Pareto-based multi-objective hot forging optimization using a genetic algorithm

Catarina F. Castro\textsuperscript{1}, Carlos C. António\textsuperscript{2}, Luisa C. Sousa\textsuperscript{3}

\textsuperscript{1}IDMEC-FEUP, University of Porto, Porto, Portugal, ccastro@fe.up.pt
\textsuperscript{2}IDMEC-FEUP, University of Porto, Porto, Portugal, cantonio@fe.up.pt
\textsuperscript{3}IDMEC-FEUP, University of Porto, Porto, Portugal, lcsousa@fe.up.pt

Abstract
Efficient control techniques must be preceded by well-designed processes. A generally accepted definition of a well-designed process is one that is Pareto optimal, i.e., no design objective can be improved without degrading at least one other design objective. Indeed, optimal design enables effective trade-off of competing design objectives, including controllability and robustness goals.

The main goal of the present work is the design of a Pareto-based multi-objective optimization model for multi-stage hot forging processes. The optimization methodology considers a genetic algorithm supported by an elitist strategy. An iterative procedure is considered and Pareto optimal solutions are found managing the drawing of the Pareto front and enabling the extraction of optimal solutions according to selected preferences. The design example consists of a two-stage forging process applied to a pre-heated billet of AISI 1018 steel. Two sets of variables are considered: shape design variables and process variables. The objective functions are related to the minimization of the forging load, the control of the forged shape and the material microstructure.

Keywords: Forging simulation, Multi-objective optimization, Genetic algorithms.

1. Introduction
It is rare for any well-designed process to concern only a single value or objective. Generally multiple objectives or parameters have to be met or optimized before any solution is considered adequate. In most real-world optimization scenarios multiple and often conflicting objectives arise naturally. Evolutionary multi-objective optimization algorithms form a class of search strategies to solve multi-objective optimization problems by combining the fields of evolutionary computation and classical multiple criteria decision making [1-4].

Many complex industrial components, as well as many consumer goods, are produced through forging. Shapes of complex geometry are commonly produced from billets of very simple initial geometry. In the process, the metal is forced to acquire the shape of the surrounding die cavities requiring large forces and imposing high stresses on the tools. In hot forging, the metal is heated to temperatures above its recrystallization temperature so as to soften the metal and decrease the material yield stress reducing the maximum required loads.

In multi-stage forging one or more sets of preform dies are usually designed and produced to complete the schedule of manufacturing the final product. Production cost is highly dependent on development time, tool/die life, required energy, waste material and part rejection. Forging is a complex nonlinear process and simulation based on the finite element method has been an ongoing research field. Optimal forging design considering geometric, material and process properties of cold and hot operations have been presented in the literature [5-8]. Metal forming optimization is a typical multi-objective problem involving simultaneous optimization of several often mutually concurrent objectives. In multi-objective optimization problems, gradient based methods are often impossible to apply. However, multi-objective genetic algorithms (GAs) are able to find optimal trade-offs in order to get a set of solutions that are optimal in an overall sense [9-12].

Our previous studies on forging optimization consider the weight coefficients approach to solve multi-objective problems by using a single-objective functional which combines multiple objective functions into one [6,8]. Nevertheless, it is difficult to make sure whether the solution achieved is an optimal one since each set of coefficient combination will induce one optimal solution. In this paper a Pareto-based multi-objective optimization approach is considered.

2. Genetic Algorithms
The basic idea for genetic algorithms is to imitate the natural process of biological evolution [13]. For a given problem or design domain of significant complexity, there exists a multitude of possible solutions that form a solution space. In a GA, a highly effective search of the solution space is performed, allowing a population of strings representing possible solutions to evolve through basic genetic operators. The problem to be solved is described using a certain number of controllable parameters. In this project, design variables expressed by real numbers are converted to binary numbers. The design vector is then assembled in a string and each binary string is called an individual by analogy to natural evolution. Then a group of N different design vectors is created and each
set of solution candidates is called a population of individuals \( P \). The quantity \( N \) is called the population size. The quality of each individual in the population (each design vector) is expressed in terms of a scalar valued fitness function related with the objective function of the optimization problem. Individuals (i.e. design vectors) of lower or greater fitness are considered worse or better fitted, respectively. The algorithm advances by choosing individuals out of the population to become the parents of the next generation (natural selection, survival of the fittest). A new population of solutions \( P^{t+1} \) is generated from the previous \( P^t \) using the genetic operators: Selection, Crossover, Elimination/Substitution and Mutation. The outline of the considered genetic algorithm is presented in Figure 1.

![Figure 1: Outline of the GA](image)

The considered GA operates as follows [6,8]:

**Selection:** Individuals of a population are ranked according to their fitness. The elite group is defined as the group containing the highest ranked individuals and the elite solutions of the population are not lost, they are transfer into the next population. Selection of the progenitors is performed by randomly choosing pairs of progenitors with one individual from the best-fitted group (elite) and another from the least-fitted group.

