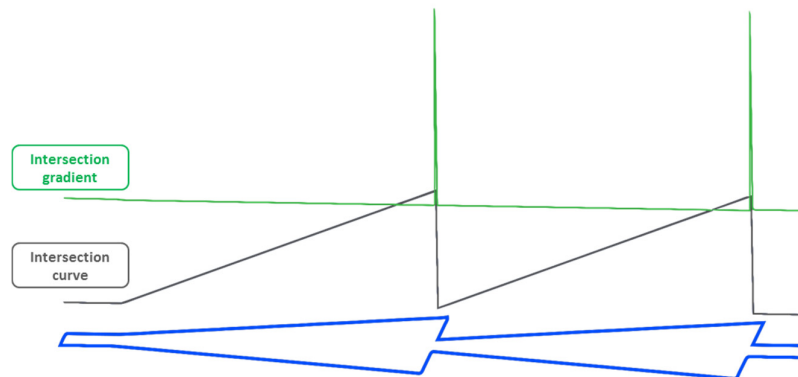


Figure 7. Geometry to test discretisation coherences

The cutting position however, is not exactly located as expected. This is due to the discretisation. The intersection curve and therefore the intersection gradient cannot be created continuously, because the computation of every intersection between incremental plane and part costs time and there should only be made as many intersections as necessary. Hence the resulting cutting position is the nearest intersection at the exact position.

There are two possibilities to increase the accuracy and minimize the deviation caused by discretisation. The first one is to abolish the constant increment and make it variable. If this value is calculated after every incrementation dependant of the intersection gradient, it is possible to reach an optimal compromise between accuracy and number of intersections. The desired exactness respectively resolution can be affected with parameters of the equation that is used to compute the increment value. Another influence is the calculation of the intersection gradient. Because of the discretisation, the intersection curve (and therefore also the gradient) is described by a polygonal chain. In many cases it is possible to get closer to the exact position, if the curves which lead to intersections are interpolated. In some special cases, the user may not want to use the two described algorithm options. In this case it can be deactivated by editing a configuration file.

4. Conclusion

This paper describes the first step of the development of a support tool to predict feasibility of simple geometries in early stages of designing deep drawing geometries. The prediction algorithm bases on processing geometry information which is read from a STEP file. The most important rating criteria taken into account is the intersection gradient, computed by means of a geometric modelling kernel which allows the performance of geometric operations with step geometry and programmatically created planes. These operations lead to sub faces which can be rated on basis of intersection gradient and other criteria like Gaussian curvature. The final rating value is rendered with the model to lead to fast conclusions.

The completed software tool can be used to get fast predictions for simple geometries alongside the first steps of the design process. The usage of frameworks and programming interfaces accelerates the implementation duration, deliberate test geometries ensure reliability and reduce adjustment effort.

5. Next steps

There are several things to be done before it can be used entirely and integrated in work flows.

The most essential of these tasks is the implementation of the rating algorithm. Presently it is already possible to create patches from any parametric geometry imported as STEP, hence the segmentation of the part in many little faces which can be rated as unity. At the present state, a very simple rating algorithm is implemented to be able to develop and work with the system. The rating is solely based on the mean curvature of a face. This algorithm however has to be exchanged by an advanced one, based on quantities like Gaussian curvature and intersection gradient.

If the rating is finished it is planned to take a closer look at patch boundaries to determine dependencies of neighbour patches and influences among each other. If the dependencies appear to be crucial, boundary effects have to be considered during patch generation and patch rating to maintain consistency. Last step of the development will be validation, meaning practicing the application on a variety of different geometries from industrial partners. In some points adjustments will probably have to be made according to test results. At the very end, the benefit compared to conventional methods should be ascertained and quantified.

Acknowledgement

We thank the Bayerische Forschungsstiftung for funding this project as part of the collaborative research centre 'FORPRO² – Research alliance for efficient product and process development by knowledge based simulation'.

References

- Altair Engineering Inc., "AltairHyperForm", Available at <<http://www.altairhyperworks.de/Product,4,HyperForm.aspx>>, 2015.
- AutoForm, "Autoform-OneStep-for-Catia", AutoForm Engineering GmbH, Available at <<http://www.autoform.com/de/produkte/autoform-onestep-for-catia/>>, 2015.
- Birkert, A., Haage, S., Straub, M., "Umformtechnische Herstellung komplexer Karosserieteile - Auslegung von Ziehanlagen", Springer-Verlag Berlin, 2013.
- Cao, J., Cheng, H., Wang, H. P., Wang, C. T., "Buckling of sheet metals in contact with tool surfaces", *Annals of the CIRP*, Vol.56, 2007, pp. 253-256.
- Feldhusen, J., Grote, K.-H., "Pahl/Beitz Konstruktionslehre", 8th edition, Springer-Verlag Berlin, 2013.
- Freecadweb, "FreeCAD", Available at <<http://www.freecadweb.org/>>, 2015.
- Goller, D., "Untersuchung von ausgewählten 3D-CAD-Daten für prozesstechnische Analysen und semantische Verknüpfungen im virtuellen Produktentwicklungsprozess", University Bayreuth, 2014.
- Hoffmann, H., Neugebauer, R., Spur, G., "Handbuch Umformen", Carl Hanser Verlag München, 2012.
- Hutchinson, J. W., Neale, K. W., "Wrinkling of curved thin sheet metal", *Plastic Instability*, Presses Pontset Chaussées, 1985, pp. 71-78.
- ISO, "ISO 10303-214 - Industrial automation systems and integration - product data representation and exchange - Part 214: Application protocol: Core data for automotive mechanical design processes", International Organization for Standardization, 2010.
- OPEN CASCADE SAS, "Salome Platform", Available at <<http://www.salome-platform.org>>, 2015.
- Tekkaya, A., "State-of-the-art of simulation of sheet metal forming", *Journal of Materials Processing Technology*, Vol.103, 2000, pp. 14-22.
- Wang, J., Wu, X., Thomson, P. F., Flitman, A., "A neural networks approach to investigating the geometrical influence on wrinkling in sheet metal forming", *Journal of Materials Processing Technology*, Vol.105, 2000, pp. 215-220.
- Wang, X., Cao, J., "On the prediction of side-wall wrinkling in sheet metal forming processes", *International Journal of Mechanical Sciences*, Vol.42, 2000, pp. 2369-2394.
- Yoshida, K., Hayashi, J., Hirata, M., "Yoshida buckling test", IDDRG, Kyoto, Japan, 1981.

Daniel Goller, Dipl.-Ing.
Universität Bayreuth, Engineering Design and CAD
Universitätsstraße 30, 95447 Bayreuth, Germany
Email: daniel.goller@uni-bayreuth.de