PREFORM DESIGN FOR FLASH-LESS DIE FORGING

Final Report Submitted to FIERF

Karthikeyan Kumaran, Research Assistant

Gracious Ngaile, Professor

Advanced Metal Forming and Tribology Laboratory

Department of Mechanical and Aerospace Engineering,

North Carolina State University

Raleigh NC

January 6, 2026

TABLE OF CONTENTS

Chapter 1:

INTRODUCTION

1.1 Introduction

Forging is a manufacturing process in which a workpiece is plastically deformed into a desired shape under high compressive forces. Forged components typically exhibit superior mechanical properties, high structural integrity, and enhanced fatigue resistance compared to cast or machined counterparts. In addition, forging offers advantages such as improved material utilization, reduced waste, and high production efficiency, making it a preferred manufacturing route for critical components in automotive, aerospace, energy, and heavy machinery applications.

The forging process is governed by a complex interaction of multiple parameters, including billet geometry and material properties, die geometry and material, lubrication conditions, deformation mechanics, forming equipment, and thermal environment. Among these parameters, billet and preform design play a particularly critical role in controlling material flow, ensuring complete die filling, and preventing common forging defects such as laps, underfilling, fold-over, and cracking.

As industrial demand for complex, high-performance forged components continues to increase, multi-stage forging processes are frequently employed. In such processes, the final component is produced through a sequence of intermediate forming steps, each requiring a carefully designed preform. Improper preform geometry can lead to nonuniform strain distribution, excessive flash formation, increased forging loads, and premature tool wear. Consequently, accurate preform design is essential to achieving defect-free parts while minimizing manufacturing cost and energy consumption.

Material cost constitutes a significant fraction of the total production cost in forging operations. Precision forging and flashless forging techniques have therefore attracted considerable attention due to their ability to reduce material waste and eliminate secondary trimming operations. In flashless forging, the billet volume closely matches the final part volume, resulting in improved material utilization and reduced post-processing requirements. Moreover, the absence of flash

promotes more uniform metal flow and can lead to enhanced mechanical properties in the forged component.

Despite these advantages, flashless forging places stringent demands on billet volume control and preform geometry design. Even small deviations in billet shape or volume can result in incomplete die filling or excessive forging loads. As a result, the success of flashless and near-net-shape forging processes depends heavily on the accuracy and robustness of the preform design methodology.

Traditional preform design approaches often rely on empirical rules, trial-and-error methods, or designer experience. While such approaches may be effective for simple geometries or well-established processes, they become increasingly inefficient and unreliable for complex part geometries or new materials. The widespread adoption of finite element analysis (FEA) has significantly improved the ability to simulate forging processes and predict material flow, strain distribution, and defect formation. However, determining an optimal preform geometry through iterative finite element simulations remains computationally expensive and time-consuming.

To address these challenges, several systematic preform design methodologies have been proposed in the literature, including optimization-based techniques, topology-based methods, and reverse engineering approaches. Among these, the geometrical resemblance methodology offers a practical and physically motivated framework for preform design. This method is based on the concept that intermediate preforms should geometrically resemble the final forged part while accounting for material flow behavior under realistic forming conditions.

The geometrical resemblance approach typically begins with the definition of an oversized part geometry that maintains similarity to the final desired shape. A finite element simulation is then performed to forge this oversized geometry using an assumed initial preform. Although the resulting forged shape may contain defects, it provides valuable information regarding material flow paths. By tracing material points backward from intermediate forged geometries to their undeformed configurations, a sequence of preforms can be generated. This iterative backtracking process continues until the forged shape closely matches the desired final geometry.

One of the key advantages of the geometrical resemblance methodology is that it inherently captures the effects of friction, thermal gradients, and nonuniform deformation, as these factors are embedded within the finite element simulations. As a result, the generated preforms are physically realistic and well-suited for practical forging applications. However, conventional implementations of this methodology often require extensive manual intervention, particularly during material point selection and geometry reconstruction, which limits its efficiency and scalability.

In response to these limitations, recent efforts have focused on automating the geometrical resemblance methodology to improve repeatability and reduce user dependency. Automation enables systematic generation of material points, streamlined data transfer between CAD and finite element platforms, and rapid iteration of preform designs. Such advancements are particularly important for industrial applications, where time constraints and consistency are critical.

This report builds upon the geometrical resemblance methodology and focuses on its application to progressive die forging processes. Emphasis is placed on the development and implementation of an automated framework for preform design, with the goal of improving material utilization, reducing defect formation, and enhancing process robustness. The work presented herein is based on a Master's thesis and has been adapted and refined to meet sponsor reporting requirements, with an emphasis on clarity, technical rigor, and industrial relevance.

1.2 Research Objectives

The overall objective of this study is to develop preform design schemes for reducing flash in die forging. The preforms are iteratively searched by backward tracing of material points in the FE models. The specific objectives are to

(a) Survey families of forgings that are carried out in closed die forging with and without flash for quantifying material waste caused by forging with flash and identifying potential candidate parts that might benefit from the proposed study.
(b) Automate the preform design methodology based on geometrical resemblance for 2D forging parts.

(c) Develop preform design schemes for forgings that exhibit 3D deformation behavior. The schemes will be developed for use in conjunction with the point tracking function in DEFORM 3D software.

(d) Build a scaled-down forging test setup to experimentally validate the proposed schemes for 3D geometries.

(e) In collaboration with an industry partner, validate the developed schemes in an industrial setup.

Chapter 2:

LITERATURE REVIEW

Preform design plays a critical role in multi-stage forging processes, as it directly influences material flow behavior, defect formation, forging load, and overall process efficiency. An effective preform design ensures complete die filling, minimizes defects such as laps and folds, reduces material waste, and prolongs tool life. As a result, considerable research effort has been devoted to developing systematic methodologies for preform design in forging operations.

Early approaches to preform design were largely based on empirical rules and designer experience. These methods relied on geometric intuition and historical knowledge of similar parts and processes[1-8]. While empirical approaches remain useful for simple geometries and well-established forging operations, they are generally inadequate for complex parts or new materials, where material flow behavior is difficult to predict. In such cases, trial-and-error approaches can result in excessive development time and increased production costs.

With the advancement of computational capabilities, finite element analysis (FEA) has become a powerful tool for analyzing metal forming processes. FEA enables detailed prediction of material flow, strain distribution, temperature evolution, and defect formation during forging [9-12]. Numerous studies have demonstrated the effectiveness of FEA in evaluating forging process parameters and improving part quality. However, while FEA provides valuable insight into process behavior, it does not by itself yield an optimal preform geometry. Identifying a suitable preform typically requires multiple simulation iterations, which can be computationally expensive and time-consuming.

To overcome these limitations, several systematic preform design methodologies have been proposed. Optimization-based approaches formulate preform design as a numerical optimization problem, in which objective functions, such as minimizing forging load, material waste, or defect severity, are evaluated subject to geometric and process constraints. Genetic algorithms, response surface methods, and gradient-based optimization techniques have all been applied to forging preform design with varying degrees of success [3-16]. Although these approaches can produce

optimized solutions, they often require a large number of simulations and careful parameter tuning, limiting their practical applicability in industrial settings.

Topology-based methods represent another class of preform design techniques. In these approaches, material distribution within a design domain is optimized to achieve desired performance characteristics [17-18]. Topology optimization has been successfully applied to forging preform design by identifying regions where material can be added or removed to improve flow behavior. However, topology-based methods often produce complex geometries that are difficult to manufacture and require additional post-processing or geometric smoothing before practical implementation.

Reverse engineering and backward tracing methods offer an alternative strategy for preform design. These methods are based on the principle of tracing material points backward from a desired final geometry to an undeformed configuration. By reversing the material flow path, an initial preform geometry can be reconstructed. This approach is physically intuitive and directly linked to the deformation mechanics of the process. However, early implementations of backward tracing methods were often limited by manual intervention and difficulties in accurately tracking material points through large deformations.

Among reverse engineering approaches, the geometrical resemblance methodology has gained significant attention due to its practical effectiveness and physical basis. Originally proposed by Yang and Ngaile[19], this method assumes that intermediate preforms should geometrically resemble the final forged part while accounting for realistic material flow behavior. The methodology begins by defining an oversized part geometry that maintains similarity to the final shape. A finite element simulation is then performed to forge this oversized geometry using an assumed preform. Although the resulting forged shape may contain defects, it provides valuable information about material flow paths.

Using the results of the finite element simulation, material points are traced backward from intermediate forged geometries to their undeformed configurations. This backward mapping process generates a sequence of preform geometries that progressively approach the desired final shape. The process is repeated iteratively until the forged geometry closely matches the target part.

Because friction, thermal effects, and nonuniform deformation are inherently captured in the simulation, the resulting preforms are physically realistic and well suited for industrial forging applications.

Several researchers have extended the geometrical resemblance methodology to different forging scenarios, including multi-stage forging, axisymmetric components, and complex three-dimensional geometries. These studies have demonstrated that the method can significantly reduce material waste, improve die filling, and lower forging loads. Despite these advantages, most reported implementations rely heavily on manual selection of material points and user-dependent geometry reconstruction. This manual intervention introduces variability, reduces repeatability, and limits scalability for industrial deployment.

To address these limitations, recent research has focused on automating key aspects of the geometrical resemblance methodology. Automation efforts include algorithmic generation of material points, scripted data extraction from finite element simulations, and automated reconstruction of preform geometries within CAD environments. By reducing user involvement, automation improves consistency, shortens development time, and enables systematic exploration of design alternatives. However, existing automated implementations are often tailored to specific part geometries or forming conditions and may lack generality. In addition, integration between CAD platforms and finite element solvers remains a challenge, particularly when dealing with large datasets and complex geometries. These limitations highlight the need for a robust, flexible, and fully automated framework that can be applied to progressive die forging processes in an industrial context.

In summary, the literature demonstrates that while numerous preform design methodologies have been developed, each approach exhibits trade-offs between accuracy, computational efficiency, and practical applicability. Empirical methods lack robustness, optimization-based techniques are computationally intensive, and existing implementations of reverse engineering approaches often require substantial manual effort. These gaps motivate the development of an automated geometrical resemblance framework that combines physical realism with computational efficiency and industrial scalability. This need forms the basis for the methodology presented in Chapter 3.

Chapter 3:

METHODOLOGY

3.1 Principle of the Preform Design Using Geometrical Resemblance

The primary criterion in the preform design of a multi-stage forging process is to ensure that the final forged part is free of defects such as die under-fill, folding, laps, or fractures. A properly designed preform should facilitate material flow in a way that avoids these issues. The methodology proposed here leverages the capability of finite element (FE) simulation software to track material points throughout the deformation process, as illustrated in Fig. 3.1.

The process begins by assuming an initial preform that results in a part slightly larger than the target geometry (Fig. 3.1A). FE simulation of the forging process is then conducted (Fig. 3.1B), which may yield a defective part. If the defects are localized at the boundaries, a backward tracking approach can be used. By overlaying the desired part geometry onto the simulation result (Fig. 3.1C), material points along the boundary of the desired part can be traced back to their original positions in the billet.

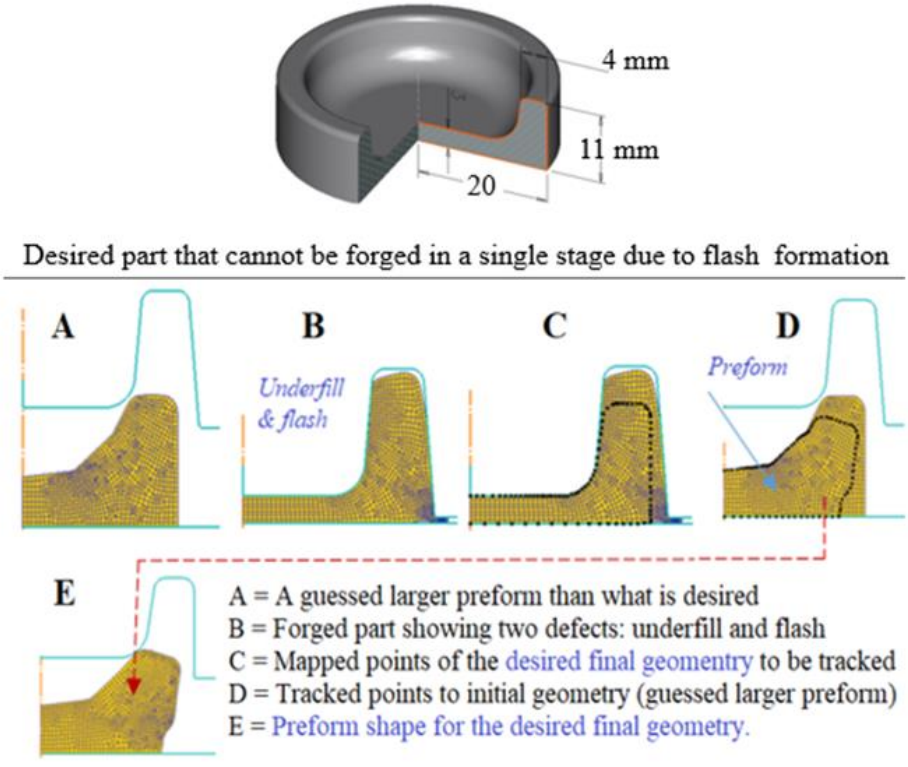


Figure 3.1: Illustration of point tracking in an FE model to obtain a preform

This mapped geometry becomes the new preform (Fig. 3.1E). A comparison between the initial guess (Fig. 3.1A) and the obtained preform (Fig.3.1E) highlights the geometric refinement achieved through this method.

One of the key advantages of the geometrical resemblance methodology is that it inherently captures the effects of friction, thermal gradients, and nonuniform deformation, as these factors are embedded within the finite element simulations. As a result, the generated preforms are physically realistic and well-suited for practical forging applications.