**Crossover:** This operator has a mechanism of combining the genetic material from two chromosomes (progenitors) to create a new chromosome (offspring). Crossover can find solutions and avoid hill climbing problems because it is not a directed search. The new individuals created by crossover will join the elite group in order to start forming the next population.

**Elimination/Substitution:** Solutions with similar genetic properties are eliminated and then substituted with new randomly-generated individuals. The original size population is recovered including a group of new solutions obtained with the mutation operator.

**Mutation:** Mutation is an important part of the genetic search as it helps to prevent the population from stagnating at any local optima. Implicit mutation is considered quite different from classic techniques where a reduced number of genes are changed. In this way, the diversity of the population is guaranteed. After mutation, the new population is found and the evolutionary process pursues.

3. Multi-objective optimization

Multi-objective optimization seeks to optimize the components of a vector-valued objective function. Unlike single objective optimization, the solution to this problem is not a single point, but a family of points known as the Pareto-optimal set. By maintaining a population of solutions, GAs can search for many Pareto-optimal solutions in parallel. This characteristic makes GAs very attractive for solving multi-objective problems.

The optimization process can be mathematically formulated as

\[
\text{Minimize } F(b) = (f_1(b), f_2(b), \ldots, f_m(b))
\]

subject to
\[
b^i_{\text{lower}} \leq b_i \leq b^i_{\text{upper}}, \quad i = 1, \ldots, n
\]
\[
g_k(b) \leq 0, \quad k = 1, \ldots, p
\]

where \( b = (b_1, b_2, \ldots, b_n) \) is the design vector, \( b^i_{\text{lower}} \) and \( b^i_{\text{upper}} \) represent the lower and upper boundary of the \( i \)th design variable \( b_i \), \( f_j(b) \) is the \( j \)th objective function and \( g_k(b) \) the \( k \)th constraint function.

Introducing the Pareto optimal solution in a minimization problem, a feasible solution \( b^* \) is a Pareto optimal solution if and only if, there is no other feasible solution \( b \) such that

\[
f_j(b) \leq f_j(b^*) \quad , \quad j = 1, \ldots, m
\]
and for at least one \( k \) satisfying

\[ f_k(b) < f_j(b^*) \]  

(4)

Here, optimal solutions, i.e., solutions not dominated by any other solution, may be mapped to different objective vectors. In other words: there may exist several optimal objective vectors representing different trade-offs between the objectives. The set of optimal solutions in the decision space is in general denoted as the Pareto set and its image in objective space as Pareto front. With many multi-objective optimization problems, knowledge about this set helps the decision maker in choosing the best compromise solution.

Many problems in engineering domains are multi-objective problems being difficult or even impossible to minimize all objective functions simultaneously. As soon as there are many (possibly conflicting) objectives to be optimized simultaneously, there is no longer a single optimal solution but rather a whole set of possible solutions of equivalent quality. Often, the design space is reduced to the set of optimal trade-offs and then design space exploration is performed to learn more about the Pareto set.

4. Design example

The presented multi-objective design example is a two stage hot forging upsetting [6,8]. The set-up considers a first stage forging using a preform die and a final forging stage with a die shape that matches exactly the required final product. The goal of this design example is to search for a preform die shape, an initial work-piece temperature and the stroke length for each stage that will produce after forging a flashless cross-sectional H-shaped axisymmetric product with complete die fill. Axisymmetric turbine disks and ribs are examples of H-shaped industrial forging applications.

4.1 Two-stage hot forging upsetting

The initial billet is a cylinder of 25 mm diameter by 20 mm height of AISI 1018 steel. AISI 1018 is a low carbon steel, having higher manganese content than certain other low carbon steels, used in many industrial applications. The temperature dependent material constitutive relation is given by

\[
\sigma = \begin{cases} 
173.73 \varepsilon^{0.070} \text{ [MPa]} & T < 1143K \\
108.93 \varepsilon^{0.152} \text{ [MPa]} & 1143K < T < 1363K \\
75.83 \varepsilon^{0.192} \text{ [MPa]} & T > 1363K 
\end{cases}
\]  

(5)

Table 1 presents the material properties of the work-piece for the thermal model necessary for the calculation of heat transfer, where \( \lambda_T \) is the conductivity, \( \rho \) is the density, \( c_T \) is the specific heat supply, \( h_{iuh} \) is the lubricant heat transfer coefficient, \( h_s \) is the surface heat transfer coefficient and \( rad \) represents the radiation heat flow. The fraction of plastic work transformed into heat is \( k_T = 90\% \). The constant shear friction factor \( m \) is taken as 0.6. The die matrix is assumed to be rigid with no internal heat generation with an initial die temperature \( T_{die} = 285K \).

