A comparison between optimisation algorithms for metal forming processes

M.H.A. Bonte[1], T.T. Do[2], L. Fourment[2], A.H. van den Boogaard[1], J. Huétink[1], A. Habbal[3]

1 Faculty of Engineering Technology, University of Twente, P.O. Box 217, 7500 AE Enschede, Netherlands
URL: www.tm.ctw.utwente.nl e-mail: m.h.a.bonte@utwente.nl

2 CEMEF, Ecole des Mines de Paris, BP 207, 06 904 Sophia Antipolis Cedex, France
URL: www-cemef.cma.fr e-mail: tien-tho.do@ensmp.fr; lionel.fourment@ensmp.fr

3 Lab. Mathématiques, Parc Valrose, 06 108 Nice Cedex, France
URL: www-math.unice.fr e-mail: habbal@unice.fr

ABSTRACT: Coupling optimisation algorithms to Finite Element (FEM) simulations is a very promising way to achieve optimal metal forming processes. However, many optimisation algorithms exist and it is not clear which of these algorithms to use. This paper compares an efficient Metamodel Assisted Evolutionary Strategy (MAES), three variants of a Sequential Approximate Optimisation (SAO) algorithm, and two iterative algorithms (BFGS and SCPIP). They are compared to each other and to reference situations by application to two forging examples. It is concluded that both MAES and SAO outperform the iterative algorithms. Moreover, they yield significant improvements with respect to the reference situations, which makes them both very interesting algorithms for optimising metal forming processes.

Key words: Optimisation algorithms, Finite Element Method, metal forming, forging

1 INTRODUCTION

During the last decades, Finite Element (FEM) simulations of metal forming processes have become important tools for designing feasible production processes. More recently, several authors recognised the potential of coupling FEM simulations to mathematical optimisation algorithms to design *optimal* metal forming processes instead of only *feasible* ones.

A way of optimising metal forming processes is using classical iterative optimisation algorithms (Conjugate gradient, BFGS, etc.), where each function evaluation means running a FEM calculation. These algorithms are well-known, but suffer from a number of disadvantages: they do not allow for parallel computing, require difficult to obtain sensitivities, and may be trapped in a local optimum.

Several authors have tried to overcome these disadvantages by applying genetic or evolutionary optimisation algorithms. Genetic and evolutionary algorithms look promising because of their tendency to find the global optimum and the possibility for parallel computing. However, they are known to require many function evaluations [1].

A third alternative is using approximate optimisation algorithms such as Response Surface Methodology (RSM) or Kriging (DACE). RSM is based on fitting a lower order polynomial metamodel through response points, Kriging interpolates exactly through these response points. Approximate optimisation algorithms allow for parallel computing, find global optima and do not need sensitivities.

Which of the above algorithms to use for the optimisation of metal forming processes using time-consuming FEM simulations is still not clear. This paper compares an efficient evolutionary strategy, three variants of a Sequential Approximate Optimisation algorithm, and two iterative optimisation algorithms. The algorithms are introduced shortly in Section 2. They are compared to each other by applying them to two forging examples in Section 3. Section 4 discusses the results of the comparison.

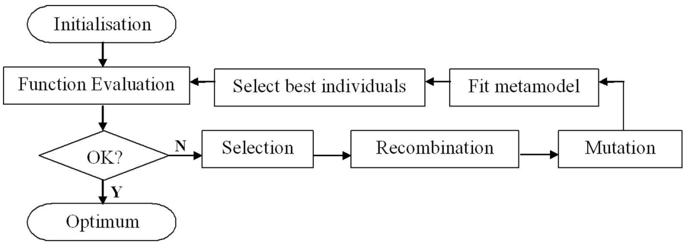


Figure 1: Metamodel Assisted Evolutionary Strategy

2 THE OPTIMISATION ALGORITHMS

2.1 Metamodel Assisted Evolutionary Strategy (MAES)

The first algorithm is a Metamodel Assisted Evolutionary Strategy (MAES) [1, 2], which is depicted in Figure 1. Like any evolutionary algorithm, it comprises selection, recombination and mutation. As mentioned in the introduction, the large disadvantage of evolutionary algorithms is the necessity for performing many function evaluations, i.e. many time-consuming FEM simulations. This problem is overcome by predicting the objective function values required by the evolutionary strategy first by fitting a Kriging metamodel. Subsequently, only the best 20% of the offspring individuals are evaluated by running a FEM simulation. This significantly reduces the total number of simulations that need to be performed for optimisation.