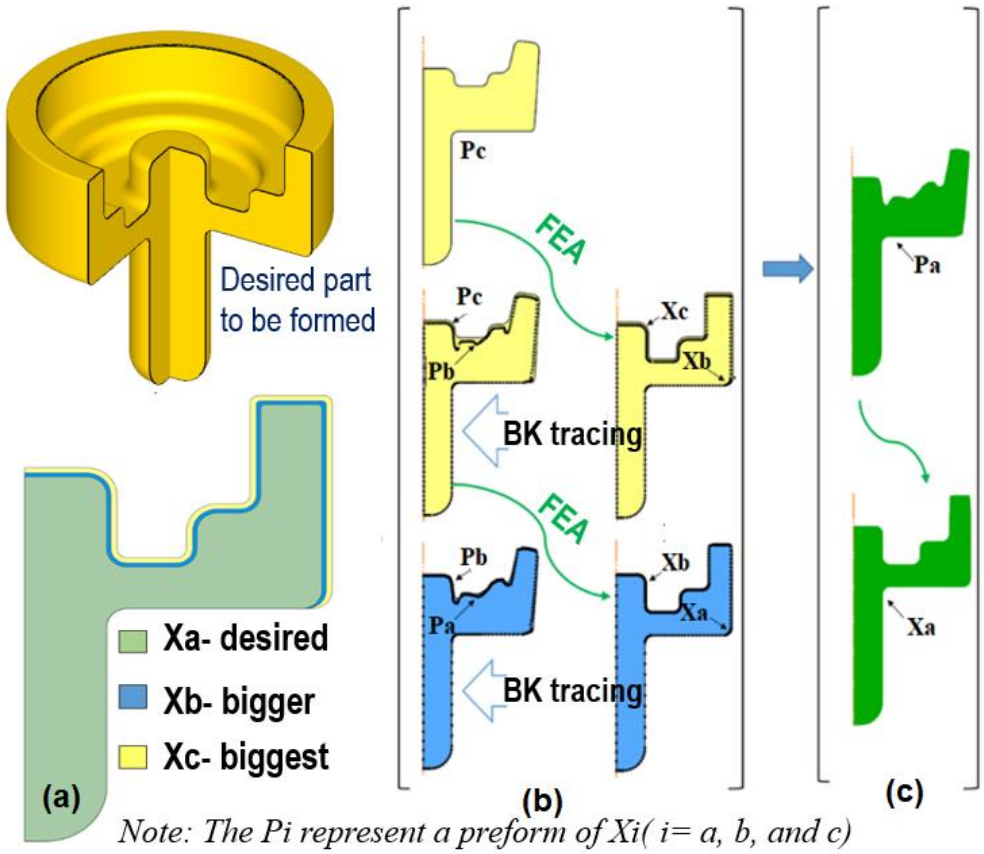


Figure 3.2: (a) geometrically resembling parts and (b) Illustration of preform design of part Xa

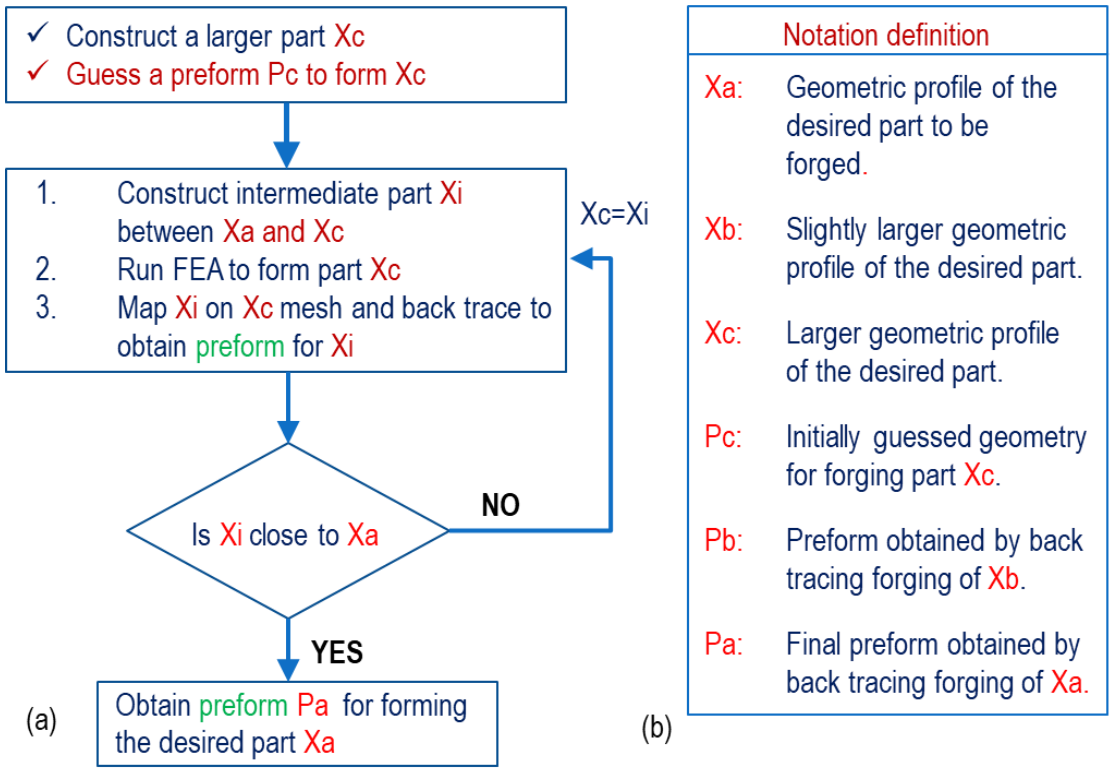


Figure 3.3: (a) FE Simulation flow chart to obtain a preform, (b) Definition of notations

Figure 3.2 further illustrates this process using a three-stage iterative scheme to converge on the optimal preform Pa. Figure 3.3 provides a flow chart and clarifies the notations used in this iterative preform design approach. The key steps are outlined below:

(i) Define the desired part geometry as Xa. Construct an oversized, geometrically similar part Xc as an initial target.

(ii) Guess a preform Pc to forge part Xc. While Xc may exhibit defects such as underfill or flash, it must completely enclose an intermediate part Xb.

(iii) Identify the geometry Xb and trace its boundary points backward from Xc to preform Pc to define the intermediate preform Pb.

(iv) Construct a new intermediate geometry Xi that closely resembles Xa. Backtrack the material points of Xi from its forging step (preform Pb) to define a new preform Pi.

(v) Repeat step (iv) iteratively until Xi closely approximates Xa. The final preform Pa is determined by tracing material points of Xa back to the forging step associated with Pi.

This iterative scheme significantly reduces the number of trial-and-error attempts and results in a defect-free preform.

While the proposed preform design methodology has been shown to converge quickly, particularly for two-dimensional (2D) forging problems, the processes of material point tracking within the finite element (FE) models and CAD-based geometry reconstruction during iterative steps require repetitive operations that can become tedious. To improve the overall efficiency and practicality of the methodology, fully or semi-automated schemes for these repetitive tasks are necessary. The following sections discuss strategies for material point creation for 2D and 3D parts, respectively.

3.2 Material Point Creation for 2D Parts

The points representing the part geometry, denoted as X_c, X_b, X_a, and $X_i$in Figure 3.2, are generated at each iteration of the geometrical resemblance methodology. In the conventional approach, these points are selected manually; however, this process should be automated to improve repeatability and reduce user dependency.

The most straightforward approach to obtaining the material points required to represent a part's geometry is to generate them directly within the CAD software after creating the part model. This procedure involves the following steps:

- Creating the part geometry using CAD software.
- Inserting points along the edges of the part to represent the 2D geometry.
- Extracting the coordinate data of the points inserted along the part edges.
- Formatting the point cloud data into a structure acceptable by the DEFORM software package.
- Saving the point cloud data in a file format compatible with DEFORM.
- Importing the saved point cloud data into DEFORM using the material point tracking feature.

The specific format required for the material point data file to be imported into DEFORM is shown in Figure 3.4. The material point data file must be saved with a *.dat* extension.

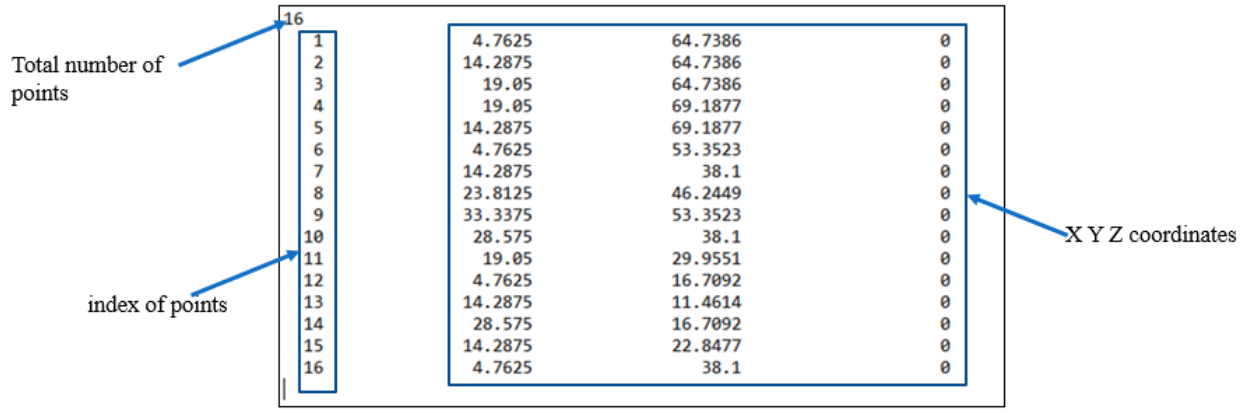


Figure 3.4: Format for saving the point data for importing in DEFORM.

Material point creation can be automated using either SolidWorks or SpaceClaim software packages. In this work, only the procedures implemented using SpaceClaim are discussed, as this platform was found to be significantly faster and more efficient.

3.2.1 Material Point Creation Using SpaceClaim

Material point creation using SpaceClaim is particularly effective because it requires only a single Python script to perform all necessary operations, including converting the part geometry into material points and formatting the data into a DEFORM-compatible file. As illustrated in Figure 3.5, material points for a given 2D part can be generated by first creating the CAD model of the required geometry, as shown in Figure 3.6a. SpaceClaim treats lines, arcs, fillets, and other geometric features of the part as individual curves.

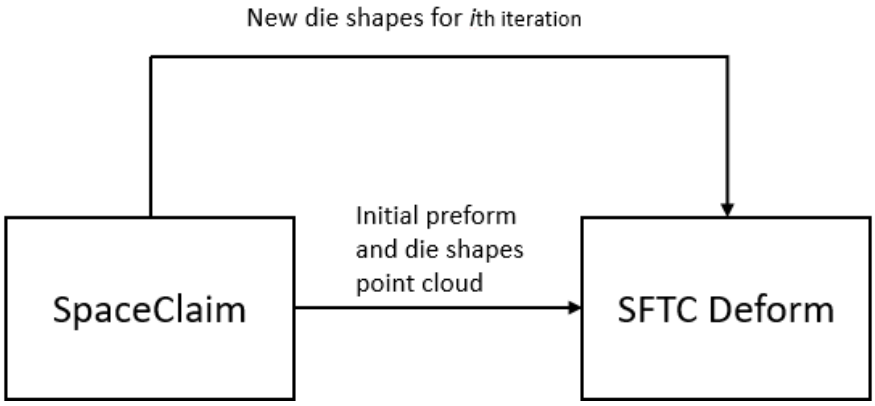
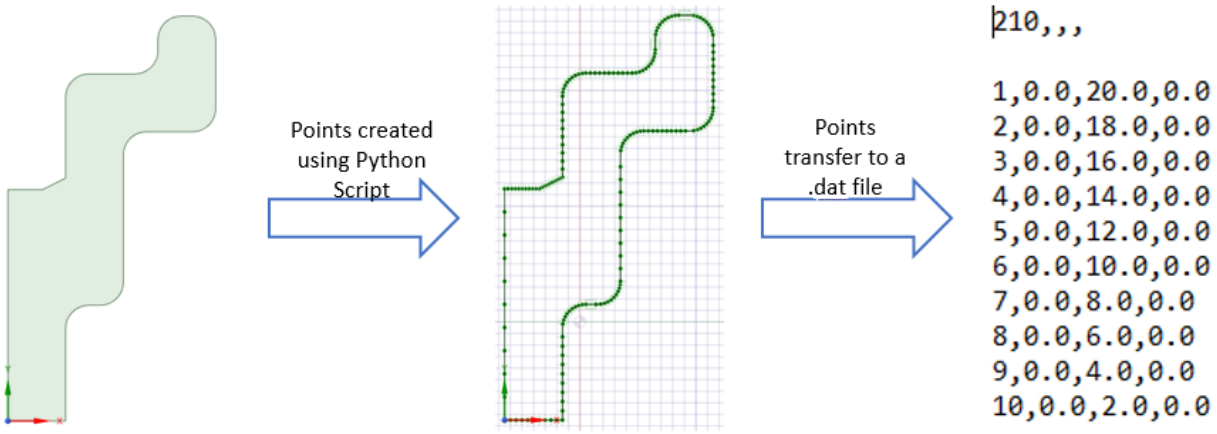


Figure 3.5: Flow of data between Space Claim and DEFORM software package.

A fixed number of datum points is then inserted along each curve using the point creation tool. At this stage, the 2D part is represented as a collection of discrete points, as shown in Figure 3.6b. The coordinates of all generated points are subsequently saved as a *.dat* file, as illustrated in Figure 3.6c. This file can be directly imported into the DEFORM software package.



a) 2D part in SpaceClaim b) Part represented as points c) Points stored as *.dat* file

Figure 3.6: Procedure for material point creation in SpaceClaim.

Using the scripting tools available in SpaceClaim, the entire material point creation process can be automated through Python code. Automation significantly reduces the time required to generate point coordinates. For simple 2D geometries, the *.dat* file containing point coordinates can be created in less than three seconds.

The script begins by importing the required libraries. It then identifies the total number of curves present in the part geometry. Nested *while* loops are used to traverse each curve and generate a specified number of evenly distributed points along it (Figure 3.6b). The number of points generated on each curve can be adjusted by modifying the value of the variable *p* defined in line 4 of the Python script.

DEFORM requires the material point file to follow a specific format, including the total number of points listed at the top of the file and an index value assigned to each point. In addition, point

coordinates generated in meters must be converted to millimeters. These operations are handled within the script, with loops used to store the processed data in a two-dimensional array. The array contents are then written to a *.dat* file and saved in the same directory as the script file. The Python script used for automating the material point creation procedure is shown in Figure 3.7.

```
# Python Script, API Version = V17
import csv
p=10.0                          #Number of points in a Curve
fileName='PointCloud.dat'       #Name of the .dat file
n=len(GetRootPart().Curves)     #Varibale n stores the number of curves in the sketch
j=0
while(j<n):                     #Loop to traverse along the curves
  x=0
  while (x<p) :                 #Loop to create points in a single curve
    # Create Datum Point
    selection = Selection.Create(GetRootPart().Curves[j])
    result = DatumPointCreator.Create(selection, (x*(1/p)), False, Info2)     #creating even distribution of points in a curve
    x=x+1
    # EndBlock
  j=j+1

points=[[0]*4]*((n*int(p))+1)     #initializing 2D array to store point coordinates
length=len(points) -1             #Storing total number of points created in the variable length
j=0
points[j]= [length,"","",""]      #Storing length as first row in the 2D array

with open(fileName, 'w',) as f:     #Opening file to write
  write = csv.writer(f)             #Initializing write
  write.writerow(points[j])         #Writing total number of points as 1st row in the file
  f.close()                         #Closing file

while(j<length):                    #Loop to assign coordinate values of points and write in file
  points[j+1][0]= j+1               #Index column of a point
  points[j+1][1]=float(GetRootPart().DatumPoints[j].Position.X*1000)    #X coordinate with units in mm
  points[j+1][2]=float(GetRootPart().DatumPoints[j].Position.Y*1000)    #Y coordinate with units in mm
  points[j+1][3]=float(GetRootPart().DatumPoints[j].Position.Z*1000)    #Z coordinate with units in mm
  with open(fileName, 'ab',) as f:       #Opening file to write
    write = csv.writer(f)                #Initializing write
    write.writerow(points[j+1])          #Writing point coordinates with index in the file
  j=j+1
```

Point Creation

Formatting point data and saving file as .dat

Figure 3.7: Python Script for Automating Material Point Creation Procedures.

Because the entire process is automated, the user is only required to import or sketch a 2D geometry and execute the script from the saved script files. The script automatically generates the point cloud and saves the corresponding coordinate data as a *.dat* file. This file can then be imported into DEFORM for subsequent material point tracking and analysis.

3.3 Comparison Between SpaceClaim and SolidWorks for Automation

Both SolidWorks and SpaceClaim offer scripting and automation capabilities that can be used for material point creation. However, SpaceClaim was preferred in this work due to its more streamlined geometry handling and scripting interface. SpaceClaim treats geometric entities such as lines, arcs, and fillets as independent curves, which simplifies automated traversal and point distribution along part boundaries. In contrast, SolidWorks often requires more complex feature-tree navigation and additional preprocessing steps to access and manipulate equivalent geometric entities. Furthermore, SpaceClaim's Python-based scripting environment allows direct access to geometry objects and faster execution of automation routines, making it particularly well suited for rapid and repeatable material point generation. These advantages result in reduced development time and improved robustness of the automation framework when compared to SolidWorks for the specific application considered in this study.