Table 1: Material properties for the low-carbon steel AISI 1018

<table>
<thead>
<tr>
<th>Property</th>
<th>Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \lambda_T ) [N/m s K]</td>
<td>0.0469 ( T + 77.467 )</td>
</tr>
<tr>
<td>( \rho c_T ) [N/mm(^3)K]</td>
<td>4.10 ( \times 10^8 \times T^3 - 6.10^2 \times T^2 + 0.0344 \times T - 2.3977 )</td>
</tr>
<tr>
<td>( h_{iuh} ) [N/s mm K]</td>
<td>4.0</td>
</tr>
<tr>
<td>( h_s ) [N/s mm K]</td>
<td>0.00295</td>
</tr>
<tr>
<td>( rad ) [N/m s K(^3)]</td>
<td>567.10(^{-13})(1.10(^{-7})\times T(^2) - 9.10(^{-5})\times T + 0.044)</td>
</tr>
</tbody>
</table>

4.2 Objective functions

In most forging sequences, the optimization process searches the minimization of at least three components: the die under-fill, the product unwanted flash and the total cost of the process. Accounting for die under-fill and product unwanted flash, a measure of the distance between the forged shape at the end of the process and the prescribed shape can be calculated as

\[ f_1(b) = \int_{\Gamma_{end}} \| \pi(X) - X(b) \|^2 dS \]  

(6)

where \( \pi(X) \) is the projection of a material point \( X \) of the work-piece boundary \( \Gamma_{end} \) onto the surface of the prescribed shape at the end of the forming process.

The total energy is a measure of the actual cost of the process given by
where \( \mathbf{t} \) is the traction vector and \( \mathbf{v} \) the die velocity.

Microstructure development has a close relationship with the temperature of the initial billet. Temperature affects the forming processes by influencing the distribution of strain and strain rate and the metal micro-structural behavior. So the constraint function

\[
g(\mathbf{b}) = \frac{T_{\text{end}}(\mathbf{b})}{T_a} - 1 \leq 0
\]

is introduced, being \( T_{\text{end}}(\mathbf{b}) \) the maximum temperature registered in the work-piece along the forging process and \( T_a \) the maximum allowed temperature.

### 4.3 Design Variables

For this example, a total of nine design variables define the design vector \( \mathbf{b} \). The first 8 variables represent the controllable variables of the first stage: six control points used to simulate the B-spline curve of the preform die shape, the seventh component of the design vector \( \mathbf{b} \) is the initial temperature of the work-piece and finally the eighth component is associated to the stroke length of this first stage. Only one controllable variable will be considered for the second stage: the stroke length of the final stage as the ninth component.

The prescribed H-shaped final product presents a 30 mm diameter and a height variation going from a lowest 12mm at the z-axis to a high 16 mm at the top. The 4 mm amplitude variation for the H-shape and considering the usual temperatures for steel hot forging, the lower and upper boundary of the design variables are

\[
-3 \leq b_i \leq 3 \text{ mm} \quad , \quad i = 1, ..., 6
\]

\[
1000 \text{ K} \leq T_0 = b_7 \leq 1400 \text{ K}
\]

\[
1 \text{ mm} \leq L_i = b_8 \leq 5 \text{ mm}
\]

\[
1 \text{ mm} \leq L_{\text{final}} = b_9 \leq 5 \text{ mm}
\]

The design variables expressed by real numbers are converted to binary numbers, and each binary string is looked as an individual. Figure 2 illustrates the string structure adopted for the optimization problem. A genetic string is made up of sub-strings representing the number of stages, where the number of stages is assumed fixed. Each of the substrings consists of product and process variables.

As a compromise between computer time and population diversity, parameters for the genetic algorithm were taken as \( N_{\text{pop}} = 12 \) and \( N_e = 6 \) for the population and elite group size, respectively. The number of bits in binary codifying for the design variables has been \( N_{\text{bit}} = 5 \), making a chromosome length of 45 bits.

### 4.4 Single-objective optimization approach

In this work, a two stage forging upsetting is considered.

![Figure 3: Geometry and finite element mesh for the forging problem](image-url)
The set-up considers a first stage forging using a preform die and a final forging stage with a die shape that matches exactly the required final product. The simulation considers only one quarter of the process, taking advantage of symmetric conditions.

Figure 3 presents the geometry and finite element mesh considered for the forging simulation. The two-dimensional computer program [6,8] models the geometry of the work-piece and dies by a combination of four node and linear friction elements. The initial work-piece is heated and the dies are considered at room temperature. The preform die shape will be optimized considering different geometries as given by B-spline curves.

