Prod. Eng. Res. Devel. (2013) 7:409–416
DOI 10.1007/s11740-013-0461-7

Fast 3D inverse simulation of hot forging processes via Medial Axis Transformation: an approach for preform estimation in hot die forging

A. Santangelo · P. Blanke · T. Hadifi ·
F.-E. Wolter · B.-A. Behrens

Received: 21 January 2013 / Accepted: 21 March 2013 / Published online: 5 April 2013

Abstract In hot die forging processes, the selection of an ideal preform is of great importance with respect to cavity filling and mechanical load. The common procedure in order to define an adequate preform is the usage of Finite-Element-Analysis (FEA), usually as an iterative process in which various preforms are tested with regard to their suitability. An approach that aims at reducing the number of trials by proposing a first estimation of a suitable preform is presented in this paper. It is conjectured that the material flow paths and resistance can be described by the cavity shape using the Medial Axis Transformation. Based on this, a local inverse material flow for time discrete steps is calculated. The result is a first estimation of an adequate preform shape within a few minutes as an input for further FEA. FE-based parametric design optimization procedure is then presented and compared to the inverse approach, which is identified as a useful complement for the forward simulation technique.

Keywords Hot die forging · Medial Axis Transformation · Preform estimation

A. Santangelo (✉) · T. Hadifi · B.-A. Behrens
Institute of Forming Technology and Machines (IFUM),
Leibniz Universität Hannover, An der Universität 2,
30823 Garbsen, Germany
e-mail: santangelo@ifum.uni-hannover.de

P. Blanke (✉) · F.-E. Wolter
Welfenlab, Division of Computer Graphics,
Leibniz Universität Hannover, Welfengarten 1,
30167 Hanover, Germany
e-mail: blanke@gdv.uni-hannover.de

1 Introduction

In hot die forging, complex shapes are commonly produced in a multiple stage process. The design of the process' intermediate shapes plays a decisive role for the quality of the formed product as well as the required forming load and hence the economic efficiency of the production process. Classically, the selection of appropriate preform and die design is based on experience and supported by Finite Element Analysis (FEA). The forming sequence is usually designed backwards, starting with the given finished parts and going through intermediate shapes to the initial billet. Each forming stage requires several forward simulation cycles since simulation tools do not support a backwards procedure. This trial and error process has a high cost of time and computing resources. Therefore, a software tool that supports the (backwards) design of the process and calculates satisfactory preforms in one step would result in a faster and more economic process design.

Due to the economic impact, many studies deal with the design of preforms pursuing different approaches, most of them in combination with FEA. Various knowledge-based systems have been suggested, which per definition depend on an experience database. Also different methods have been proposed using optimization algorithms that modify process conditions in a systematic iterative manner until an objective is reached. The major drawbacks are computing and time cost. For an overview of the state-of-the-art of the different approaches for the forging preform design, see the work of Yang and Ngaile [1].

A reverse simulation technique combining FE-procedure with the upper-bound method was proposed by Chang and Bramley [2]. The main idea is to find a velocity field that results in a minimum energy requirement for plastic deformation, whereas dies move backwards and boundary areas detach from the die cavity. The velocity field results from the contact condition between material nodes and die surface. The detaching of nodes depends on the distance from contact nodes to the boundary of a target shape (billet). Although the method was successfully implemented on 2D plane strain cases there is scarcely information available about an extension to the 3D-space.

The 3D simulation program presented in the following is a further development of a software prototype for backward determination of the material flow developed by Wienströer [3, 4], which is limited to 2D-simulations. The inverse simulation system aims at a fast, design-integrated determination of a reasonable preform for each forming stage. Based on the calculation of resistance along main material flow paths an adequate preform as a first estimation for further FE-simulation can be achieved by this method. Main paths of material flow are conjectured to be defined by the geometry of the cavity and lie on the Medial Axis (MA, see below) of the forging die cavity. A local measure of flow resistance for points of the MA is then defined, which relies on geometric entities such as the local distance to the cavity surfaces. This local measure is used to define a global measure of geometric resistance by integrating it along shortest paths of the medial graph. This leads to a displacement field which drives the inverse simulation. The model of geometric resistance, the Medial Axis Transformation (MAT) and the displacement graph are the fundamentals of the inverse simulation and are described below.