3.4 Preform Design for Three-Dimensional Parts

For two-dimensional (2D) forging problems, such as axisymmetric or plane strain conditions, preform design is relatively straightforward. These problems are easier to visualize, and the geometrical construction of the preform is constrained by linear and curvilinear boundaries. As a result, material flow patterns are more predictable, and the iterative refinement of the preform geometry can often be performed with limited computational and geometric complexity.

Extending the preform design methodology to three-dimensional (3D) forging geometries, however, introduces several additional challenges. These include: (a) Difficulty in generating an appropriate initial guess for an oversized 3D preform; (b) Increased complexity in reconstructing a continuous 3D geometry from scattered point data (point clouds); (c) Greater mesh refinement requirements in 3D FE models to avoid numerical instabilities and distorted preform shapes; and (d) The need for robust algorithms capable of handling large volumes of material point data during backward tracking and reconstruction.

Figures 3.8 and 3.9 illustrate the fundamental differences in material flow behavior between 2D and 3D forging processes. In 2D deformation, material points tend to move along constrained paths, typically toward die boundary lines. In contrast, material flow in 3D forging is governed by

more complex trajectories, with points moving toward three-dimensional die surfaces. These trajectories are influenced by multiple degrees of freedom, local strain gradients, frictional resistance, and evolving contact conditions with the die geometry.

The increased complexity of material flow in 3D forging makes preform design significantly more challenging. Material may preferentially follow paths of least resistance within the die cavity, potentially leading to uneven filling, excessive local deformation, or unintended flash formation if these effects are not properly captured. Consequently, establishing a robust preform geometry using backward-tracking methods requires careful treatment of material point generation, tracking, and geometric reconstruction.

Therefore, while the fundamental principles of the geometric resemblance methodology remain consistent with those outlined in Figures 3.1–3.3, additional procedures are required to enable its effective application to 3D forging problems. The following subsections present the material point generation, backtracking, and solid reconstruction techniques developed to support reliable and automated preform design for three-dimensional closed-die forging operations.

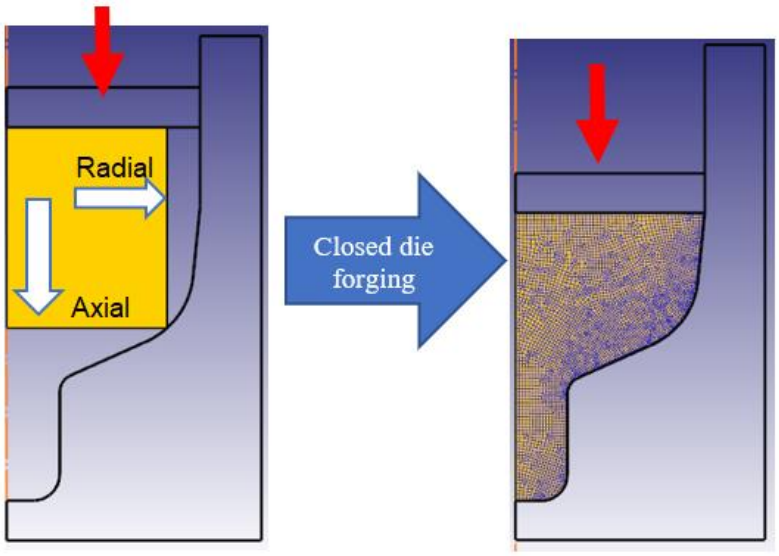


Figure 3.8: Material flow in 2-D Axisymmetric forging

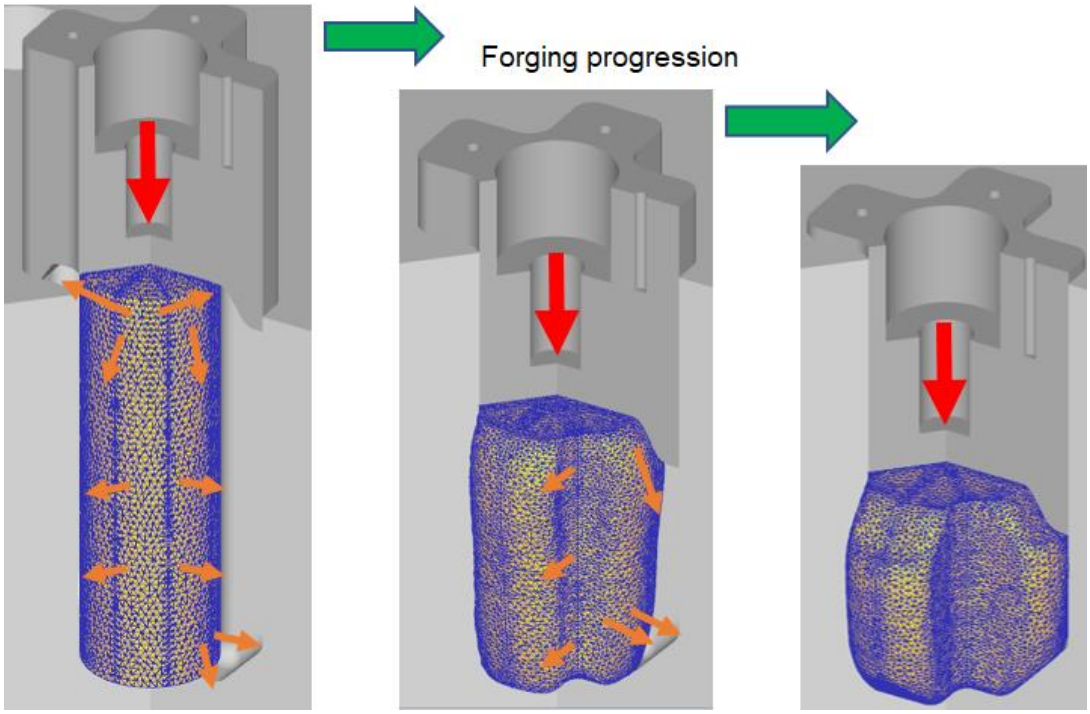


Figure 3.9: Complexity in 3-Dimensional Material flow in forging

3.4.1 Material Point Generation and Mapping for Backtracking

To implement the backward-tracking method in a three-dimensional finite element (FE) environment, material point tracking must begin with the accurate generation of representative surface or boundary points from the forged geometry. These material points serve as virtual markers that allow deformation paths to be traced backward from the forged part to the undeformed billet. In this study, DEFORM 3D is used for forging simulations, SpaceClaim is employed for CAD modeling and surface processing, and Python scripts are used for data formatting and automation. The overall procedure is summarized as follows.

(i) A solid three-dimensional part geometry is first created using CAD software and meshed for FE analysis. The meshed geometry is typically stored in STL (Stereolithography) format (Fig. 3.10a–b). An STL file represents the surface of a 3D object using a collection of triangular facets and is widely used for geometry exchange because it preserves surface shape while remaining computationally efficient.

(ii) The STL file is then exported as a .xyz file to generate a surface point cloud (Fig. 3.10c). In this step, the triangular surface representation is converted into a discrete set of points, each defined by its three-dimensional Cartesian coordinates. This conversion enables direct access to surface geometry information without reliance on mesh connectivity.

(iii) The resulting .xyz file contains a list of 3D coordinate data corresponding to surface points of the part (Fig. 3.10d). These coordinates are subsequently processed and reformatted using either Excel or Python scripts (Fig. 3.10e). This step allows filtering, indexing, and restructuring of the data to conform to the input requirements of the FE software.

(iv) The formatted coordinate data are saved as a .dat file, which is the required format for material point input in DEFORM 3D. This file is then imported into DEFORM to enable material point tracking through the simulated deformation process.

Once imported, the material points act as virtual tracers embedded in the FE model. During backward tracking, their positions are mapped from the deformed configuration to their corresponding locations in the undeformed billet. This mapping enables reconstruction of the preform geometry and provides the foundation for automated preform design within the geometrical resemblance framework.

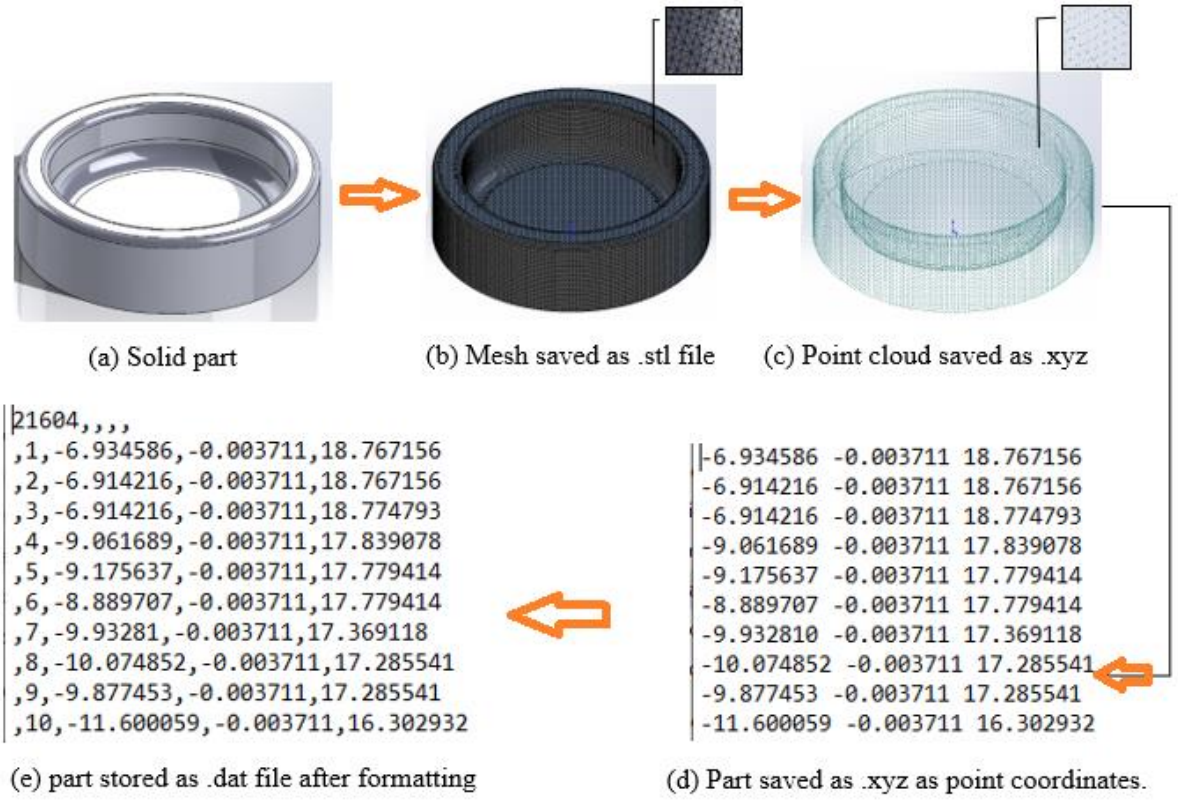


Figure 3.10: Material point creation and formatting

3.4.2 Reconstruction and Solid Modeling from Point Cloud Data

Following the backtracking procedure, the output consists of a set of three-dimensional coordinate points that define the boundary of the reconstructed preform geometry (Fig. 3.11a). These points collectively represent a *point cloud* describing the external surface of the preform. To enable further finite element (FE) simulations, this discrete point cloud must be converted into a continuous, watertight solid model that can be meshed and analyzed reliably.

The reconstruction and solid modeling procedure is carried out as follows:

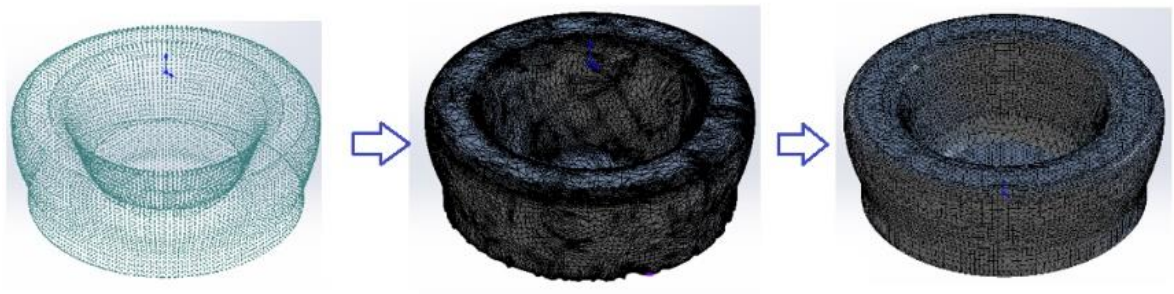
(i) The backtracked material points obtained from DEFORM 3D are first exported and saved as a .xyz file. This file contains the three-dimensional Cartesian coordinates of the surface points that define the preform boundary.

(ii) The .xyz file is then imported into SolidWorks using the *Mesh Prep Wizard*, where an initial surface mesh is generated from the point cloud data (Fig. 3.11b). During this step, the discrete points are connected through surface triangulation, forming a polygonal representation of the preform geometry. This triangulated mesh provides a preliminary geometric approximation but is not yet suitable for direct FE meshing due to potential inconsistencies.

(iii) The initial mesh often contains geometric imperfections such as overlapping or inverted triangles, non-manifold edges, gaps, or irregular element distribution. These issues arise from numerical noise in the backtracking process and variations in point density. To address these challenges, the mesh is imported into SpaceClaim, where mesh healing operations are performed (Fig. 3.11c). These operations include smoothing, gap closure, removal of degenerate elements, and re-meshing to ensure geometric continuity and topological consistency.

(iv) After mesh repair and validation, the cleaned geometry is exported as an STL file, representing a closed and manufacturable surface suitable for FE meshing. This STL file is then used as the input geometry for the next iteration of the finite element simulation within the geometrical resemblance framework.

This reconstruction step serves as a critical bridge between simulation-derived data and practical geometry modeling. Successful execution requires an understanding of geometric modeling concepts such as *point cloud representation*, *surface triangulation*, *mesh healing*, and *solid body reconstruction*. Ensuring geometric integrity at this stage is essential for achieving stable meshing, accurate material flow prediction, and convergence in subsequent FE simulations.



(a) (b) (c)

Figure 3.11: (a) backtracked preform represented in points, (b) point cloud converted mesh, (c) Smoothened and remeshed preform

3.4.3 Summary of Automated Backtracking and Reconstruction Workflow

Together, Sections 3.4.1 and 3.4.2 define an automated pipeline for material point generation, backward tracking, and geometry reconstruction within the geometrical resemblance framework. Starting from a simulated forged geometry, representative surface points are systematically generated, formatted, and imported into the finite element environment to enable backward material point tracking. The resulting backtracked point cloud is then converted into a clean, watertight solid model through automated surface reconstruction and mesh healing procedures. This closed-loop process seamlessly connects finite element simulation, data-driven geometry extraction, and CAD-based solid modeling, enabling iterative preform reconstruction with minimal user intervention. By integrating point cloud processing and solid model regeneration into a unified workflow, the methodology enhances repeatability, reduces manual effort, and supports scalable application to complex three-dimensional forging problems.

With the automated backtracking and reconstruction framework established in Chapter 3, the following chapter applies the proposed preform design methodology to representative three-dimensional closed-die forging processes to demonstrate its effectiveness, robustness, and practical applicability.

Chapter 4:

APPLICATION OF THE PREFORM DESIGN METHODOLOGY FOR 3D CLOSED-DIE FORGING PROCESSES

This chapter presents the application of the proposed geometric resemblance–based preform design methodology to representative three-dimensional closed-die forging problems. Two industrially relevant forging case studies with complex geometries are examined: (1) a cross-joint and (2) a three-lobed drive hub. These case studies were selected to demonstrate the robustness of the methodology when applied to parts featuring non-uniform cross sections, complex material flow paths, and high sensitivity to flash formation.

In each case, the objective is to design an effective preform that ensures complete die filling while minimizing flash generation and forging load requirements. The automated backtracking, material point mapping, and geometry reconstruction procedures developed in Chapter 3 are integrated into the preform design process, enabling systematic refinement of the preform geometry through iterative finite element simulations. The results illustrate the capability of the proposed approach to support practical preform design for complex three-dimensional forging applications.

