AUTOMATIC OPTIMIZATION APPLIED TO ON DIFFERENT MATERIAL FORMING PROCESSES

R.Ducloux[1*], S.Marie[1]

Transvalor S. A., Parc de Haute Technologie, Sophia-Antipolis, 694, av. du Dr. Maurice Donat,
06255 Mougins Cedex, France

ABSTRACT: In recent years, material forming simulation has become a day to day activity in many manufacturing plants. Time has come to make another step and to couple simulation algorithms with automatic optimisation. This is the only way to get all the numerical simulation benefits. Three very different examples are presented here and four others can be found in the quoted references. They include billet volume minimisation under constrain of die filling, press loading and die wear minimisation under constrain of die filling, inverse analysis and process parameters optimisation to improve part quality or geometry. All of them have been computed with the standard version of Forge 2009. Most of the presented work is the result of the ANR French agency funded LOGIC project. This project gathers 10 Forge users, a research centre (Cemef) and a software editor. It aims to bring to the market a fully integrated simulation/automatic optimisation tool dedicated to forging process.

KURZFASSUNG:

In den letzten Jahren ist die Simulation von Umformprozessen zu einer täglichen Aktivität in vielen Produktionsbetrieben geworden.

Heute können wir einen Schritt weitergehen und die Simulation mit einer automatischen Optimierung kuppeln. Dies ist der einzige Weg, um alle Vorteile der numerischen Simulation auszunutzen.

Drei sehr unterschiedliche Beispiele werden hier vorgestellt, vier weitere werden in den Referenzen zitiert. Dazu gehören die Minimierung des Rohlingvolumens mit Gewährleistung einer Vollgravur, Minimierung der Presskraft und des Verschleißes mit Gewährleistung einer Vollgravur, Rückwärtsanalyse und die Optimierung der Prozessparameter, um die Qualität oder Geometrie des Werkstückes zu verbessern. Alle Beispiele wurden mit der Standard-Version von Forge 2009 berechnet. Die meisten der gezeigten Arbeiten sind Ergebnisse aus dem Projekt LOGIC, von der französischen “Agentur für Finanzierung von Forschungsprojekten”, ANR, finanziert. Dieses Projekt gruppiert 10 Forge Anwender, ein Forschungs-Zentrum (CEMEF) und einen Software-Editor. Es ist darauf ausgerichtet, ein Simulationswerkzeug mit voll integrierter automatischer Optimierung für Schmiedeprozesse auf den Markt zu bringen.

KEYWORDS: Integrated Automatic optimisation, Meta Model, Inverse analysis, Die filling and folding prediction
SCHLÜSSELWÖRTER: Integrierte automatische Optimierung, Meta Modell, Rückwärtsanalyse, Vollgravur und Faltenvorhersage, Eigenschaften des Werkstückes.

1 INTRODUCTION

The recent history of material forming science shows that most of the technologies are first introduced by specialists and then start to spread when stability and ease of use reach a level compatible with a more widespread usage.

In the eighties, FEM was only accessible to specialists able to draw the right mesh and now many designers use simulation every day without even displaying the mesh. Dedicated packages for material processing simulation have been the major reason of this evolution. By narrowing the range of applications to a specific process, it is possible to set automatically all the numerical parameters and remove this burden from the user’s shoulders.

Until recently High Performance Computing was only used in R&D centres of large groups and today, just gathering 4 bi-quad cores PC, you can build a 10k€ 32-core powerful cluster which, thanks to Forge’s parallel computing capacities, can be used to compute millions elements parts. By tailoring typical parallelisation techniques to the specific algorithms used in Forge, it was possible to achieve very robust and efficient parallel software.

---

[*] Corresponding author: postal address, telephone number, fax, email

Automatic optimisation is probably about to follow the same track. Several packages are available and give good results providing you are able to select right algorithm and DOE. The next step to go further is to lower the needed expertise and increase the ease of us in order to make this technology available for every forging designer. To achieve this, we have used once again the same strategy. We have decided to include this feature into Forge's environment and to fine tune it to forged part design needs.

To achieve this goal, with the support of French ANR agency, we have created the LOGIC consortium gathering:

- Nine forging companies, working in different fields (automotives, aerospace, energy),
- The French technical centre in charge of mechanical industry (CETIM),
- A research laboratory (CEMEF/Ecole des Mines)
- TRANSVALOR, the Forge software editor.