2 General

2.1 Geometric resistance

In the inverse simulation model it is assumed that the greatest impact on material displacement is given by the die geometry. Further, the intensity of material flow under load depends on the resistance along the possible directions for material flow which are determined by displacement paths. These paths are defined by the cavity of the dies and run parallel to the surface of the forming elements. The possible displacement paths are represented in a graph structure and are calculated by using the MAT. After setting up the graph, a shortest path algorithm is used to calculate the minimal geometry-dependent cost to go from an inner point to points on the graph's boundary which correspond to the free surface of the workpiece, i.e., the surface with no contact to the die. The resulting distribution of geometric resistance values on the boundary is a measure for the ratio of volume displacement at the contact-free workpiece surface.

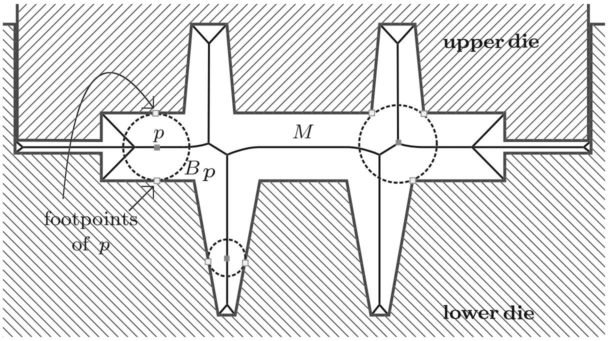

Fig. 1 Cut through upper and lower die. The Medial Axis M of the die gap S is shown with 3 exemplary maximal circles exhibiting 2-point and 3-point contact with the die surfaces ∂S

2.2 Medial Axis Transformation

The MAT is a fundamental geometric construct [5, 6, 8] and an alternative to common representations of geometric objects such as boundary representation (BRep), constructive solid geometry (CSG) or discrete pixelized as well as voxelized representations. A good overview of MA computation up to 2004 was given by Attali et al. [7]. The MAT in its original definition was formulated for 2D objects, but is extendable to any number of dimensions. A definition for a 3D setting is used in this paper. Furthermore, here the MAT is only computed for polyhedral reference sets, since the geometries in focus, i.e., the die cavities of a forging press, are represented by triangulated surfaces in this project.

The MAT describes a reference set S with boundary surface ∂S by two unique, well-defined parts: the *Medial Axis* M, which is the set of centers of maximal touching balls and a real-valued *radius function* r defined on M [8]. An open ball B bounded by the sphere ∂B is *touching* ∂S, if ∂B intersects ∂S in a non-empty subset of ∂B, and $B \cap \partial S = \emptyset$. A touching ball is called *maximal*, if it is not contained in any other touching ball. The radius function gives the radius of the maximal ball for each point p of M, which is the distance to the boundary in this point. The points where ∂B_p with center p intersect ∂S are called *footpoints* of p. An example for a 2D Medial Axis is shown in Fig 1. For a 3D body, the MA is in general of dimension 2, i.e., consists of surfaces[1].

In general M consists of several connected components, one inside and the others outside of S, compare Fig. 2. Here, a connected component is a subset M_i of M so that for any two points of M_i there exists a path in M_i joining them. In the following, only the inner component is used, where each point of M is also a point of S.

[1] Note, that it is in general not a 2-manifold, since it can have branches and fin-like structures.

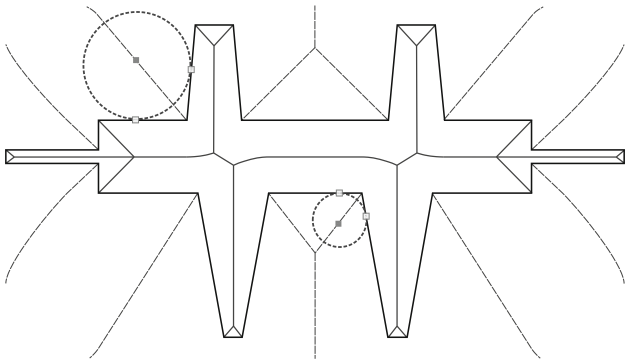

Fig. 2 Example for MA consisting of several components (in *gray*) with some maximal circles. Only the inner one is considered

A point $p \in M$ can be classified looking at the type of contact of its respective touching ball, see [9, 10]. The point p is a

- *sheet point*, if B_p touches ∂S in exactly 2 points;
- *seam point*, if B_p touches ∂S in exactly 3 points;
- *junction point*, if B_p touches ∂S in at least 4 points;
- *boundary point*, if B_p has radius 0.