4.1 Cross-Joint Forging

The preform design methodology based on geometrical resemblance was applied to determine a preform for a cross-joint forged from AISI 1016. To evaluate flash formation during single-stage forging, finite element (FE) simulations were conducted using the DEFORM software package. The initial billet measured 50 mm in diameter and 60 mm in length. Forging was performed at a billet temperature of 1000°C and die temperature of 250°C, with a friction factor of m = 0.3 at the die–workpiece interface as given in Table 1. The workpiece's material flow stress behavior was modeled using data from the DEFORM-3D software's built-in material library, which is routinely maintained by the software developers and widely used for hot forging applications. Figure 4.1a shows the initial cylindrical billet, and Figure 4.1 (b) depicts the resulting forged cross-joint with significant flash formation.

Table 1: FE Simulation Conditions for Three-Lobe Drive Hub Forging

Parameter	**Value/ Description**
Material Model	AISI 1016 (sourced from DEFORM Material Library)
Billet Temperature	1000 °C
Die Temperature	250 °C
Interface Friction Model	Shear friction law (friction factor m=0.3)

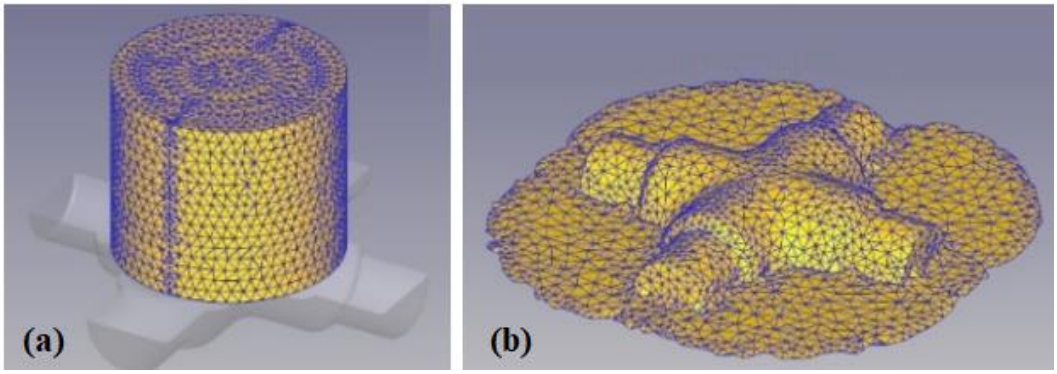


Figure 4.1: (a) Initial cylindrical billet, and (b) Forged cross joint with flash.

To reduce material waste due to flash exhibited in Figure 4.1 above, the preform design methodology based on geometrical resemblance (outlined in Section 2.1) was used. Larger shapes, Xb and Xc, were constructed, and an estimated preform Pc was established, as illustrated in Figure 4.2.

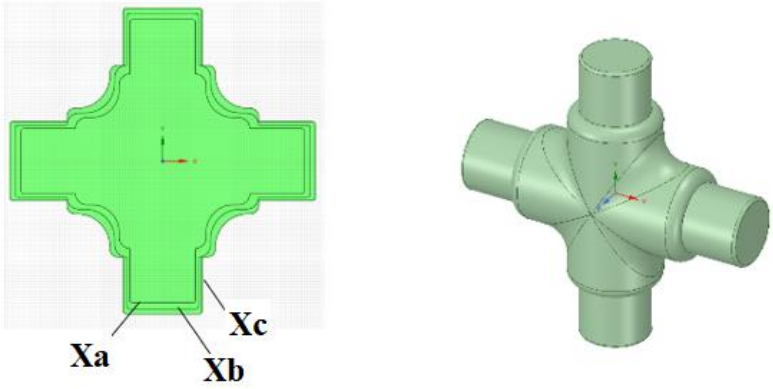


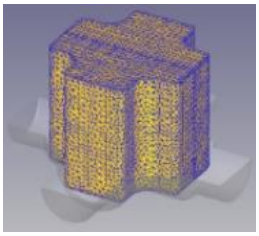
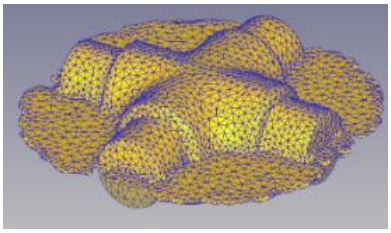
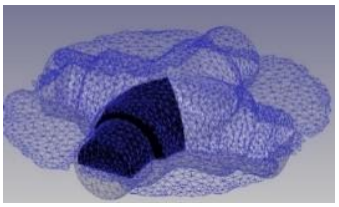
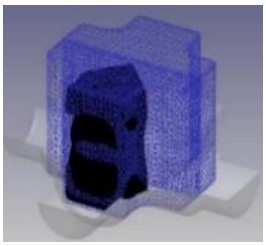
Figure 4.2: Series of parts constructed to resemble the tooth adapter and the desired part Xa.

FE iterations for preform determination are shown in Figure 4.3. Due to symmetry, only a half 3D FE model was simulated. In the first iteration, the expanded baseline geometry Pc was used to

forge a larger part Xc as shown in Fig. 4.3A(b). Subsequently, the boundary surface of the target part, Xb, was mapped onto Xc to identify the region of interest for backtracking, as depicted in Fig. 4.3A(c).

Interpreting Figures 4.3A(c) and 4.3A(d) require some spatial reasoning, as they involve curved 3D surfaces. In Fig. 4.3A(c), the *dark blue* region represents material points corresponding to one-eighth of the boundary surface of the target cross-joint Xb, superimposed onto Xc. This 1/8th sector is selected for clarity and computational efficiency, leveraging the multiple symmetry planes inherent to the cross-joint geometry. These mapped material points are then backtracked through the deformation field to reconstruct the original point positions, thereby generating the preform geometry Pb. The resulting point cloud is shown in *dark blue* in Fig. 4.3A(d). This 3D procedure closely parallels the 2D backtracking illustration provided earlier in Fig. 3.1(C & D), where dark-colored lines were used to indicate mapped and traced boundaries. The consistency between the 2D and 3D examples reinforces the generality of the proposed physics-informed approach.

A second iteration started by forging Xb using preform Pb, the simulation results of which are shown in Fig. 4.3B(f). By backtracking material points from Xb, Pa can be obtained. The last FE iteration involved forging preform Pa, resulting in part Xa with a minimal amount of flash [Fig. 4.3C(j)]. The uniformity in the material flow observed in the final part suggests that an additional FE iteration will completely eliminate the flash.

<table>
<tr><td rowspan="2">Iteration # 1</td><td> 
(a) Guessed Pc</td><td> 
(b) Forged part Xc</td></tr>
<tr><td> 
(c) Points of Xb on Xc</td><td> 
(d) Pb of Xb- initial step</td></tr>
</table>

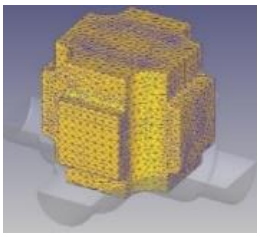
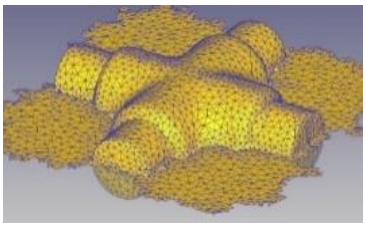
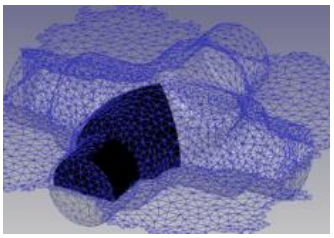
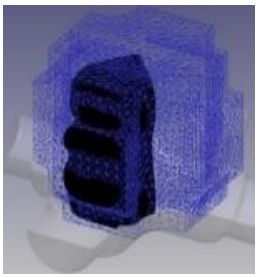
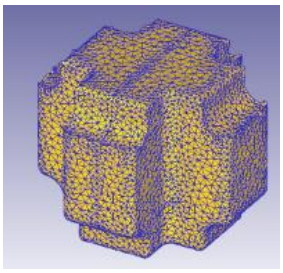
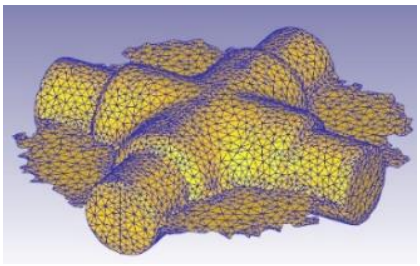
Iteration # 2	 (e)Pb smoothened	 (f) Forged part Xb
	 (g)Points of Xa on Xb	 (h) Pa of Xa- initial step
Iteration# # 3	 (i) Pa smoothened	 (j) Xa final step of forging

Figure 4.3: FE iterations and material backtracking to obtain a preform for the Cross joint

Fig. 4.4 shows the effective strain distribution for forging the cross-joint using the established preform. Flash regions exhibited maximum effective strains around 7.4, while the bulk material exhibited strains near 5. Forging loads for both the preforming and final forging stages are shown in Figure 4.5. The maximum load obtained for preforming was 480 tons, whereas the maximum load for the final stage was 1000 tons. It should be noted that at the end of the stroke, the load jumps, indicating that the actual forming load should be lower than what is reported here. It can be observed that by using the intermediate preform stage, the volume of the billet is reduced by approximately 13.5%

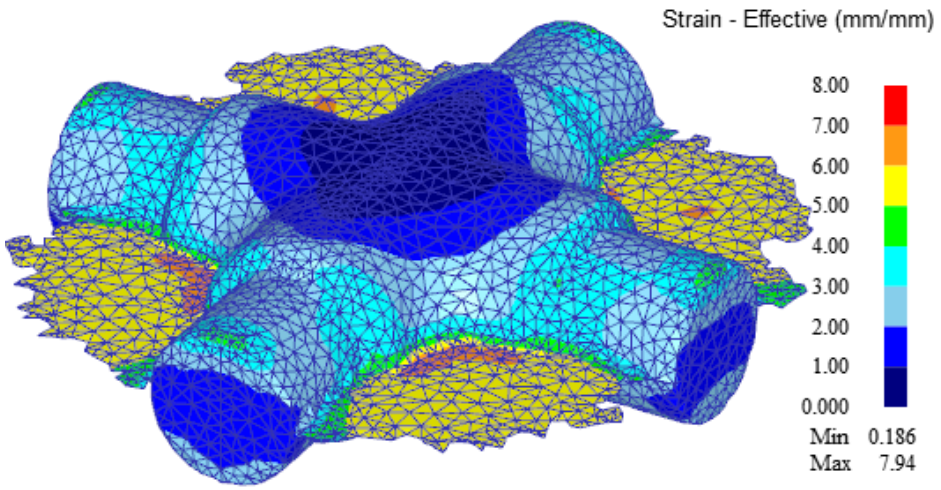


Figure 4.4: Effective strain distribution

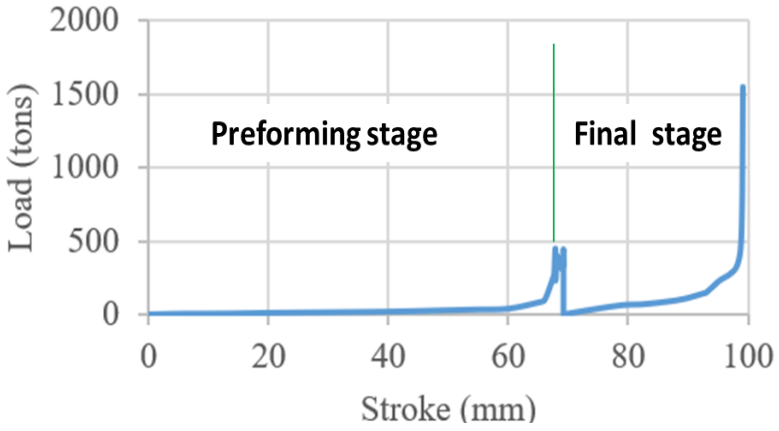


Figure 4.5: Forging load, preforming and final forging

While constructing the initial expanded baseline geometry is relatively straightforward for simple parts, significant challenges arise when dealing with complex geometries that exhibit intricate three-dimensional material flow. In such cases, achieving an adequately oversized preform shape may necessitate non-proportional scaling of boundary surfaces rather than uniform expansion. In this study, all geometries tested employed proportional scaling; however, future applications may require more sophisticated scaling strategies to accommodate localized geometric features. It is also important to recognize that, despite being informed by the underlying physics of deformation, some backtracked preform shapes may prove impractical to forge due to geometric intricacies or manufacturing constraints. In such instances, the initial expanded geometry may need to be revised or simplified to ensure manufacturability while preserving the intended flow characteristics.

Furthermore, during the preform reconstruction for the cross-joint forging case, surface smoothing was applied to the backtracked geometry due to insufficient point cloud resolution, highlighting the importance of adequately sampled data in obtaining usable CAD models.

4.2 Three-Lobe Drive Hub Forging

The preform design methodology was further applied to a three-lobe drive hub featuring lobes on one side and an axisymmetric cup shape on the other. As the forging is carried out using a cylindrical billet, the deformation of the material during forging is such that the three lobes will be extruded out on one side, and the other side will feature backward cup extrusion. Due to the presence of three lobes, the material flow exhibits complex three-dimensional flow. Figure 4.6 shows the target hub geometry (Xa) and its geometrically enlarged counterparts (Xb and Xc) which were used in the preform search scheme. One of the goals for this case study was to utilize the preform design methodology based on geometrical resemblance to establish a preform that will result in minimal flash formation.

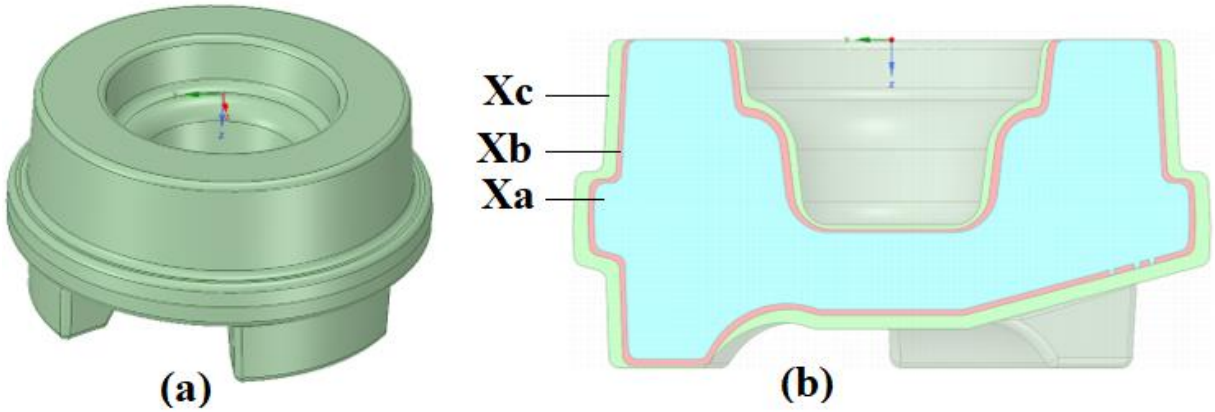


Figure 4.6: (a) desired drive hub Xa, (b) geometrically resembling larger shapes

The procedures for determining the preform die are the same as those used for the cross-joint forging presented earlier. The study was conducted on AISI 1035 cylindrical billet material measuring 48 mm in diameter and 100mm in height. Forging simulations were performed at 1200°C for the billet, 150°C for the dies, and with a friction factor of m = 0.3 [Table 2]. Due to symmetry, only one-sixth of the part was modeled.