Considering the weight coefficients approach and using the objective functions of the optimization problem given by Eq. (6) and Eq. (7), a single-objective metal forming optimization can be formulated [6,8]

\[
\text{Minimize } f(\mathbf{b}) = 10^2 f_1(\mathbf{b}) + 10^{-4} f_2(\mathbf{b}) 
\]

subject to the same conditions as before. The constraint defined in Eq. (8) that is the maximum allowable temperature during forging is taken as \( T_a = 1450 \) K.

Considering the fitness value for each individual, \( \phi(\mathbf{b}) = K - f(\mathbf{b}) \), being \( K \) a constant to ensure positiveness, one run of the developed GA has converged to the optimal shape solution presented in Figure 4. The termination of the evolutionary process was considered when the mean fitness of the six best individuals (elite group) did not change during five consecutive generations. An almost defect free forging product was produced using an optimized preform die geometry.

Figure 4: Two-stage single-objective optimized forging solution

The objective functional given in Eq. (10) represents in fact a near-net shape optimization problem. The term \( f_1(\mathbf{b}) \) associated with the first objective function measuring the distance between the forged shape at the end of the process and the prescribed shape is enhanced comparatively to the second term associated to the required energy. In order to try to enquire if the considered GA operators were able to ensure diversity among the solution searched space, values of distances and energies calculated along the GA process are presented in Figure 5.

Figure 5: Objective space along the search for the single-objective optimized forging solution
The set of function values is quite spread over the objective space giving an interesting foundation for a multi-objective development.

4.5 Multi-objective algorithm design
Genetic algorithms are known to be robust and have been enjoying increasing popularity in the field of numerical optimization in recent years. As solvers the following two features are expected: the solutions obtained are Pareto-optimal and they are uniformly sampled from the Pareto-optimal set. A GA must guide the search towards the Pareto set and at the same time must prevent non-dominated solutions from being lost in order to maintain diversity avoiding that the population contains mostly identical solutions.

The fitness assignment is considered for each individual taking into account both dominating and dominated solutions. Each individual in the population has a strength value representing the number of solutions it dominates. The fitness value $\phi(b)$ of each individual is determined by the strengths of its dominators, $\phi(b) = 0$ corresponds to a non-dominated individual, while a high $\phi(b)$ value means that the individual is dominated by many.

The developed GA operator selection defines an elite group as the group containing the highest ranked individuals. This elitism ensures that the best solutions of the population are not lost. On the other hand, two other operators, namely, elimination /substitution and implicit mutation are built avoiding that the population contains mostly identical solutions. Elimination/substitution operator plays an important role because the solutions with similar genetic characteristics are controlled. Only one chromosome among similar individuals is kept in the population. Then other chromosomes generated in a random way replace the eliminated solutions. With the implicit mutation operator a group of chromosomes randomly generated is introduced into the population avoiding the rising of local minima.

Starting from 60 independently generated populations, applying the Pareto-based multi-objective optimization developed model and using the optimization functions defined in Eq. (6) and Eq. (7) and constraint given by Eq. (8), the obtained results are shown in Figure 6. The presence of a Pareto front confirms the conflicting relationship between die under-fill, product unwanted flash and required energy for the process.

![Figure 6: Pareto front for the two-stage multi-objective forging problem](image)

Since the search space is not known in absolute terms, results shown in Figure 6 were obtained for the objective space. Comparison of the random search space and the achieved results confirms that the solution algorithm has been able to converge to the Pareto front locating a reasonable spread of multiple optimal solutions. Based on the visualization of the objective space and the convergence to the near Pareto front with a good spread of solutions the method is validated.

5. Conclusions
Optimization problems involving multiple objectives are common. In this context, since genetic algorithms are flexible with respect to the problem formulation they represent a valuable tool. Due the fact that they operate on a set of solution candidates, genetic algorithms are well-suited to generate Pareto set approximations. This is reflected by the rapidly increasing interest in the field of multi-objective optimization. Finally, genetic algorithms are able to deal with highly complex problems and therefore they can be seen as an approach to more traditional methods.

This paper describes the optimization algorithm applied to a multi-stage hot forging process using an approach
based on genetic algorithms. The presented example is a multi-objective problem involving design variables and their dependency relationship between stages. The near-optimal solutions obtained with the proposed algorithm offer the capability to designers to trade-off solutions at various dimensions such as parameter level, objective level and inter-stage level.

There are a number of proposed future research activities. Genetic algorithms are very attractive for solving multi-objective problems and by maintaining a population of solutions, a more elaborated GA can search for many Pareto-optimal solutions in parallel. As for optimization problems within a multistage forging framework it would be useful to investigate the nature of interactions between the different stages during the evolutionary search. Since the problem parameters are idealized, it would also be interesting to test the performance of the algorithm with real forging processes.

Acknowledgements

The authors acknowledge FCT-Fundação para a Ciência e Tecnologia, Portugal, for the financial support through research unit UMNME (10/225).

References