Sheet points lie on 2D manifolds which are bounded by seams, boundary points and junctions. Seam points lie on 1-manifolds where 3 sheets intersect, and junctions are points of intersection of several seams. Boundary points are limit points of M and arise where M touches ∂S in sharp, convex edges of the solid. An edge is convex, if two points neighboring a point on the edge can be connected by a line that lies completely inside the solid.

The MA concept is connected to Voronoi diagrams. A *Voronoi diagram* of a discrete set of points $X = \{x \in R^3\}$ called *sites* is a partition of space, such that each site x_i is mapped to a cell of the partition and the points in the cell are closer to x_i than to any other site x_j. The cell boundaries contain points which have two closest sites and junction points of cell boundaries have more than two closest sites. So, the boundary graph of a Voronoi diagram can be described as the MA of $\mathbb{R}^3 \setminus X$ with boundary X [6] (Fig. 3).

The computation of Voronoi diagrams is a well understood problem and there exist several implementations of stable algorithms, see [11]. It is possible to approximate the Medial Axis of a reference solid by filtering the Voronoi diagram of a point sample of the solid's surface. Since in this project the solids are described by triangulations, the vertices of the triangulation are the input. The filtering is based on an approach by Turkiyyah et al. [12], using a refinement for the triangulation proposed by Rivara [13]. The 3D algorithm then has 5 main steps:

1. Compute a Voronoi diagram of the input vertices.
2. Refine the input triangulation, so that a sampling criterion is reached.
3. Filter all parts of the Voronoi diagram that lie inside the solid, i.e., the closed die gap.
4. Clean up parts that arise due to discretization, but are not a part of the Medial Axis.
5. Use an iterative least-squares scheme to locally converge Voronoi junctions to medial points. This takes the triangulation of the solid into account.

During the computation, radii are computed for all Voronoi junctions, resulting in an approximation to the complete MAT. The result is stored as a graph with the filtered and converged Voronoi junctions as vertices and the Voronoi edges as graph edges. The algorithm for the Medial Axis computation was implemented in C++, based on the CGAL library and has an interface to MATLAB.

2.3 Displacement graph

Behrens et al. [4] used the Medial Axis in the 2D case to describe main flow paths, i.e., paths of material displacement. More precisely, they computed the MA respective the die gap between the tool surfaces. In 2D, the Medial Axis can be represented in a tree structure. This concept is now extended to 3D. The MA of the die gap, represented as surfaces in STL format, is computed and the displacement paths are described by the weighted medial graph $G(V, E)$ called *displacement graph*, see Fig. 4. Its vertices $v_i \in V$ store their positions, radii and footpoints, while the edges $e_i \in E$ are weighted by the difference of radii of their incident vertices. Graph edges contain references to the Voronoi facets they border, which allows to classify them as boundary (1 incident facet), sheet inner edges (2 incident facets) or seam edges (>2 incident facets). Vertices are considered to lie on the boundary, if they are incident to at least one boundary edge. The shortest weighted paths between vertices in the graph can then be calculated. A shortest path from vertex v_i to vertex v_j has the property that there exists no other path with a lower weight. Now, the single-source shortest paths from a defined point or a quantity of points to points on the boundary of the graph are calculated. The boundary of the graph represents the free surface of the material, i.e., the part where it is not in contact with the tool. In the following, it is shown how to define the weight of the graph's edges so that they are a measure of geometric resistance that has to be overcome to move material along this path.