Table 2: FE Simulation Conditions for Three-Lobe Drive Hub Forging

Parameter	**Value/ Description**
Material Model	AISI 1035 (sourced from DEFORM Material Library
Billet Temperature	1200 °C
Die Temperature	150 °C
Interface Friction Model	Shear friction law (friction factor m=0.3)

In the first iteration (Figure 4.7A), a pancake-shaped billet was employed as the initial expanded geometry or oversized billet, to forge the intermediate part Xc. The boundary of the target geometry, Xb, was then overlaid onto Xc to identify the relevant deformation region. By backtracking the material points from Xc, the corresponding preform geometry, Pb, was reconstructed. This mapped configuration became the initial estimate of the preform shape. As with the earlier cross-joint case presented in Section 3.1, interpreting Figures 4.7A(c) and 4.7A(d) requires some spatial visualization, given the complexity of 3D surface mapping. In Fig. 4.7A(c), the *dark blue* region represents 1/6th of the boundary surface of Xb, projected onto the forged part Xc. This fraction was selected to reflect the part's threefold rotational symmetry. In Fig. 4.7A(d), the *dark blue* region illustrates the backtracked material points, traced from Xc back to the initial billet, which defines the geometry of preform Pb. This approach mirrors the conceptual strategy outlined in the 2D example of Fig. 3.1(C & D), but adapted here to a three-dimensional, lobed geometry.

The second iteration starts with meshing Pb obtained from iteration 1 as shown in Fig.4.7B. However, the geometry from back tracing required smoothing. It should be noted that the Pb is extracted from inside the bulk of Pc as a point cloud and must undergo triangulation as discussed in sections 2.2.1 and 2.2.2. During reconstruction tiny geometric features may be lost in the process. After Xb part is simulated, Xa is overlaid on the simulation results, followed by tracing back the material points of Xa onto Xb. Backtracking these points produced the final preform Pa. Figure 4.7C(i) shows Pa, with the three lobes oriented upward. Figure 4.7C(j–k) shows forgings obtained from the rougher and finisher dies for the three-lobe drive hub.

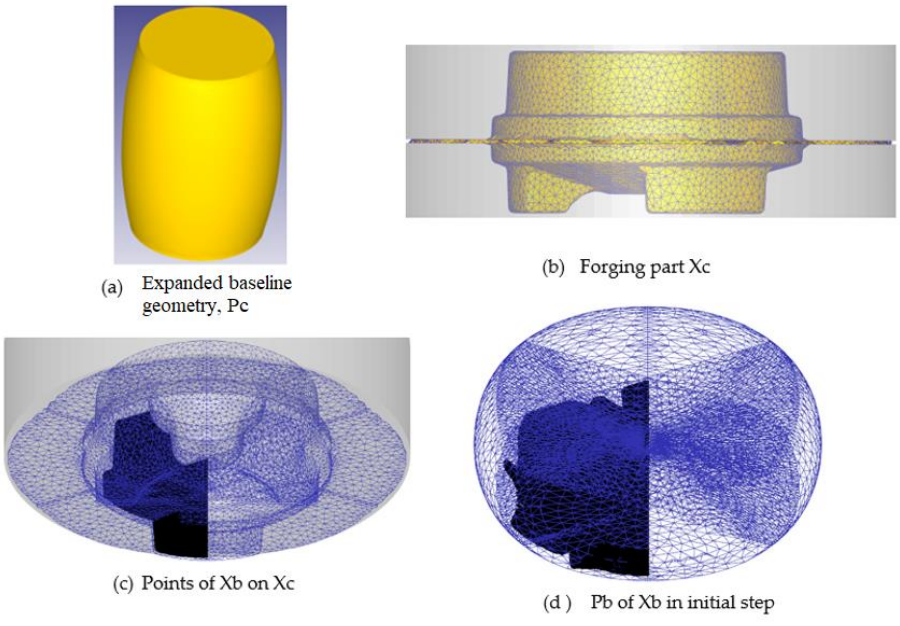


Figure 4.7A. FE iteration #1: Material backtracking to obtain a preform for the three-lobe drive hub.

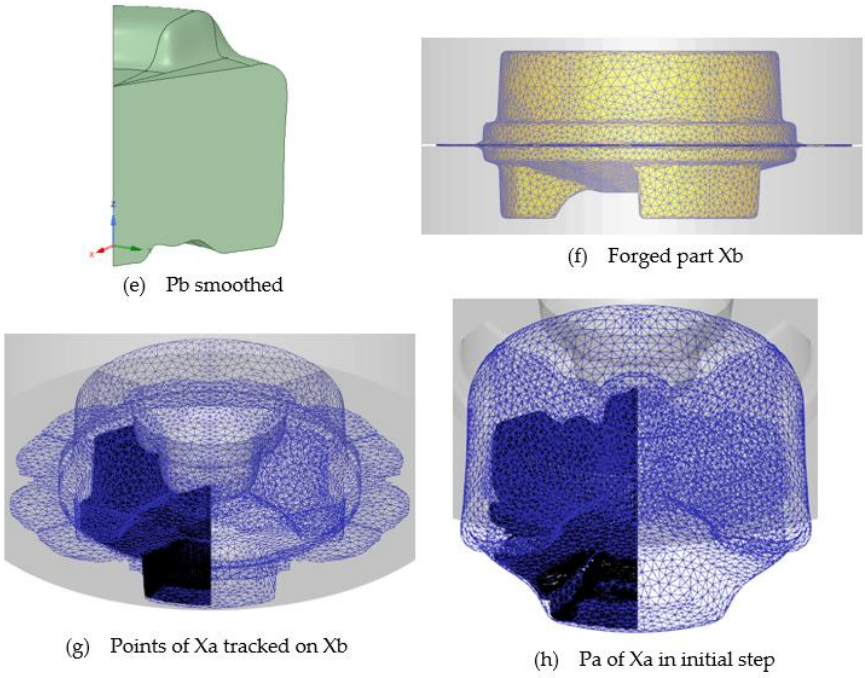


Figure 4.7B. FE iteration #2: Material backtracking to obtain a preform for the three-lobe drive hub.

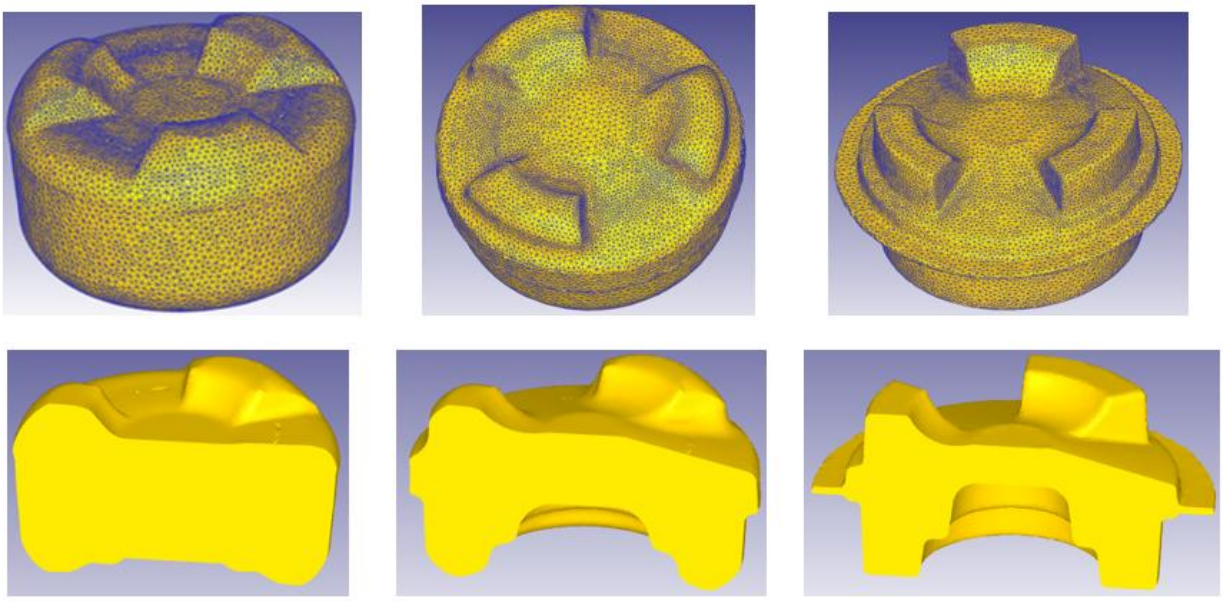


Figure 4.7C. Final preform and final forging of the three-lobe drive hub.

Effective strain distributions for preforming and final forging are shown in Figure 4.8. The preform resulted in a maximum effective strain of 2.81, where the final stage forging exhibited a maximum strain of 4.85. The forging load versus stroke is shown in Fig. 4.9, where the maximum forging load for preforming was 130 tons, and for the final stage, the force rose to 545 tons. Although minor flash remained at the end of forging, the flash volume was significantly reduced compared to conventional forging practices. Further refinements could achieve fully flashless forging.

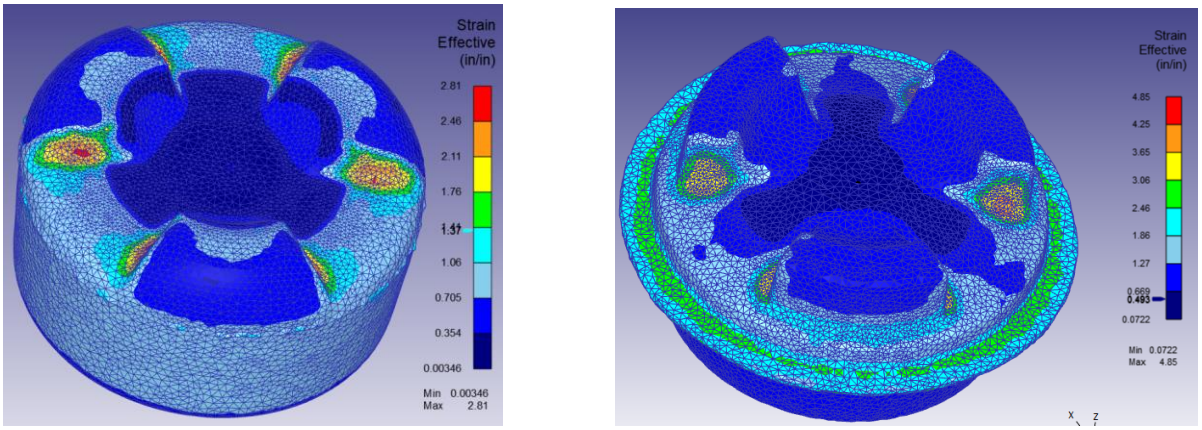


Figure 4.8. Strain distribution: (a) preform and (b) finish forging.

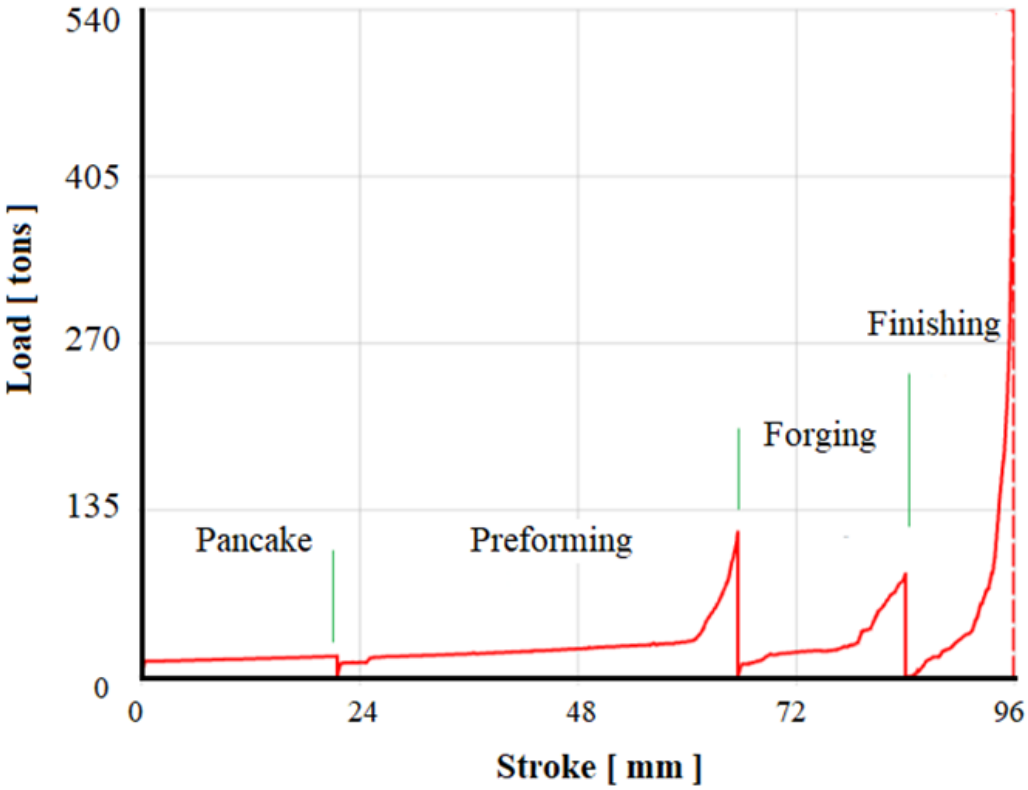


Figure 4.9. Forging load evolution for hot forging of a three-lobed drive hub.

Chapter 5:

EXPERIMENTAL VALIDATION

As discussed in the previous sections, the goals for preform forging are threefold: (a) eliminate flash during forging, (b) reduce flash formation (near-flashless forging), and (c) enable the forging of complex parts that would otherwise be impossible to forge in a single stage. It is important to evaluate the efficiency of the preform design methodology to determine how well the preforming stage contributes to reducing flash and improving overall forging efficiency.

To validate the effectiveness of the proposed preform methodology, forging tests were conducted on a downscaled cross-joint component made from AL6061. This validation was primarily aimed at minimizing material utilization by reducing the amount of flash. Preliminary finite element (FE) simulations were carried out for different billet sizes to establish the billet volume limit below which a cross-joint could not be forged successfully in a single stage

5.1 Preliminary FE Forging Simulations

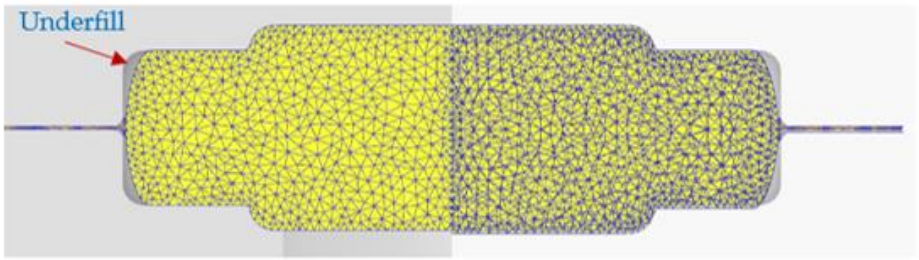
Several forging simulations were conducted to determine the billet volume threshold that would require multi-stage forging for an AL6061 cross-joint. Billets with a diameter of 11 mm and varying lengths were evaluated. These simulations were necessary because cross-joints can typically be forged with flash in a single stage, and the major benefit of preform use is to reduce flash while maintaining part soundness. The FE simulation conditions are summarized in Table 3. The simulation workflow accounted for billet cooling during manual transfer from the heating furnace to the forging dies, with an estimated transfer time of 15 seconds. A constant shear friction model was employed, using a friction factor of $m = 0.3$, which is representative of hot forging conditions with copper-based lubricants.