Together we have selected more than 30 cases typical of the different components the partners use daily. We then successfully applied automatic optimisation techniques to all of them. As most of the cases are real cases they can't be published but we have obtained the right to show some characteristic examples and they all have been used to define the specification of the Forge2009 optimisation module.

2 CONCEPTS

2.1 DEFINITIONS

Before going further it can be useful to clarify the language we are going to use. To make it easier let's take an example. Imagine you have an issue with press tilting and you want to find part positioning which would minimise the phenomena.

The first thing to do is to identify the simulation result which describes your real-world issue. We will call it '*minimisable*'. Depending on the case, the most relevant choicie is not always the same. If you make fully coupled part/die/press computations you could consider relative displacement of the die versus theoretical position but if you only compute the part you may consider distance between the centre of the die and resulting force application point.

The second stage is to select the process parameters you want to play with to achieve your goal. It can be many different things but here it could be the initial positioning of the billet which can be seen as a displacement relative to a fixed position. We simply call them '*parameters*'.

If you solve the problem with no additional information, it is likely that the system will suggest you to put the billet completely out of the die. To avoid this you have of course to limit the range of *parameters* variation but it is not always enough and you may have to add '*constraints*'. Probably, in this case, the most natural *constraint* is to enforce die filling. Nothing prevents us from having several *minimisables* and *constraints.*

2.2 ALGORITHM SELECTION

Having said this, one can imagine that, if the *minimisable* is plotted as a function of the *parameters* values, there is no guaranty to have a unique minimum and, under some constraints, it is even less clear. An important point is then to select an algorithm which is robust regarding local extrema issues.

Another point is the difficulty to compute derivatives. Due to the complexity of the process, analytical derivatives are out of reach and numerical derivatives are polluted by remeshing artefacts (if you change any parameter, the flow will change, the remeshing will be then slightly different and it is difficult to sort among result variations what is created by the *parameter* change and what is related to remeshing variation).

For these 2 reasons we have selected a genetic type algorithm. The limitation of such algorithms is that they are supposed to need several hundreds of evaluations to reach the optimum. As each evaluation is a computation, this could be a real stopper. Fortunately Meta-model Assisted Evolution Strategy (MAES) makes it possible to considerably reduce the number of effective computations. This algorithm, combined with the fact that it makes no sense to find a very precise optimum (we are just optimising a model and selected parameters will be applied within a tolerance in the production plant), makes it possible to get good results within a few tens of computations. As several of these computations can be performed at the same time the optimisation elapsed time is reduced again.

2.3 MAES ALGORITHM BASIS

A more detailed presentation of the algorithm can be found in [4]. The idea is just to introduce here the concepts to make the following more clear.

The main idea of genetic type algorithms is:

- to start from a population: a predefine number of individuals which will be called first generation,
- to evaluate them (here compute them and evaluate *minimisables* and *constraint*),
- to select the best, breed them and create a new generation.

With MAES type algorithms, you use an evaluation of the *minimisables* and *constraints* (the Meta Model) to select the individual you really want to compute. There are different techniques to build it and evaluation time using Meta Model varies with the selected technique but in all cases it is much shorter than the FEM analysis and you can increase the size of the population keeping the time for the optimisation more or less constant.

3 PREPROCESS INTEGRATION

3.1 PRE-REQUISIT TO OPTIMISATION

Most of the forging sequences are made of several stages. Usually, the *parameters* are defined on the first stages while *constrains* or *minimisables* have to be measured on final stages. The first thing is then to be able to compute without any manual intervention the whole chain which of course may include 2D stages and 3D stages. This feature has been available in Forge since 2007. The second point is that MAES type optimization makes it theoretically possible to run several simulations at the same time. Clusters or new SMP's are ideal hardware to achieve this. Of course, to take benefit of such a facility, a batch managing system is needed in order to compute the simulations as soon as it is compatible with the algorithm and with respect to the machine availability. This feature was introduced in 2005.