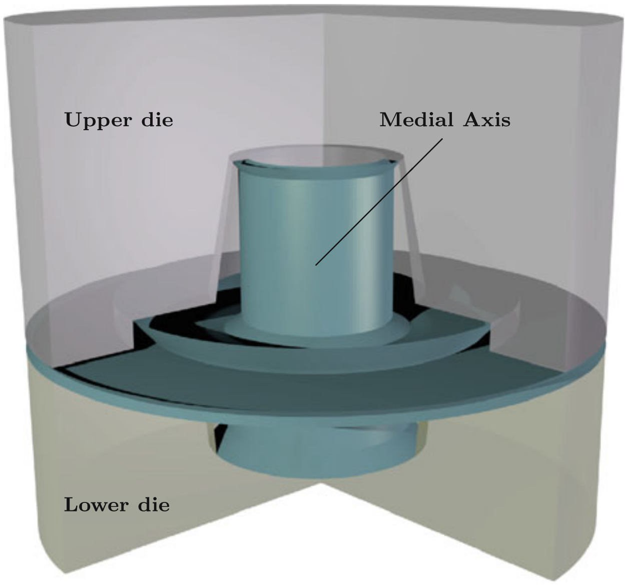

Fig. 3 3D model of forging dies with the Medial Axis of the die gap. The MA consists of several surface patches, connected at their borders

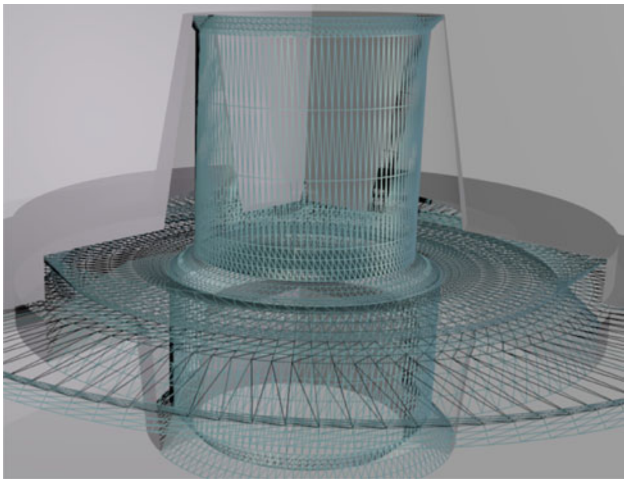

Fig. 4 Close view of MA triangulation which is used as approximation to the MA graph

3 Inverse 3D simulation program

The 3D inverse simulation is based on the geometric resistance approach and the MAT. It is an iterative process with discrete time steps that consist of four basic sub steps:

1. Calculation of displacement paths via MA.
2. Calculation of resistance values for boundary points of the resistance graph which correspond to the free material surface.
3. Remove volume at the free surfaces according to their resistance values.
4. Attach volume at contact area between tool and material.

The input geometries for the calculation of the MA are the upper and lower die models in a Standard Tessellation Language (STL) representation. The MA of the workpiece is a subset of it that is limited by the workpieces surface. The geometry-dependent flow resistance on the boundary vertices of the MA-graph determines the quantitative displacement of the material and is computed in MATLAB.

In a cutting plane, the Medial Axis or displacement paths can be interpreted as a binary tree where directions of flow are determined, see Fig. 1. For 3D objects, complexity increases: the tree becomes a collection of surface elements bordered by a graph, see Fig. 4. This leads to a massive increase of the degrees of freedom for the material flow. In the graph structure, a shortest path algorithm identifies the paths of lowest cost from all resistance elements to the boundary points and assigns them a weight in dependence of the cost for reaching them. Dijkstra's algorithm has been chosen to identify shortest paths in an efficient manner [14]. The edges are weighted corresponding to their adjacent MA-points resistance. The direction of displacement is tangential to the local MA and along the gradient of decreasing resistance. The assigned weights give an estimation for the quantity of material displacement from the free surface of the forging. The volume removed at the free surface is then attached to the contact surface between tool and workpiece so that volume constancy is fulfilled. The total amount of material displacement for each calculation step can be chosen manually and affects the height change of the die.