Table 3: FE Simulation Conditions for Cross- joint Forging

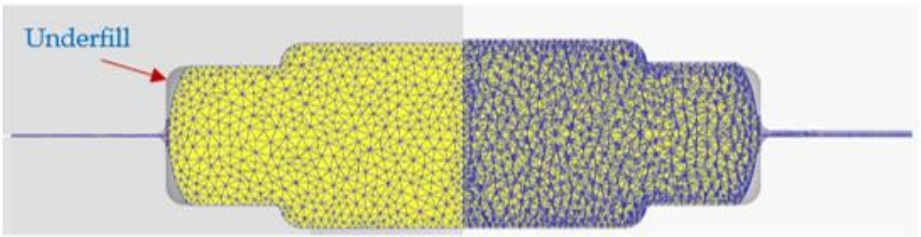
Parameter	**Value/ Description**
Material Model	AL6061 (sourced from DEFORM Material Library)
Billet Temperature	400 °C
Die Temperature	150 °C
Interface Friction Model	Shear friction law (friction factor m=0.3)
Billet transfer	15 seconds set for billet heat loss (air heat transfer)

Figure 5.1 summarizes the simulation results for three billet sizes: (a) 3478 mm³ (110% of the net part volume), (b) 3379 mm³ (107% of the net part volume), and (c) 3326 mm³ (105% of the net part volume). At 110% billet volume [Fig. 5.1A(a)], the cross-joint was fully formed but with approximately 10% material lost as flash. Reducing the billet size to 107% and 105% [Figs. 5.1A(b) and 5.1A(c)] led to underfill, indicating that successful forging with these billet sizes would require the use of a preform. Using the preform methodology discussed in Section 3, a sound cross-joint was produced in two stages, as shown in Figure 5.1B.

(a) 110% of the desired part volume (3472 mm³): Fully formed



(b) 107% of the desired part volume (3379 mm³): Under fill



(c) 105% of the desired part volume (3326 mm³): Under fill

Figure 5.1A. Influence of billet volume on single-stage forging of the cross joint.

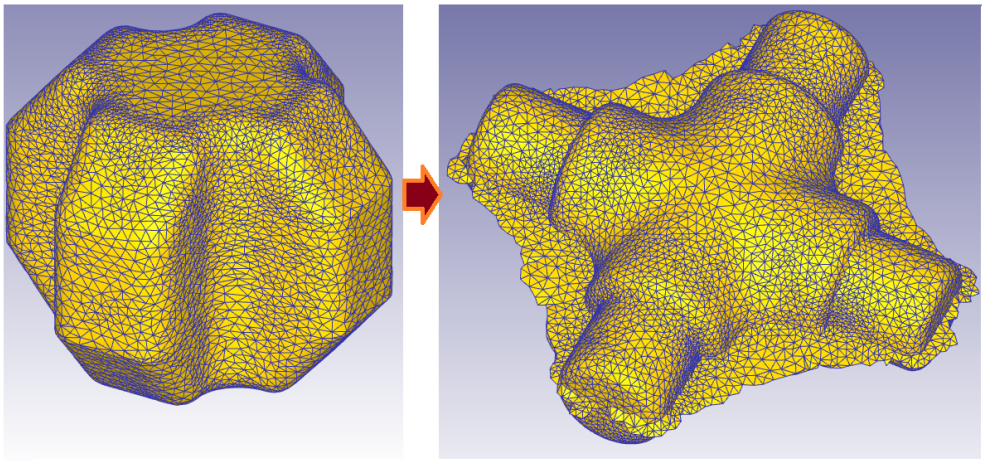


Figure 5.1 B. Two-stage forging for 105% billet volume(3326 mm³): Fully formed and reduced flash

The cross-joint example demonstrates that by introducing a preform stage, at least 5% of billet material can be conserved. While this comes with the added complexity of a multi-stage forging process, the improvement in material utilization can be substantial, particularly for geometrically complex forgings where flash losses are typically significant. In such cases, preforming may be the only technically viable approach to achieving full die fill and sound part formation. The experimental validation presented below was based on the FE model shown in Figure 5.1B.

5.2 Laboratory-Scale Forging Tests at NC State University

5.2.1Test Setup Design

To experimentally validate the proposed preform design methodology, downscaled cross-joint components were forged in two stages using a 150-ton hydraulic press. Based on the finite element (FE) simulation results, dies and punches for both the preforming and final forging stages were designed and fabricated from H13 tool steel and heat treated to a hardness of 56 HRC to ensure sufficient strength and wear resistance under forging conditions.

The preforming die set is shown in Figure 5.2. In this configuration, a cross-hole in the top die is used to guide the punch during forging. The bottom of the punch contains a cavity that shapes the preform geometry. The top cylinder is rigidly fastened to the punch and transfers the forging load from the ram of the hydraulic press to the tooling. The final forging die set, shown in Figure 5.3,

consists of matched top and bottom dies with cavity geometries corresponding to the final cross-joint shape. Both die sets incorporate multiple through-holes for securing the tooling assembly. A set of precision-aligned holes located near the die cavities is specifically designed for accurate alignment of the top and bottom dies using dowel pins, ensuring proper die closure and repeatable forging conditions.

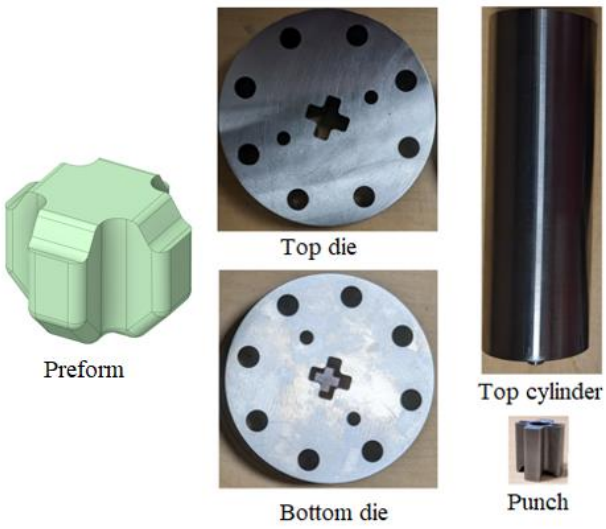


Figure 5.2. Preforming die set.

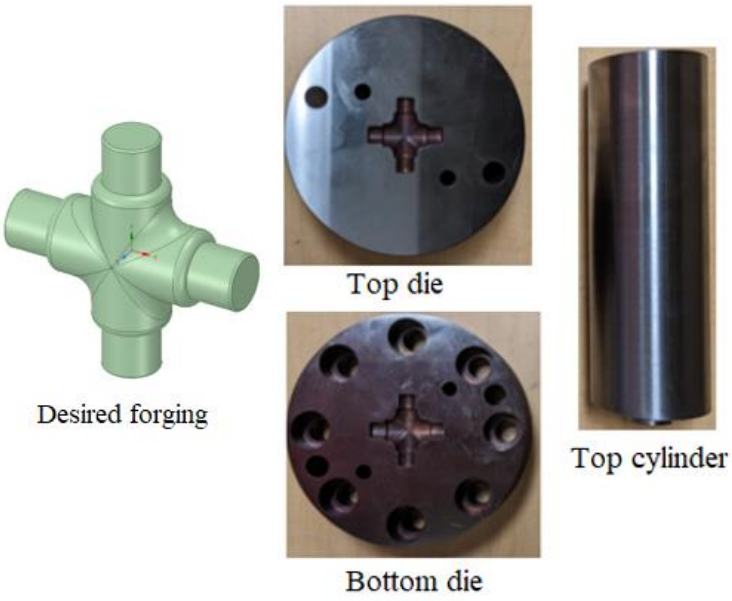
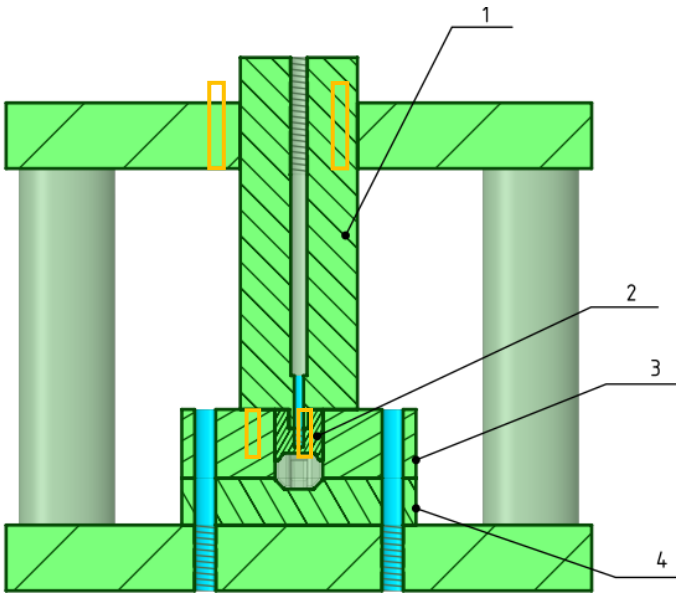


Figure 5.3. Final stage forging die set.

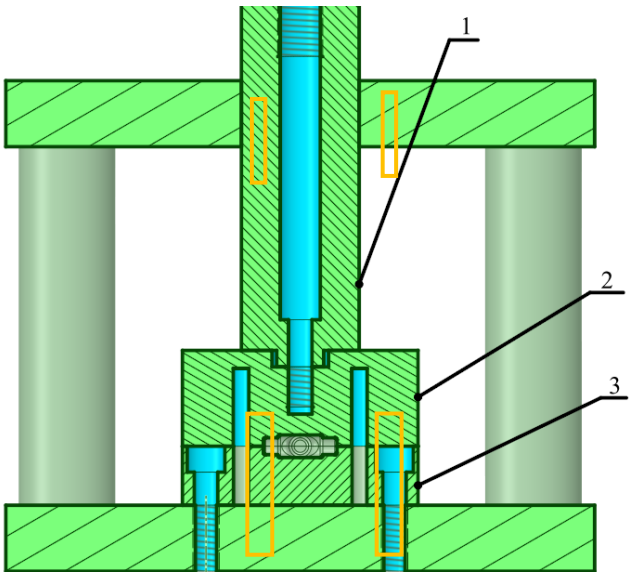
A CAD model of the preform die assembly is shown in Figure 5.4. The dies are mounted within a four-column die shoe to provide structural rigidity and maintain alignment during forging. Similarly, Figure 5.5 shows a CAD model of the finisher die assembly. During the forging experiments, the die shoe was securely fastened to the bed of the 150-ton hydraulic press available at NC State University. A photograph of the assembled tooling mounted on the press bed is shown in Figure 5.6.

Because the forging trials were conducted at elevated temperatures, both the top and bottom dies were preheated to approximately 150 °C using band heaters, as illustrated in Figure 5.6. The billets were independently heated in an electric furnace prior to forging. To minimize heat loss through metal-to-metal contact and promote more uniform die temperatures, a thermal insulation mat was placed between the bottom die and the tooling base.



Item No.	Preform die components
1	Cylinder punch holder
2	Top preforming Punch
3	Preforming Middle Die
4	Preforming Bottom Die

Figure 5.4: Upper and lower preform dies assembly for the cross joint forging, CAD



2	Finisher Top Die
3	Finisher Bottom Die

Figure 5.5: Upper and lower finisher die assembly for the cross joint forging, CAD.

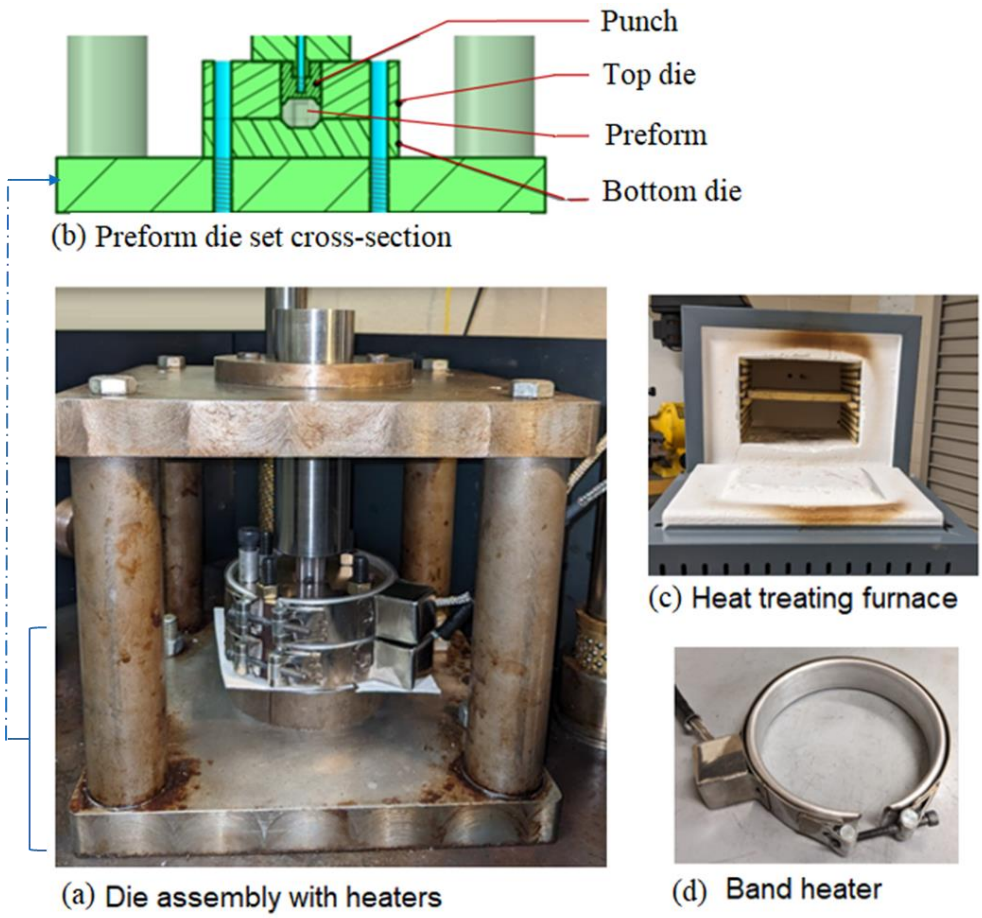


Figure 5.6. Test setup assembly.

5.2.2 Stress Analysis of the Die Assembly

Structural stress analyses of the major components of the die assembly were conducted using both ANSYS Workbench and DEFORM software packages. The forging loads applied in these analyses were extracted directly from the FE simulations of the cross-joint forging process discussed earlier in the report. These loads were then applied to the structural models to evaluate tooling integrity under representative operating conditions.

Although the experimental forging trials were performed using AL6061 aluminum billets only, stress analyses were carried out for both aluminum and steel billet materials to evaluate potential worst-case loading scenarios. The results indicate that forging steel billets induces significantly higher stresses in the die components, with maximum stress levels on the order of 1880 MPa, whereas the maximum stress observed when forging aluminum billets was approximately 1270 MPa. These results are summarized in Table 4.

The elevated stress levels associated with forging steel billets approach or exceed the yield strength of the tooling material and may result in localized plastic deformation. This outcome underscores the importance of accurate load prediction, material selection, and structural verification during die design, particularly when tooling is intended to accommodate multiple billet materials or forging conditions.

Table 4: Summary of die stress and displacement.

	Aluminum Billet (Al 6061)		**Steel Billet (AISI- 1016 steel)**	
	Maximum Stress	**Maximum Displacement**	**Maximum Stress (MPa)**	**Maximum Displacement**
Preforming- Top Punch	1270	0.052	1840	0.030
Preforming- Bottom Die	425	0.008	789	0.018
Finisher forging- Top	923	0.056	1570	0.125
Finisher forging- Bottom	702	0.038	1880	0.097

5.2.3. Test Procedures

Cylindrical AL6061 billets were heated to 400°C in a furnace and soaked for 15 minutes to ensure uniform temperature distribution. The dies were simultaneously heated to 150°C. A copper paste lubricant was applied to the die surfaces before forging. Once the target temperatures were reached, billets were transferred using tongs into the preform dies, and the ram was immediately activated.