2.2 Sequential Approximate Optimisation (SAO)

The approximate optimisation algorithm used for the comparison is shown in Figure 2 [3, 4]. It comprises a spacefilling Latin Hypercubes Design Of Experiments (DOE) strategy, RSM and Kriging metamodelling and validation techniques, and a multistart SQP algorithm for optimising the metamodels. The algorithm allows for sequential improvement of the accuracy and can thus be denoted as a Sequential Approximate Optimisation (SAO) algorithm.

Three variants of this SAO algorithm are taken into account. They differ in the sequential improvement strategy. The first variant (SAO) simply adds new DOE points in a spacefilling way. The second and third variants employ all the information obtained during a previous iteration, i.e. the shape of the metamodel $\hat{y}$ (RSM or Kriging) and its standard deviation s. For Kriging, $\hat{y}$ and s are shown in Figure 3.

The second sequential improvement strategy (SAO-MMF) selects the new DOE points that Minimise the Merit Function:

$$f_{\text{merit}} = \hat{y} - w \cdot s \tag{1}$$

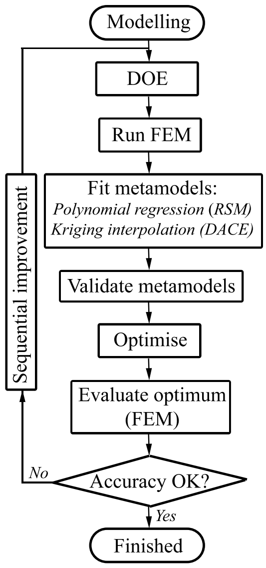


Figure 2: Sequential Approximate Optimisation [3, 4]

where w is a weight factor. If one selects $w = 0$, the new DOE points equal the optima of the metamodel $\hat{y}$. If $w \rightarrow \infty$, the new DOE points are simply added in a spacefilling way. We found that $w = 1$ provides a good compromise between both extreme cases.

The third sequential improvement strategy (SAO-MEI) selects the new DOE points that Maximise the Expected Improvement $E(I)$ [5]:

$$E(I) = (f_{\min} - \hat{y})\,\Phi\left(\frac{f_{\min} - \hat{y}}{s}\right) + s\phi\left(\frac{f_{\min} - \hat{y}}{s}\right) \tag{2}$$

in which again $\hat{y}$ and s are used. $f_{\min}$ is the lowest objective function value obtained in all previous iterations and ϕ and Φ denote the probability density and the cumulative distribution functions of the standard normal distribution, respectively. Both SAO-MMF and SAO-MEI tend to select new DOE points in the region where the global optimum is predicted to be ($\hat{y}$ is small). Additional points are selected where no points have been sampled before (s is large).

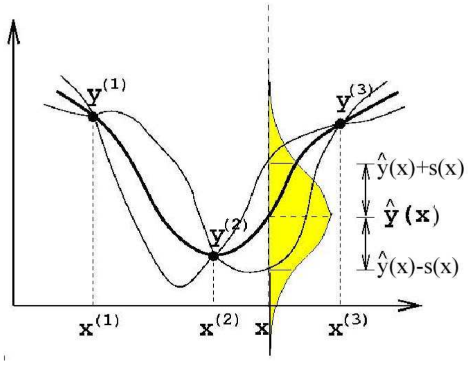
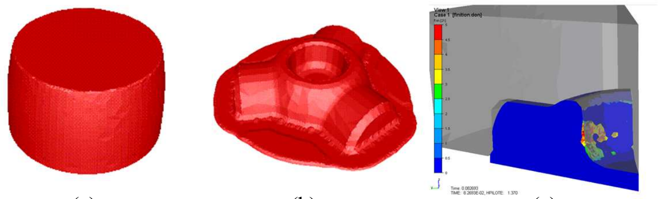


Figure 3: Sequential improvement employing metamodel information of the previous iteration



(a) (b) (c)

Figure 4: (a) The preform; (b) The spindle; (c) Folding

3 APPLICATION TO FORGING

The optimisation algorithms introduced in the previous section are compared to reference situations and each other by applying them to two forging examples. Two iterative algorithms are also included in the comparison: a widely available BFGS algorithm and a combination of Sequential Convex Programming and an Interior Point method (SCPIP) [6].