Figure 5 shows the Medial Axis M between a segment of the upper (UD) and lower die (LD). The degree of plastic deformation is given by the angle between the tool movement direction $\mathbf{s}$ and normal $\mathbf{n}_p$ of the tangent plane T_p of the MA at point p. The angle gives the information on the local orientation of the flow path with respect to the die movement direction. A small angle indicates lateral material flow, whereas a large angle indicates vertical flow. The angle α between the footpoints f_i and their projection $\mathbf{d}_p$ on

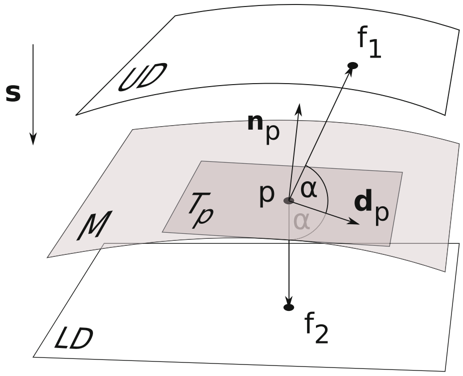

Fig. 5 Local setting at a point on the MA. Geometric resistance depends on the normal vector $\mathbf{n}_p$ of the tangential plane T_p of M in p

the MA indicates if the material moves between nearly parallel die surfaces (large angle) or is compressed/can expand due to inclined surfaces (small angle). The points p_i on the MA are connected in a graph structure. In its simplest form, the *geometric resistance* k_{gr} for a point p (in absence of friction) is calculated in dependence of the MA normal and radius by

$$k_{gr} = C \cdot |\cos(\angle(\mathbf{s}, \mathbf{n}_p))| \cdot r(p) \tag{1}$$

The parameter $r(p)$ is the radius of the maximal touching ball, i.e., the distance between the footpoints f_i and p. The factor C is a calibration factor that is deduced empirically. The material flow depends primarily on geometric parameters of the dies. However, the material flow depends also on friction between die and workpiece as well as strain, strain rate and temperature. In the first instance, the presented simulation program reduces the forming process to a geometric problem. Technological parameters have to be included in a further step to improve the accuracy.

4 Forward simulation using optimization algorithm

The classical way to obtain an optimal preform shape is to carry out FE-based simulations of the process under analysis and to modify the preform geometry in an iterative manner. This approach requires carrying out and analyzing numerous process simulations, which is very time-consuming and computationally intensive compared to the inverse simulation technique. A means of minimizing the overall analysis time is to use optimization software in order to reduce the intervention of staff.

One objective within the framework of this research work was to obtain a suitable preform geometry of the billet in order to minimize the necessary forming energy. To this end, a parametric design optimization algorithm has been developed, where a coupling of different CAE software packages has been realized (see Fig. 6).

The first instance of the algorithm is the CAD system Pro/Engineer, where a parameterized model of a cylindrical billet is created. This is exported and transferred to HyperMesh for automatic meshing. The meshed billet is then delivered to the preprocessor of the FEA software Transvalor Forge 2011 to build up the simulation model, assign material properties and to define boundary and initial conditions. After having setup the simulation model, the computation is invoked and runs in batch mode.

Once the computational simulation terminates successfully, a subsequent evaluation of the necessary forming energy is done in the postprocessor GLView. This simulation chain has been attached to the optimization software LS-OPT for further investigations, where a meta-model based, D-optimal optimization strategy with domain reduction has been chosen, see [15]. The design variable in this case is the radius of the billet with the constraint of volume constancy.