3.2 CONVENIENT LIST OF MINIMISABLEZ, PARAMETERS AND CONSTRAINTS

Most of the general purpose automatic optimisation packages offer facilities to scan through simulation result files, extract convenient numbers and evaluate *minimisables* or *constraints*. They can also edit and replace some values in the simulation input desk to modify parameters. Coupling simulation and automatic optimisation is then possible but it requires a lot of skill from the user. He has to map without help simulation results on his real optimisation case and he has to know precisely how the needed information is stored. This point is a real difficulty. 20 years ago result data structures were simple and people were used to text edit files to find what was needed. This time is gone and editors have used the fact that users only need to see results to optimise, but also to make result data structures more complex. An integrated package not only suggests *minimisable*, *constraints* and *parameters* but also manages all the dialogs between the simulation tool and optimisation kernel.

3.3 UNIFIED GUI

During the setup of an optimisation, the user needs continuously to go from the optimisation definition to the standard process pre-processing. Back to our initial example, the user will want to check graphically that the bounds of the billet displacement are compatible with die geometry, and this without switching packages. The same situation occurs also in post-processing phase. Optimisation kernels will tell you that the optimum has been reached for a given computation and with a given set of *parameters*. It is then very convenient to select by one click this specific computation and post-process it as usual to check that the results are really what was expected.

4 EXAMPLES

4.1 PREVIOUS EXAMPLES

Several examples have or will be presented in [1], [2] and [3]. They include:

a) Billet volume optimisation for common rails, crankshafts and plates under several different constraints.

b) Process parameters optimisation in cogging and cable drawing to improve part quality.

And many others have been undertaken within the LOGIC project. Our goal here is to demonstrate that the same software can be used to achieve other targets.

4.2 BRASS PART OPTIMISATION

It is very common with brass to produce complex parts in a one stage forging sequence. The example presented in Figure 1 is one of the first which has been computed. The original idea was to find the smallest cylindrical billet which would fill the die. The billet lies on the lower die aligned on the Y axis. The *minimisable* is the billet volume. *Parameters* are billet diameter, length and X position. The *constraint* is die filling.

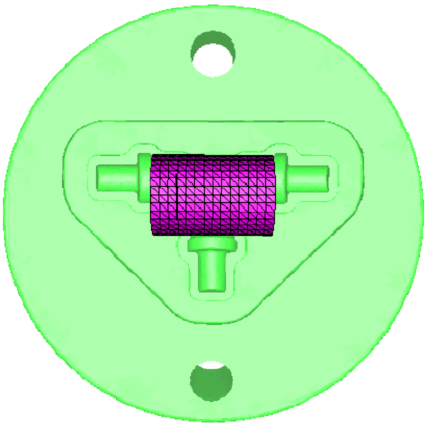

Figure 1: Initial setting, Billet on the lower die

The risk with this one stage complex part forging is to create large folds and, after a few computations, it was obvious that the optimization kernel was leading us to such a problem.

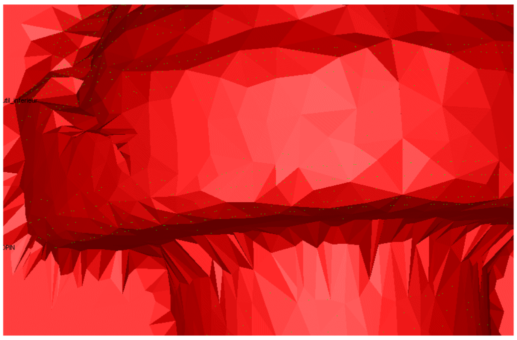

Figure 2: Major fold

We had to restart adding another constraint which is 'No folding'. The final results can be seen in Figure 3.