In industrial settings, preforming and final forging are performed sequentially without reheating. However, due to laboratory limitations, multiple billets were preformed first, then reheated to 400°C before the final forging step was carried out.

5.2.4. Test Results and Discussions

Figures 5.7(a–c) present the forged preforms, the final forged cross-joint, and a cross-sectional view, respectively. Visual inspection revealed no surface defects or underfill. A total of nine cross-joints were forged, and dimensional measurements, including overall part dimensions and flash thickness, were obtained using micrometers. The results demonstrated excellent repeatability across all samples. The final forged cross-joint exhibited full die filling, with no visible laps, folds, or discontinuities.

Figure 5.8 compares the material flow behavior predicted by the FE simulations with that observed in the preforms and final forged parts. The flash formation in the experimental samples closely matched the simulation predictions, indicating good agreement between numerical and physical outcomes. Strain distribution maps for both the preform and final forging are shown in Figure 5.9. During the preforming stage, the maximum effective strain was approximately 2.66, while the final forging reached a maximum strain of 7.2. However, the bulk of the cross-joint exhibited an average effective strain of around 3.0.

The forging load profiles from both FE simulations and experiments are presented in Figure 5.10. During the physical experiments, the peak load for the preforming stage was approximately 12 tons, while the final forging stage reached a peak load of 40 tons. In contrast, the FE simulations predicted a peak preforming load of 7 tons and a final forging load of 32 tons. Although the differences in peak load between the simulations and experiments are notable, it is important to emphasize that these peak values occur near the end of the stroke, precisely when the material is

expelled from the die cavity as flash. At this stage, even a small variation in flash thickness can lead to a significant discrepancy in predicted load.

Overall, the experimental loads were consistently higher than those predicted by the simulations. A plausible explanation for this discrepancy is the assumed friction condition in the FE model. All simulations used a constant shear friction model with a friction factor of $m = 0.3$. While this value is considered reasonable for hot forging with copper-based lubricants, it may underestimate actual interface friction. In practice, frictional behavior in closed-die forging can vary substantially due to factors such as die temperature, surface oxidation, lubricant breakdown, and contact pressure. These variations can significantly affect forming loads, especially in the late stages of the stroke.

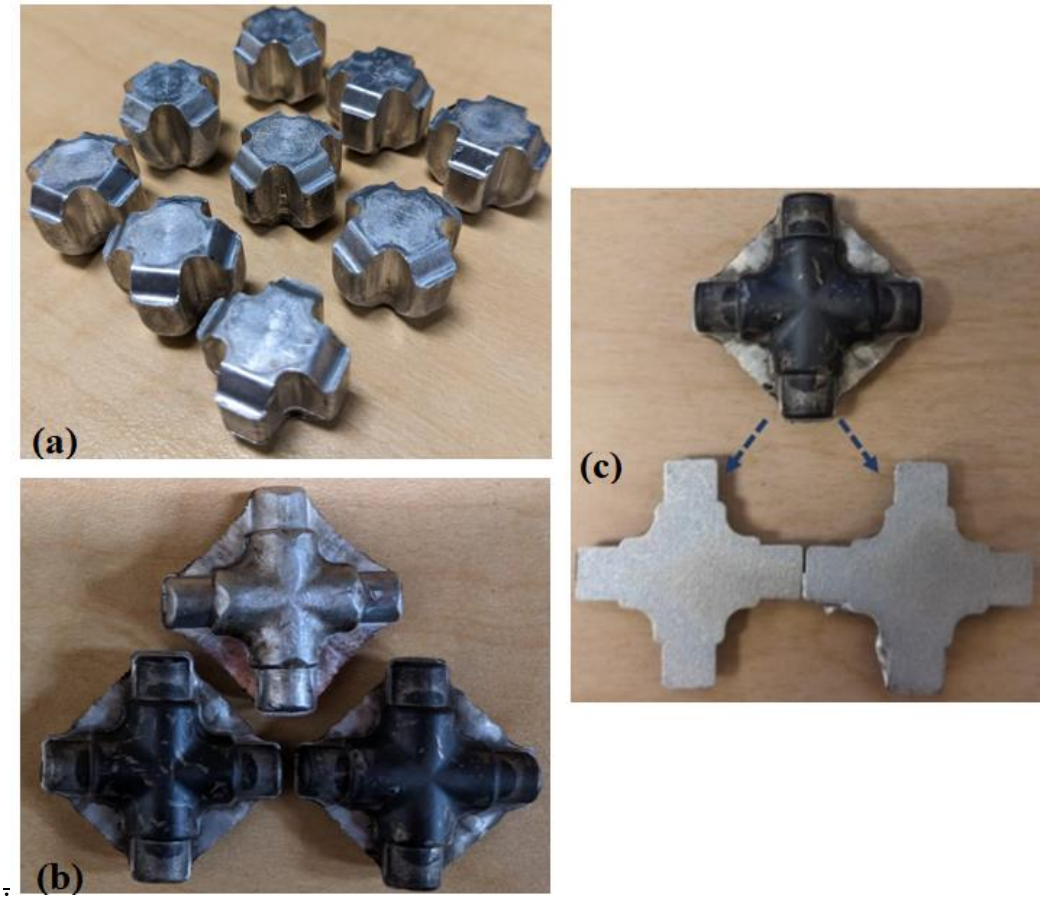
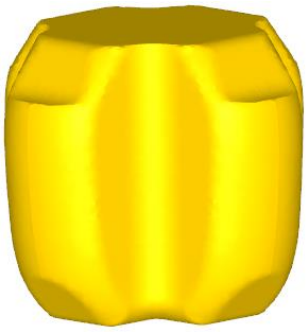
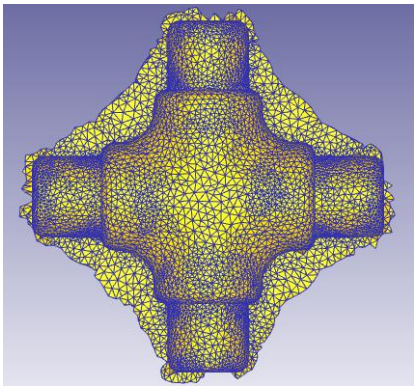


Figure 5.7. (a) Preforms, (b) forged Cross-joint, (c) cross-sectional view.

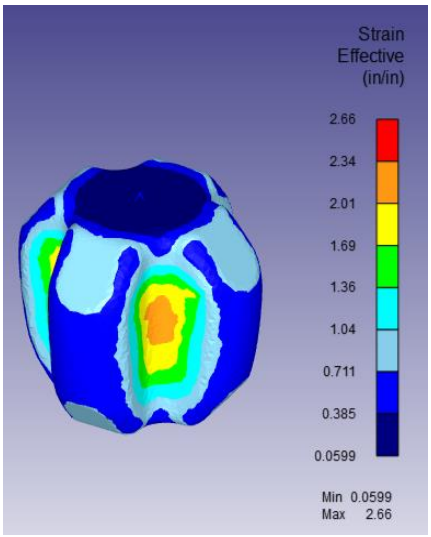


a) FE Simulation results, preform b) Experimental, preform



d) FE Simulation, final stage forging, e) Experimental, forged cross joint.

Figure 5.8. Result comparison between simulations and experiments for both preforming and final forging.



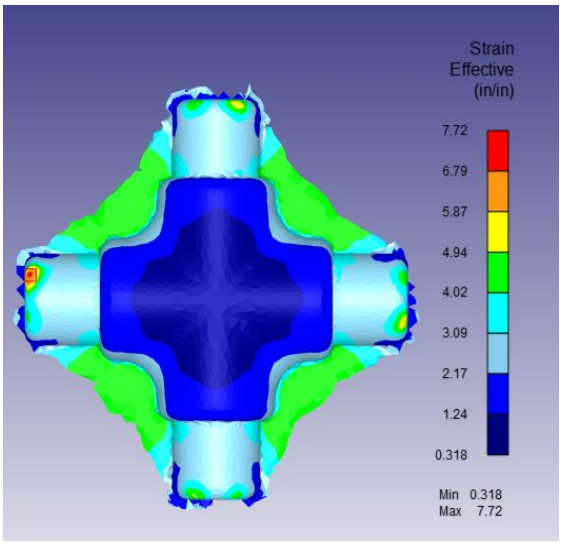


Figure 5.9: Effective strain distribution observed during preforming and final forging stage.

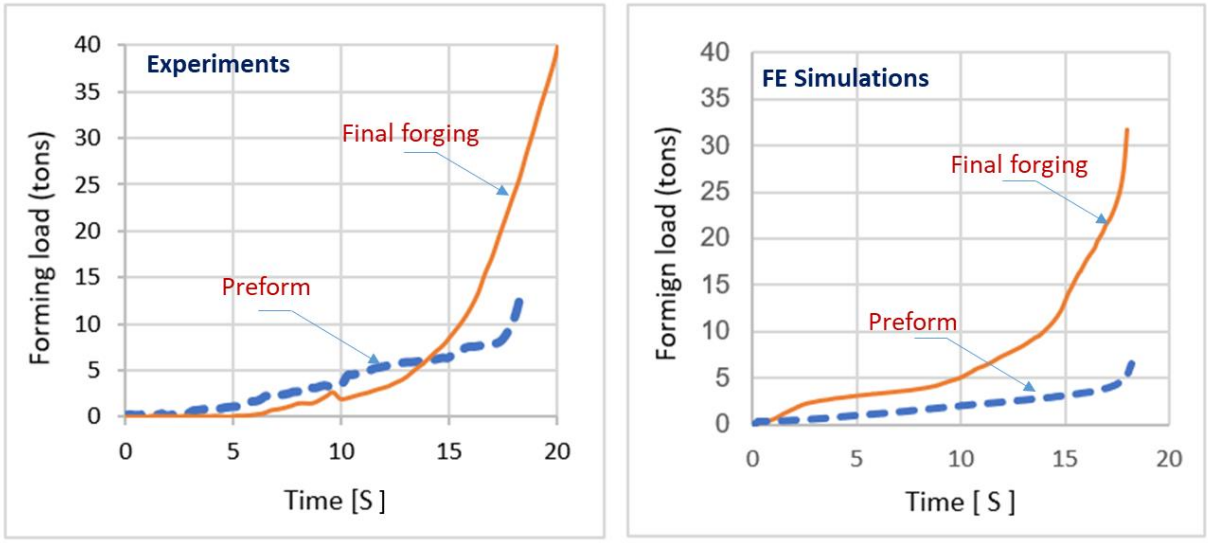


Figure 5.10. Comparison of the preform and forging loads between experiments and FE simulations.

5.2.5 Experimental Considerations and Test Limitations

While the laboratory-scale forging tests provided valuable experimental validation of the proposed preform design methodology, several considerations should be noted. First, the experiments were conducted on a hydraulic press with controlled stroke and loading conditions, which differ from the dynamic characteristics of mechanical presses commonly used in industrial forging. As a result, direct translation of press parameters between laboratory and industrial environments requires careful calibration.

Additionally, the experiments were limited to AL6061 aluminum billets due to tooling stress constraints identified in the structural analysis. Although this material selection was appropriate for validating geometric accuracy and material flow behavior, further testing with higher-strength billet materials would require reinforced tooling designs. Finally, thermal conditions during laboratory forging were simplified relative to industrial production, where continuous operation and higher throughput may result in different thermal gradients. These considerations will guide future experimental refinements and support scaling of the methodology to industrial forging applications.

5.3. Preliminary Trials Carried Out at Cornell Forge

In addition to the laboratory-scale experiments discussed in Section 5.2, one of the objectives of this project was to validate the effectiveness of the proposed preform design methodology through forging trials conducted at an industrial forging facility. Cornell Forge supported this effort by allocating production time on their 1,300-ton forging press.

Since the preform design methodology developed in this work is intended for flashless or near-flashless die forging, Cornell Forge recommended investigating the possibility of reducing flash in one of their existing forged products. The selected component was a three-lobed drive hub that is currently produced at Cornell Forge using a multi-stage die forging progression. Details regarding the design of the preform for this hub are discussed in Section 4.2. To reduce flash formation, an additional preform stage was introduced into the forging progression, resulting in a three-stage forging sequence.

After finalizing the preform design, the preform dies and punches were designed and fabricated. **Figure 5.11** shows the forging progression for the three-lobed hub. As illustrated in the figure, a billet with a length of 3.9 in. and a diameter of 1.875 in. was first pancaked to a height of 3.0 in.

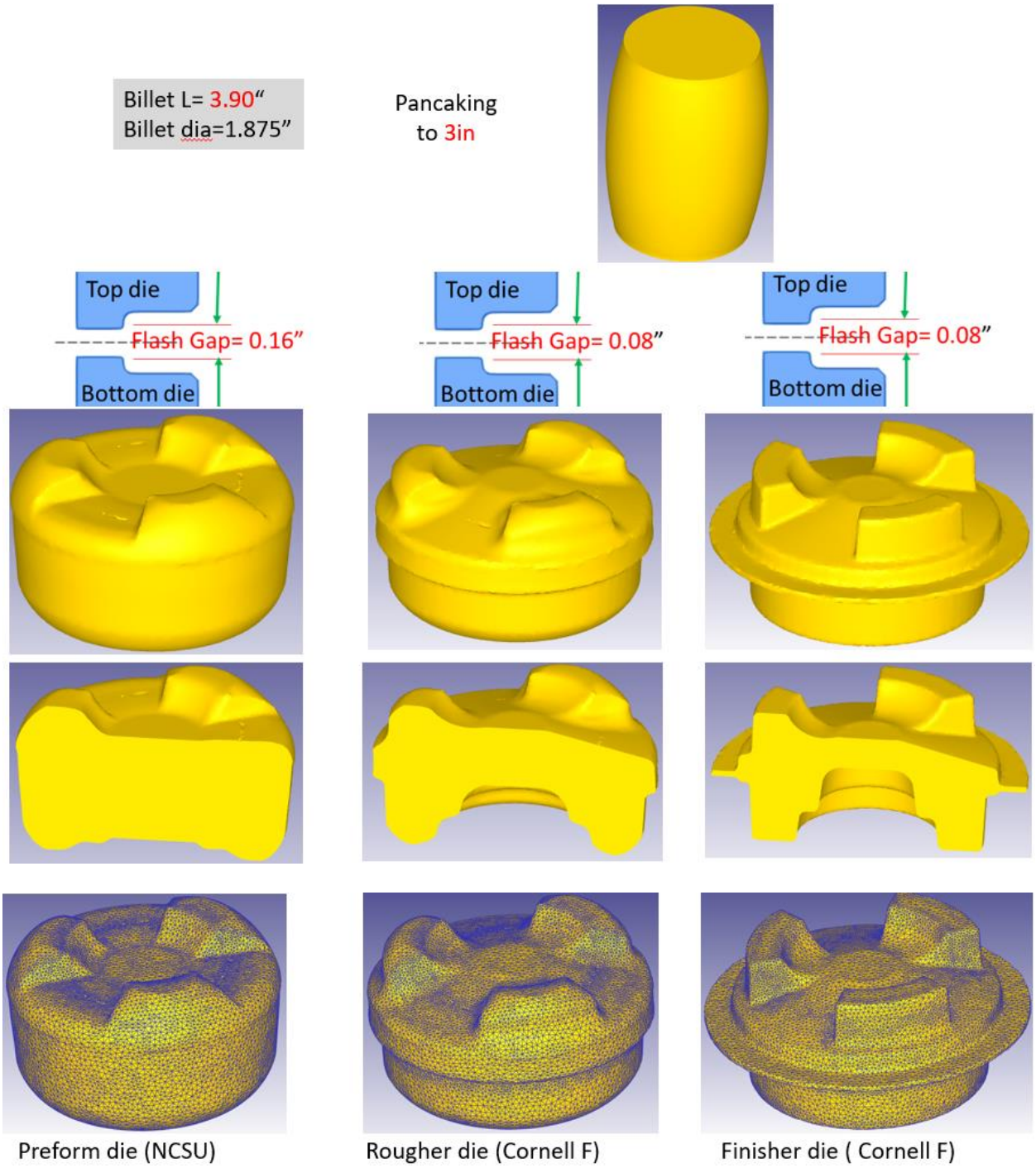


Figure 5.11 : Forging progression for three-lobe forging

Following the pancaking operation, the billet was transferred to the preform die station. At this stage, the part was intended to be formed without flash. After preforming, the part was transferred

to the rougher die and subsequently to the finisher die. The rougher and finisher dies were the same tooling currently used at Cornell Forge, and limited flash was permitted during these stages.