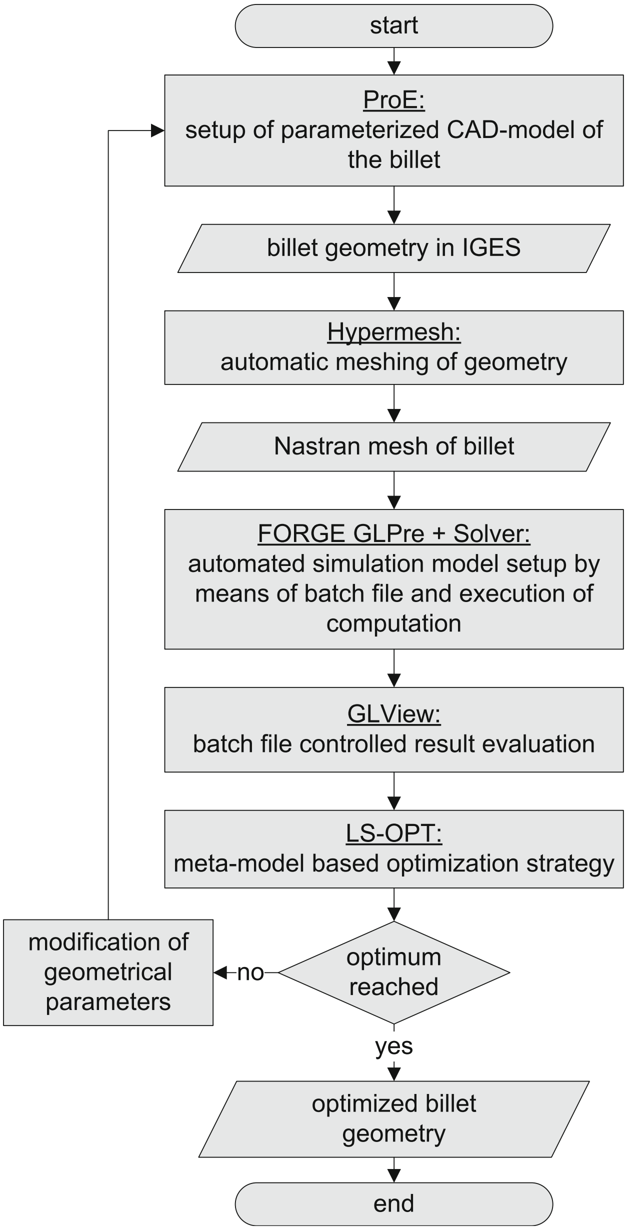
start
ProE:
setup of parameterized CAD-model of the billet
billet geometry in IGES
Hypermesh:
automatic meshing of geometry
Nastran mesh of billet
FORGE GLPre + Solver:
automated simulation model setup by means of batch file and execution of computation
GLView:
batch file controlled result evaluation
LS-OPT:
meta-model based optimization strategy
optimum reached
no
modification of geometrical parameters
yes
optimized billet geometry
end

Fig. 6 Optimization process flow chart

5 Results and discussion

5.1 Inverse simulation

The investigation on 3D bodies with basic forming elements build the groundwork for the transfer of simulation from 2D into 3D space and provide insight into potentials

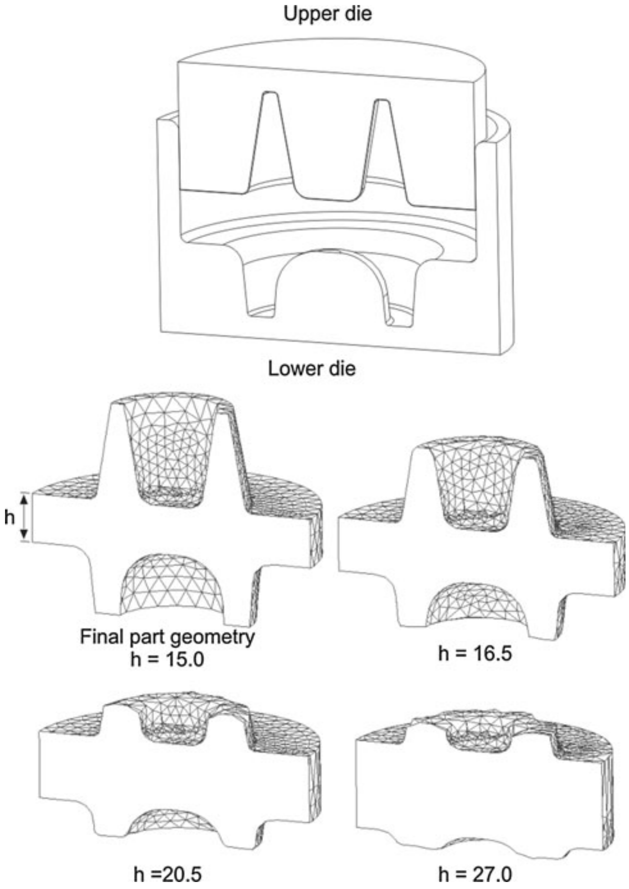

Fig. 7 Sequence of the inverse 3D simulation at different die heights

and open tasks with respect to the inverse approach. Figure 7 shows the results of an inverse simulation sequence carried out on a 3D exemplary forging.