3.1 Spindle

The first forging application is a spindle. The spindle is produced in two steps: upsetting first results in a preform (Figure 4(a)), which is subsequently forged to the final spindle presented in Figure 4(b). In the reference situation, it suffers from a folding defect as can be seen from the Finite Element Model (FEM) of the spindle in Figure 4(c). FORGE3® was used as FE code.

An optimisation problem was modelled to solve this folding problem. An objective function Φ_{fold} was formulated to minimise the folding potential, see [2]. For the reference situation $\Phi_{\text{fold}} = 10.49$. The three design variables are control points of a B-spline that is used to describe the geometry of the preform and are presented as P_1, P_2 and P_3 in Figure 5.

The optimisation algorithms introduced in Section 2 and the iterative BFGS and SCPIP algorithms were applied to solve the optimisation problem. The results are presented in Figure 6 and will be discussed in Section 4 together with the results of the second forging example.

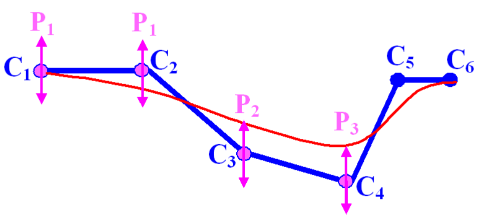


Figure 5: The B-spline describing the geometry of the preform of the spindle

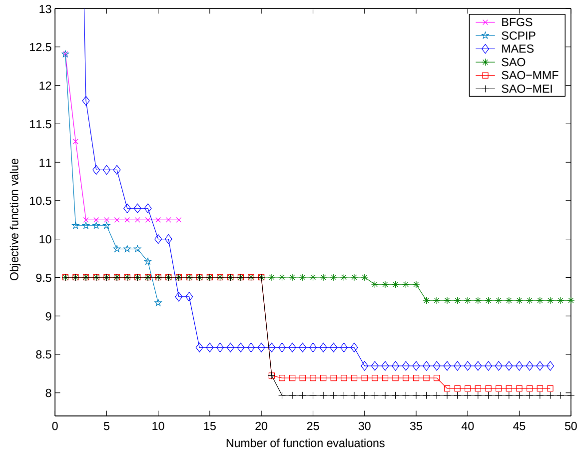


Figure 6: Convergence of the algorithms for optimising the spindle

3.2 Gear

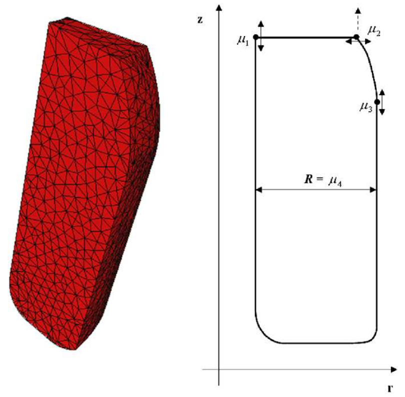
The second forging example is a gear. Its preform and the final product after forging are presented in Figure 7. It will be tried to improve the gear forging process with respect to the reference situation, which is the forging process proposed by the forging company.

Objective functions have been formulated to improve two things: the susceptibility to folding Φ_{fold} and the energy consumption needed for forming the gear Φ_{ene} [2]. They are combined to yield one total objective function Φ_{tot}. Figure 8(a) presents the FE model of the gear. Again FORGE3® is used as FE code. The design variables μ_1, μ_2 and μ_3 describe the geometry of the preform, which is visualised in Figure 8(b). Parameter μ_4 follows from the design variables when demanding volume conservation.

The results of applying the optimisation algorithms are summarised in Table 1. The table presents the optimal objective function values for the different algorithms and compares the improvement in folding potential and energy consumption with respect to the reference situation. Figure 9 presents the optimal preform geometries obtained by the different algorithms. The convergence behaviour of the optimisation algorithms is shown in Figure 10.

(a) (b)

Figure 7: A gear: (a) The preform; (b) The final product



(a) (b)

Figure 8: (a) The 3D FE model of the gear; (b) The design variables describing the geometry of the preform