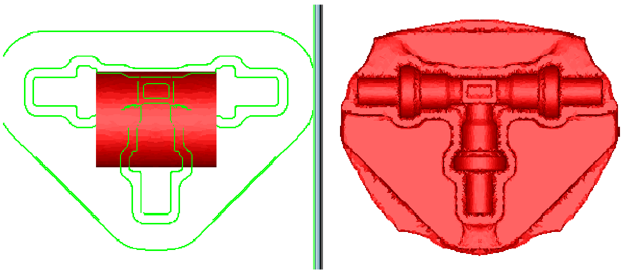

Figure 3: Best obtained results

4.3 2 SHEETS RIVETING OPTIMISATION

It is well accepted that part in-use properties may depend heavily on the forming process they underwent. Cable drawing and Cogging examples quoted in [1] and [2] try to improve part properties through the minimisation of defect criteria. In this example we try to apply a more straightforward approach to optimize a riveting sequence with a very small rivet. Thanks to Forge's automatic chaining facilities we have created a 3-stage sequence (riveting, spring back and loading) and we are looking for the smallest rivet which gives a controlled upper sheet displacement under a predefined loading (In this case, we grip the lower sheet and pull vertically the upper one). The rivet geometry variations are described with 2 *parameters* (diameter from 0.8mm to 1.2 mm and length from 1.6 mm to 2.14mm); other dimensions are deduced automatically. The *minimisable* is the rivet volume and the *constraint* is that, under the load (20kg), the sheet displacement has to remain under a prescribed value (In this case 0.02 mm). The shape of the upper forming die is adapted each time to match the rivet geometry while the shape of the lower one is kept flat. In any case the force on the moving die is limited to 100kg. In this case we have used 10 generations of 4 individuals.

Figure 4 represents the end of the riveting (left) and Z displacement under pulling. Same scaling is used of all the figure from blue= 0 to red= 0.05 mm

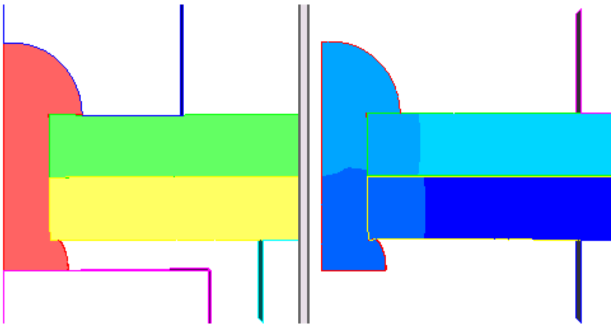

Figure 4: Best case

Figure 5 shows the same situation with a thinner and shorter rivet. Under the traction, the rivet slides a little bit and is elongated.

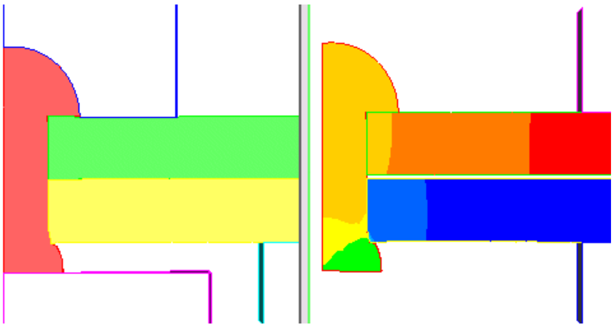

Figure 5: A to thin rivet

There are of course many cases where the rivet is able to hold the sheet but is just bigger than the best one. The last presented one is more interesting. The rivet has a larger diameter and is long enough so, it should be OK but in fact, the diameter is too big and the lower die doesn't have enough force to form the lower part of the rivet. When the upper sheet is pulled, it goes away bringing the rivet

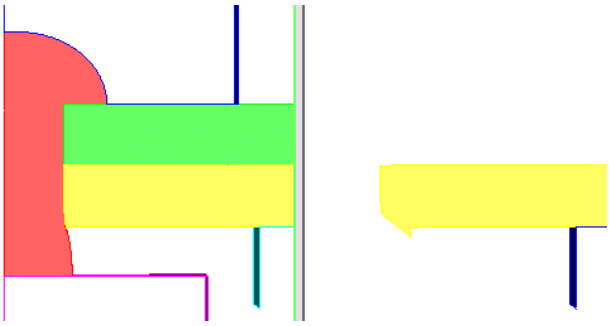

Figure 6: A too large diameter prevents the forming of the rivet. Under traction sheet and rivet go away

In this case the algorithm convergence is fast and Figure 7 shows that after 4 generations, all individuals are close to the optimum. Abscissa axis represents the volume and ordinate displacement above the *constraint.*

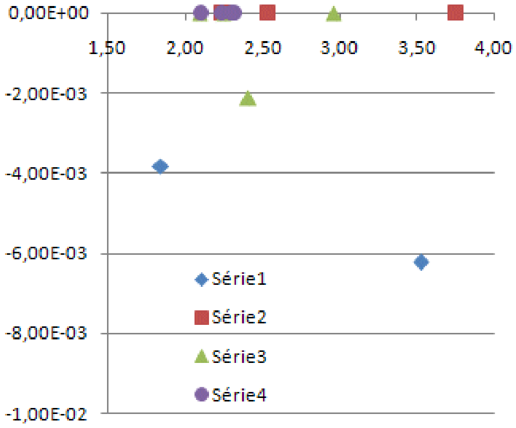

Figure 7: Individual of The 4 first generation in the *minimisable/constraint* plane.