Main forming elements of the dies are a flange with horizontal material flow and cavities with vertical flow. The upper and lower die cavities differ in depth, draft angles and curvature of the orifice and hence affect flow resistance differently. The input and output files are in the STL format and results can be exported as input for FE-simulations. Edges at points where initial material displacement takes place have no curvature as they are not regarded at present by the algorithm. Meshes which exhibit wrinkles due to re-meshing during the procedure can be smoothed by standard CAD software. The inverse simulation method gives reasonable results, though systematic quantitative analysis of results with respect to reference forming processes has to be carried out in a second validation step. In order to get process specific reference preforms, forward simulations have to be performed like they are described in Sect. 4 of this paper. It should be mentioned that the final result is not geometric primitives like, e.g., cylinder, which are usually the first input in practical applications. That reflects one of the main difficulties that have to be solved: an admissible target shape domain and an according condition for releasing contact knots have to be predetermined.

5.2 Forward simulation

The optimization from LS-OPT, see Sect. 4, showed that a maximum admissible billet radius is the case where the least amount of plastic deformation energy is needed to obtain the desired final part geometry. This radius happens to be the same radius as the lower die inner radius. If a closer look is taken on the process itself it can be seen that the forging process is a combination of an upsetting and extrusion process. The greater the radius of the billet the less material flow and thus deformation energy is needed to obtain die filling. For this billet geometry the forging process consists only of a pure extrusion process as the billet already lies against the lower die wall.

The time necessary to obtain this solution was 30 h as 48 process simulations had to be carried out until convergence was reached[2]. Since computation time of the inverse simulation was less than one hour it can give a first estimation of a reasonable preform, effectively limiting the parameter range and reducing the number of iterations and computation time for the forward optimization. Nevertheless, the inverse simulation approach does not deliver necessary information on physical quantities itself. It provides merely the geometry rather than the temperature, strain and stress distributions in the workpiece material during forming. Therefore a combination of both approaches seems to be the most efficient way for preform geometry determination. A pre-design of the initial billet geometry can be done by means of the inverse technique and a subsequent adjustment based on physical quantities of the process and the workpiece can be done by means of FEA simulations.

5.3 Voxel representation

A mesh representation of the material during the backwards transformation has several drawbacks. The mesh quickly degenerates and has to be repaired in intermediate steps. Moreover the assignment of moving body-knots to MA-knots is not unique and several definitions depending on particular cases that may occur have to be formulated.

Therefore, other representation methods were evaluated and a voxel representation may be better suited for the task. A simple approach to convert a solid, e.g., represented as a surface mesh, to a voxel representation, is to partition the 3D space in a regular cubic grid. Then each grid junction (voxel) is assigned a boolean value, depending on whether

[2] All computations were carried out on four Intel Xeon CPUs with 2.66 GHz and 8 GB RAM.

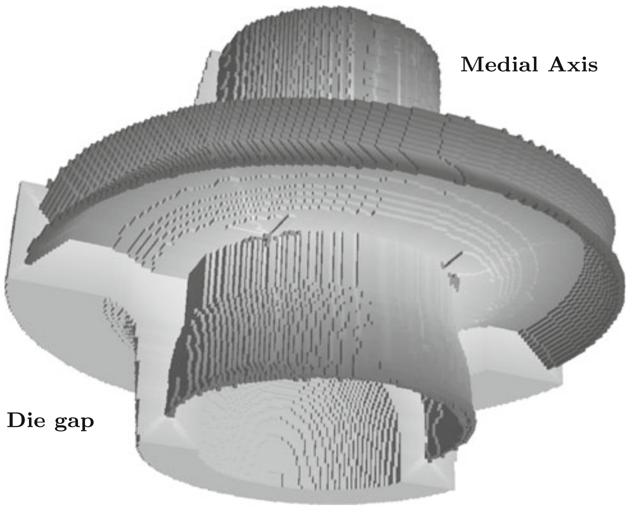

Fig. 8 Voxel representation of a die gap with MA

the voxel lies inside the solid or not. Figure 8 gives an example of voxel representation.

Additional attributes can be assigned to voxels, e.g., density distribution in medical imaging applications. This property qualifies voxel representation for assigning geometric resistance distribution over the forging part that determines the material displacement in a forging step. The Medial Axis of solids in voxel representation can be approximated by a thinning operation which simulates the grassfire propagation proposed by Blum [16].