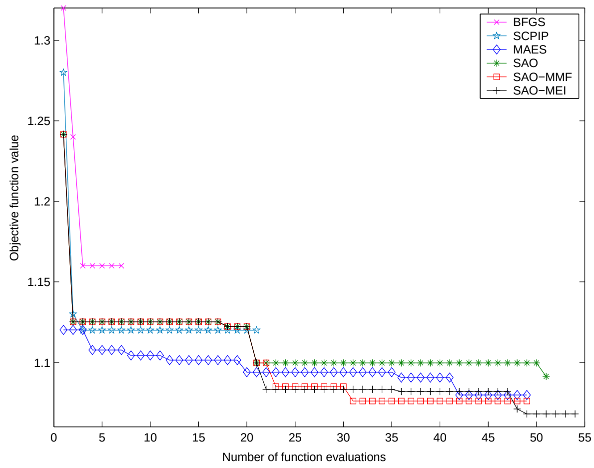


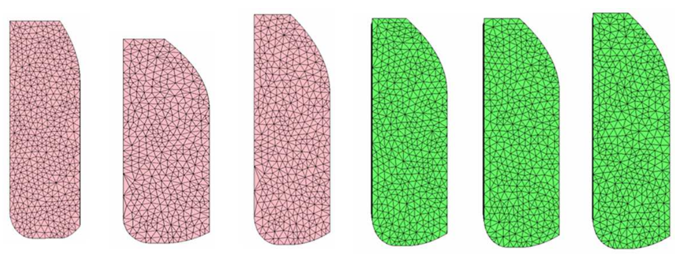
Figure 10: Convergence of the algorithms for optimising the gear

4 DISCUSSION

Concluding this comparison between optimisation algorithms, let us summarise the findings. For both forging cases, the results were very similar. All algorithms yielded better results than the reference situation. Moreover, Figures 6 and 10, and Table 1 show that the local iterative BFGS and SCPIP algorithms are outperformed by the other, global algorithms. For the spindle, the iterative algorithms proved not to be able to solve the folding defect and are insufficient.

Regarding Sequential Approximate Optimisation, SAO-MMF and SAO-MEI are more effective than SAO. For the spindle, SAO was not capable of fully removing the folding defect, whereas SAO-MMF and SAO-MEI solved the folding problem convincingly.

The Metamodel Assisted Evolutionary Strategy (MAES) performed approximately equally well as SAO-MMF and SAO-MEI. For both the spindle and the gear, SAO-MEI yielded the best results, followed closely by SAO-MMF and MAES, respectively. However, the difference is very small. This is demonstrated in Figure 9, where one can barely see a difference between the optimal preform geometries obtained by these three algorithms.



(a) (b) (c) (d) (e) (f)

Figure 9: Preforms of the gear: (a) Reference; (b) BFGS; (c) MAES; (d) SAO; (e) SAO-MMF; (f) SAO-MEI

In the end, it can be concluded that MAES, SAO-MMF and SAO-MEI all are very promising optimisation algorithms for the optimisation of metal forming processes. They have been shown to eliminate the folding defect for the spindle and have reduced both the folding susceptibility of the gear and the energy consumption needed for forging this part by approximately 10% as can be obtained from Table 1.

	Ref.	BFGS	SCPIP	MAES	SAO	MMF	MEI
Φ_{tot}	1.19	1.157	1.123	1.079	1.091	1.076	1.068
Φ_{ene}	–	–	–	-9.7%	-8.7%	-8.2%	-9.4%
Φ_{fold}	–	–	–	-7.6%	-6.6%	-9.8%	-9.8%

Table 1: Results of optimising the gear

REFERENCES

[1] M. Emmerich, A. Giotis, M. Özdemir, T. Bäck, and K. Giannakoglou. Metamodel-assisted evolution strategies. In *Proceedings of the International Conference on Parallel Problem Solving from Nature*, 2002.

[2] L. Fourment, T.T. Do, A. Habbal, and M. Bouzaiane. Gradient, non-gradient and hybrid algorithms for optimizing 2D and 3D forging sequences. In *Proceedings of ESAFORM*, Cluj-Napoca, Romania, 2005.

[3] M.H.A. Bonte, A.H. van den Boogaard, and J. Huétink. Metamodelling techniques for the optimisation of metal forming processes. In *Proceedings of ESAFORM*, pages 155–158, Cluj-Napoca, Romania, 2005.

[4] M.H.A. Bonte, A.H. van den Boogaard, and J. Huétink. Solving optimisation problems in metal forming using finite element simulation and metamodelling techniques. In *Proceedings of APOMAT*, pages 242–251, Morschach, Switzerland, 2005.

[5] M. Schonlau. *Computer Experiments and Global Optimization*. PhD thesis, University of Waterloo, Waterloo ON, Canada, 1997.

[6] C. Zillober. A combined convex approximation - interior point approach for large scale nonlinear programming. *Optimization and Engineering*, 2:51–73, 2001.