Figure 5.12 shows the CAD models of the top and bottom preform dies, while Figure 5.13 shows the fabricated and heat-treated preform dies and punches. These tooling components were manufactured at NC State University and shipped to Cornell Forge for installation in the 1,300-ton forging press.

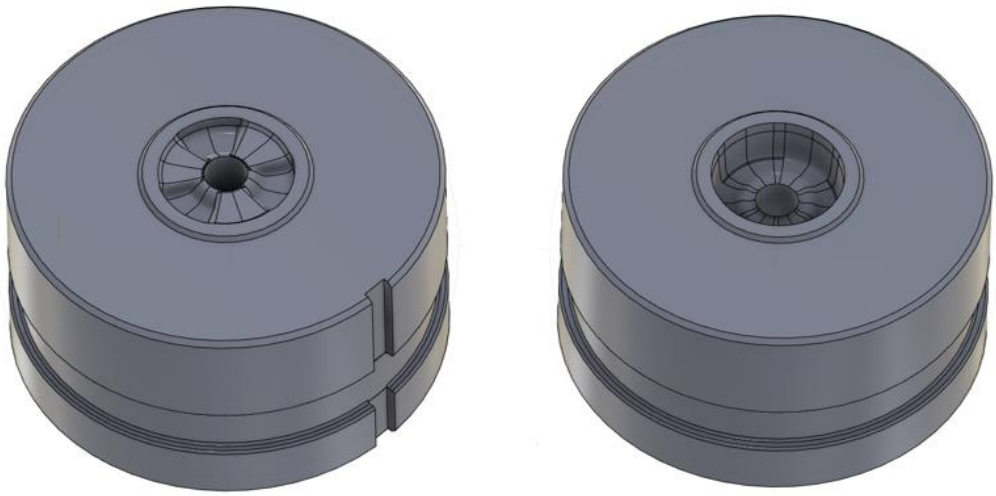


Figure 5.12: Top and bottom preform dies**, CAD models**

Figure 5.13: Top and bottom preform dies and punches fabricated at NC State University

After installing the dies and initiating the forging trials, Cornell Forge observed flash formation during the preform forging stage. As discussed earlier, the preform die was designed such that no flash should be generated during this stage. This issue was traced to inaccuracies in translating the forging press parameters used at Cornell Forge during the die design process. In summary, one of the preform dies was slightly longer than intended and therefore did not properly match the stroke characteristics of the mechanical press.

Given the large size of the dies, returning them to NC State University for modification would have been logistically challenging. As an alternative, Cornell Forge offered to perform minor machining modifications to the dies and conduct additional trials. Despite these adjustments, some flash formation persisted in the preform stage, as shown in Figure 5.14. Although the part was forged to the required overall dimensions, folding was observed in the flash region, indicating that the preform stage was not successfully executed.

Figure 5.14: Trial forged part at Cornell Forge

5.4 Lessons Learned and Industrial Considerations

The industrial forging trials conducted at Cornell Forge provided valuable insight into the practical challenges associated with implementing flashless or near-flashless forging methodologies at production scale. One key lesson learned is the critical importance of accurately capturing and translating press-specific parameters, such as stroke length, shut height, and compliance, during

die design. Even small discrepancies between assumed and actual press characteristics can lead to unintended flash formation and compromise the effectiveness of the preform stage.

These trials also highlighted the necessity of close coordination between design and manufacturing teams when transitioning from laboratory-scale simulations to industrial execution. While the proposed preform design methodology successfully produced the intended geometry in simulation, its industrial implementation requires careful calibration to account for press-specific constraints and tooling tolerances. Additionally, the willingness and ability of industrial partners to perform on-site tooling modifications proved essential in enabling rapid iteration and learning.

Overall, the trials demonstrated that the proposed methodology is promising for industrial application, but successful deployment requires early integration of press data, iterative validation, and flexibility in tooling adjustments. These considerations will inform future refinements of the methodology and support its more robust implementation in production forging environments.

Chapter 6

DISCUSSION, CONCLUSIONS, AND FUTURE DIRECTIONS

6.1 Discussion on Viability, Limitations, and Prospects of the Proposed Method

The primary objective of the proposed preform design methodology is to reduce or eliminate flash formation while producing high-quality forged parts free of defects. Achieving this objective not only minimizes material waste but also offers opportunities for improved energy efficiency and process optimization. By leveraging a material point backtracking approach, the methodology capitalizes on the predictive capabilities of finite element analysis (FEA), which captures complex interactions among die geometry, frictional conditions, and thermal gradients. In this framework, the origin of material points is traced backward through the deformation history obtained from a forward FE simulation, yielding a physics-informed estimate of the initial billet geometry. The key strength of the backtracking technique lies in its inherent incorporation of relevant physical phenomena, including localized strain accumulation, non-uniform temperature distribution, and anisotropic material flow behavior. As a result, the derived preform geometry is more realistic than those obtained using purely geometric inverse methods.

For two-dimensional forging problems, such as axisymmetric and plane strain configurations, the computational effort associated with the proposed methodology remains relatively modest, as FE iterations involve reconstruction along linear or curvilinear boundaries. In contrast, three-dimensional forging applications introduce significantly higher computational complexity. The iterative process requires material point tracking, surface mapping, geometry reconstruction, and solid modeling from point cloud data. Furthermore, for complex asymmetric 3D parts, material flow behavior becomes highly nonlinear, making proportional scaling of boundary surfaces insufficient for generating a reasonable initial preform. Instead, non-proportional geometric transformations are often required. This behavior was clearly observed in both the cross-joint and three-lobed drive hub case studies presented earlier. In these cases, reconstruction of a continuous and manufacturable preform during FE iterations necessitated extensive surface smoothing operations.

Figure 6.1 illustrates an example of surface irregularities encountered during preform reconstruction from backtracked material points in the cross-joint case. Such defects may arise from several factors, including insufficient point cloud density, mesh distortion during forging simulations, multi-valued material flow paths, and limitations inherent to CAD-based surface reconstruction tools.

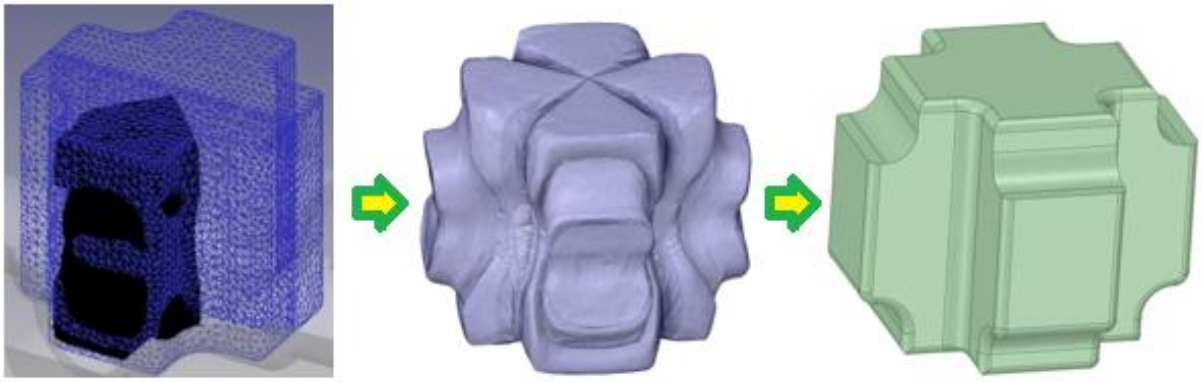


Figure 6.1. Defects and irregularities created when meshing the point cloud- Cross joint

Despite these challenges, many of the observed limitations can be mitigated through the adoption of a hybrid design workflow that combines backtracking-based preform generation with Design of Experiments (DOE) and surrogate modeling techniques. In this hybrid approach, the backtracked preform geometry serves as an initial, physics-consistent "seed" that captures the dominant material flow characteristics. This geometry defines a local design space in which key parameters—such as fillet radii, curvature distributions, and preform heights—can be systematically varied using DOE strategies. Surrogate models, including response surface models or kriging-based approaches, can then be employed to efficiently explore this design space with objectives such as minimizing flash formation, ensuring complete die filling, or reducing forging loads. By integrating high-fidelity FEA-based backtracking with parametric optimization, this strategy offers a practical pathway toward manufacturable and robust preform designs.

6.2 Conclusions and Future Directions

This study presented a physics-informed preform design methodology for forging processes based on geometrical resemblance, with particular emphasis on hot forging operations involving three-

dimensional deformation behavior. The methodology exploits the material point tracking capabilities of modern finite element software to enable backward reconstruction of preform geometries. Two representative case studies—a cross-joint and a three-lobed drive hub—were used to demonstrate the applicability and effectiveness of the approach.

Because three-dimensional forging simulations involve tracking large volumes of material point data during each FE iteration, a systematic data management framework was developed. This framework includes automated material point generation, backward tracking, surface reconstruction, and solid modeling from point cloud data.

The major conclusions of this study are summarized as follows:

- Preform design for complex 3D forgings requires careful handling of point cloud data. Unlike 2D forging problems, achieving an optimal preform for complex geometries necessitates refined meshes to capture sufficient material point information. However, because extremely fine meshes are often impractical, geometry smoothing techniques must be employed, which may result in the loss of minor geometric details in the reconstructed preform.
- The cross-joint case study demonstrated material savings of approximately 13.5% through the introduction of an intermediate preform stage developed using the proposed methodology.
- Significant reductions in material waste are achievable even for highly complex 3D forgings, as evidenced by the three-lobed drive hub case study, where iterative refinement of the preform geometry during FE simulation substantially reduced flash formation.
- While data handling and computational cost remain potential challenges, ongoing advances in computational speed and the integration of optimization tools within metal forming software environments have greatly improved the practical feasibility of the methodology for industrial applications.

Future work will focus on the following directions:

- Conducting sensitivity analyses to quantify the influence of different initial expanded geometries on convergence behavior and final preform accuracy.

- Further automating key steps in the workflow, including material point generation, surface smoothing, and solid geometry reconstruction from point cloud data.
- Developing a hybrid preform design framework that integrates FEA-based material point backtracking with DOE and surrogate modeling techniques, enabling accelerated convergence toward flashless, fully filled forgings while maintaining manufacturability.

By addressing these areas, the proposed methodology can evolve into a robust and practical tool for preform design across a broad range of industrial forging applications.

Acknowledgements

The authors would like to acknowledge the Forging Industry Education and Research Foundation (FIERF) for funding this work, Scientific Forming Technologies Corporation for their assistance with DEFORM 3D, and industrial partner Cornell Forge for technical support.

References

1. Vemuri, K.R.; Oh, S.I.; Altan, T. A Knowledge-Based System to Automate Blocker Design, *Int. J. Mach. Tools Manufact, (1989) Vol. 29, No. 4, pp.505-518*,
2. Kim, H.; Taylan, A. Computer-aided Part and Processing-Sequence Design in Cold Forging, *Journal of Materials Processing Technology, (1992) 33 (1992) 57-74*
3. Caporalli, A.; Gileno, L.A.; Button, S.T. Expert System for Hot Forging Design, *Journal of Materials Processing Technology, (1998) 80–81 131–135*
4. Osakada, K.; Yang, G.B.; Nakamura, T.; Mori, K. Expert System for Cold-Forging Process Based on FEM Simulation, *Annals of the ClRP (1990) 3/1/*
5. Bariani, P.; Knight, W.A. (1988),Computer-Aided Cold Forging Process Design: A Knowledge-Based System Approach to Forming, *Annals of the ClRP* (1988), *Vol. 37/1/*
6. Kim, D.Y.; Park, J-J. Development of an Expert System for the Process Design of Axisymmetric Hot Steel Forging, *Journal of Materials Processing Technology* (2000), *101 223-230*
7. Kim, H.-S.; and Im, Y.-T. Expert System for Multi-Stage Cold-Forging Process Design with a Re-Designing Algorithm, *Journal of Materials Processing Technology, 54 (1995) 271-285*

8. Umeda, M.; Mure, Y.; Katamine, K.; Kawahigashi, K. General Step Reduction Method for Knowledge-Based Process Planning of Non-Axisymmetrical Forged Products, *Procedia Engineering 207 448–453, DOI: 10.1016/j.proeng.2017.10.803*
9. Hedicke-Claus, Y.; Kriwall, M.; Stonis, M.; Behrens, B.A. Automated design of multi-stage forging sequences for die forging, *Production Engineering (2023) 17:689–701*
10. Park, J.J.; Rebelo, N.; Kobayashi, S. A new approach to preform design in metal forming with the finite element method, *International Journal of Machine Tool Design and Research, (1983), 23-1, 71-79*
11. Zhao, G.; Wright, E.; Grandhi, R.V. Computer aided design in forging using the inverse die contact tracking method, *Int. J. Mach, Tools Manufact (1996). Vol. 36. No. 7, pp. 755-769, 1996*
12. Santangelo, A.; Blanke, P.; Hadifi, T.; Wolter, F.E.; Behrens, B.-A. Fast 3D inverse simulation of hot forging processes via Medial Axis Transformation: an approach for preform estimation in hot die forging, *Prod. Eng. Res. Devel. (2013) 7:409–416.*
13. Park, D.; Park, J.; Kim, N. A 3D Preform Design Method Based on a Generative Artificial Intelligence Algorithm, *Journal of Manufacturing Processes 144 (2025) 190–208*
14. Torabi, S.H.R.; Alibabaei, S.; Bonab, B.B.; Sadeghi, M.H.; Faraji, Gh. Design and optimization of turbine blade preform forging using RSM and NSGA II, *J Intell Manuf (2017) 28:1409–1419*
15. Lee, S.; Quagliato, L.; Park, D.; Kwon, I.; Sun, J.; Kim, N. A New Approach to Preform Design in Metal Forging Processes Based on the Convolution Neural Network, *Appl. Sci. 2021, 11(17), 7948; https://doi.org/10.3390/app11177948*
16. Oh, M.; Oh, M.; Kim, J.; Cho, J.; Kim, M.; Joun, M.; Hong, S.; Reliability-Based Design Optimization of Bearing Hub Preform for Minimizing Defects Considering Manufacturing Tolerance in Hot Forging Process, *Appl. Sci. 2024, 14(23), 11316; https://doi.org/10.3390/app142311316*
17. Shao, Y.; Lu. B.; Xu, D.K.; Chen, J.; Ou, H.; Long, H.; Guo, P.Y. Topology-based preform design optimization for blade forging, *Int J Adv Manuf Technol (2016) 86:1593–1605*
18. Shao, Y.; Yan, L.; Guo, P.; Yang, H.; Shi, F.; Feng, D. A Comprehensive Study on Fitness Approximation Techniques in Shape Optimization of Aerofoil Forging Preform Tools, *Metals* 2019, *9*(6), 617; https://doi.org/10.3390/met9060617

19. Yang, C.; Ngaile, G. Preform design for forging and extrusion processes based on geometric resemblance. *(2009) Proc. IMechE Vol. 224 Part B: J. Engineering Manufacture, pp 1409-1423*