6 Summary and outlook

A fast inverse 3D simulation method for a first estimation of preforms in hot die forging processes starting from the finished product shape was presented. The resistance for material flow is determined by the geometric analysis of the die cavity derived from a Medial Axis Transformation. So far the simulation is being studied on rotationally symmetric parts. A forward simulation in combination with optimization technique was worked out in order to get a reference with respect to the preform and time cost.

In the future, the potential of the 3D simulation program for more complex shapes has to be investigated. The change of shape over the calculation steps requires a robust connection between moving points belonging to the medial graph on the one hand and the distribution of material volume on the other hand as well as the re-meshing of the triangulated material surface. This is connected with drawbacks for complex forgings. Voxel representations were studied, which show potential for adoption to the presented simulation system. Another crucial point is the integration of production requirements. Forging processes start with a primary preform that is subjected to production process dependent limitations. The development of a strategy of releasing knots in dependence of a target morphology is a subject for further research to meet such constraints.

Acknowledgments The authors thank the German Research Foundation (DFG) for the financial support of the research project “Schnelle inverse Materialflusssimulation für die Massivumformung mittels der dreidimensionalen Mediale-Achse-Transformation” (project numbers BE 1691/110-1 and WO 954/7-1).

References

1. Yang C, Ngaile G (2010) Preform design for forging and extrusion processes based on geometrical resemblance. Proc Inst Mech Eng Part B J Eng Manuf 224(9):1409–1423
2. Chang CC, Bramley AN (2000) Forging preform design using a reverse simulation approach with the upper bound finite element procedure. Proc Inst Mech Eng Part C J Mech Eng Sci 214(1): 127–136
3. Wienströer M (2004) Prozesssimulation der Stadienfolge beim Schmieden mittels Rückwärtssimulation. Ph.D. thesis, Universität Hannover
4. Behrens BA, Wienströer M, Conrads H, Gue J (2006) Design integrated process modelling of forging sequences via backward simulation. Prod Eng Res Dev 13(2):157–160
5. Blum H (1967) A transformation for extracting new descriptors of shape. In: Wathen-Dunn W (ed) Proc. models for the perception of speech and visual form. MIT Press, Cambridge, pp 362–380
6. Wolter FE (1992) Cut locus and medial axis in global shape interrogation and representation. MIT Design Lab, Report 92-2, US National Sea Grant Library
7. Attali D, Boissonnat JD, Edelsbrunner H (2004) Stability and computation of medial axes: a state of the art report. In: T Möller BH, Russell B (eds) Mathematical foundations of scientific visualization, computer graphics, and massive data exploration. Springer, New York
8. Wolter FE, Friese KI (2000) Local and global geometric methods for analysis interrogation, reconstruction, modification and design of shape. In: Proceedings of computer graphics international 2000. Invited paper., pp 137–151
9. Culver T (2000) Computing the Medial Axis of a Polyhedron Reliably and Efficiently. Ph.D. thesis, Department of Computer Science, University of North Carolina, Chapel Hill
10. Giblin PJ, Kimia BB (2003) On the local form and transitions of symmetry sets, medial axes, and shocks. Int J Comput Vis 54(1/2/3):143–157
11. Okabe A, Boots B, Sugihara K, Chiu SN (2009) Spatial tessellations: concepts and applications of Voronoi diagrams. Wiley, New York
12. Turkiyyah GM, Storti DW, Ganter M, Chen H, Vimawala M (1997) An accelerated triangulation method for computing the skeletons of free-form solid models. Comput Aided Des 29(1): 5–19
13. Rivara MC (2009) Lepp-bisection algorithms, applications and mathematical properties. Appl Numer Math 59(9):2218–2235
14. Dijkstra EW (1959) A note on two problems in connexion with graphs. Numerische Mathematik 1(1):269–271
15. Stander N, Roux W, Goel T, Eggleston T, Craig K (2012) LS-OPT users manual—A Design Optimization and Probabilistic Analysis Tool for the Engineering Analyst, v4.2. Livermore software technology corporation

16. Jonker PP (2000) Morphological operations on 3D and 4D images: from shape primitive detection to skeletonization. In: Borgefors G, Nystrm I, Baja GSd (eds) Discrete geometry for computer imagery. Springer Berlin Heidelberg, no. 1953 in lecture notes in computer science, pp 371–391