4.4 THERMAL EXCHANGE COEFFICIENT INVERSE ANALYSIS

Thermal exchange coefficients as well as friction coefficients are often a tedious issue if one wishes to setup precise process parameters, because they don't only depend on one object. In case of quenching, the coefficients depend on the part, the cooling media, the geometry of the cooling device and many other parameters. A simple observation shows that, during the quenching, there are different phases linked with the boiling of the cooling liquid. It can be interesting then to use a model where coefficients would be a function of the part temperature which obviously has a large influence on the boiling. In the presented example the coefficient is defined at 5 predetermined temperatures and; between 2 temperatures, linear interpolation is made. To determine these 5 values, a part containing 17 thermocouples has been quenched, the purpose being to find the parameters which would give the best fit. This is the typical repetitive and systematic activity on which automatic optimisation is a real help. *Parameters* are the value of exchange coefficient, *minimisable* is the sum of the quadratic differences between experimental and numerical curves for the 17 thermocouples and there is no *constraint*. In this example, we have used 30 generations of 20 individuals. Figure 8 represents the experimental data (lines with markers), the continuous line is the best obtained fit and the dotted line is the initial value set. The different colours correspond to 3 different thermocouples selected as far as possible from one another. At least the fit is much better than the one obtained with the original set. To go further, the model should probably be improved including at least surface orientation.

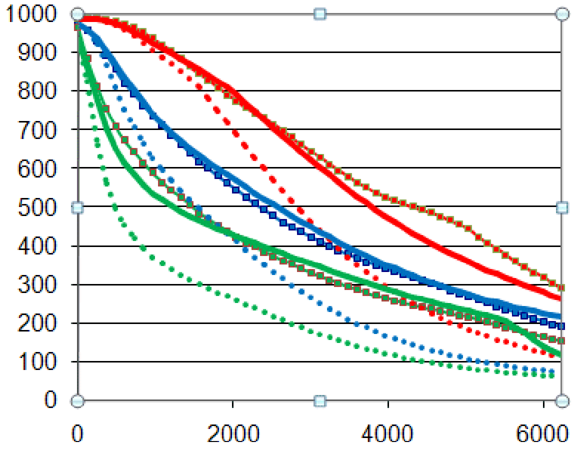

Figure 8: Experimental and computed temperatures

5 CONCLUSION

Hopefully, we have demonstrated the versatility of the method. All the presented examples both in this paper and in the quoted ones have been done using Forge2009 beta or Forge 2009 release versions. Of course progress can still be made on several aspects: multi-objective, post-processing, more adapted minimisables and constrains … but the idea was to offer something that already has customer added value. On the long term, we wish to spread the use of this feature and get maximum feedback to continue to improve it.

REFERENCES

[1] L. Fourment, T. Massé, S. Marie, R. Ducloux, M. Ejday, C. Bobadilla, P. Montmitonnet: Optimization of a range of 2D and 3D bulk forming processes by a meta-model assisted evolution strategy. 12th ESAFORM conference on material forming. Enschede, 27-29 April 2009

[2] R.Ducloux, L. Fourment, S. Marie, D. Monnereau: Automatic optimisation techniques applied to large range of industrial test cases", 13th international ESAFORM Conference, Brescia, Italy, April 7-9, 2010

[3] L. Fourment, R.Ducloux, S. Marie, M. Ejday, D. Monnereau, T. Masse, P. Montmitonnet: Mono and Multi Objective optimisation techniques applied to a large range of industrial test cases using metamodel assisted evolutionary algorithms", 10th international NUMIFORM Conference, Pohang, Korea, June 13-17, 2010

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