Optimisation and Confidence Region Estimation of Constitutive Parameters in Equations for Hot Metal Deformation

Mulyadi[1], M.A.Rist[1], L.Edwards[1], J.W.Brooks[2]

[1]*Department of Materials Engineering, The Open University, Milton Keynes, MK7 6AA, United Kingdom*
URL: http://materials.open.ac.uk *e-mail: Mulyadi@open.ac.uk; m.a.rist@open.ac.uk; l.edwards@open.ac.uk*

[2]*The Structural Materials Centre, Qinetiq Plc, Farnborough, Hampshire, GU14 0LX, United Kingdom*
URL:www.qinetiq.com *e-mail: J.W.Brooks@ qinetiq.com;*

ABSTRACT: This paper surveys a range of numerical methods for the optimisation of material parameters in a single state-variable constitutive model applied to the hot deformation of Ti-6Sn-2Zr-4Mn-6Mo. Minimisation of an objective function, describing the least squares difference between experimental results and model output, was carried out using several methods: Nelder-Mead direct search, quasi-Newton gradient, trust-region Newton and genetic algorithms. The most robust approach involved gaining an initial estimate of parameter values using the genetic algorithms and then refining this with a Nelder-Mead search. A scheme for estimating the confidence limits of the derived parameters is also presented.

Key words: constitutive model, non-linear optimisation, hybrid function, confidence limit

1 INTRODUCTION

The forming of metals at elevated temperature involves the imposition of non-uniform deformation rate and temperature paths to large strains, so that the material undergoes significant, often beneficial, microstructural change. Effective modelling of the deformation behaviour therefore requires constitutive equations that describe the evolution of strain (or strain-rate) in terms of stress, temperature and suitable internal microstructural variables. Such formulations are usually semi-empirical and necessarily complex, often involving highly non-linear, power-law or exponential relationships that incorporate a number of material-specific constants. The conventional approach has been to determine these material parameters from well-constrained laboratory experiments, often using graphical methods to reduce the problem to a linear relationship between two of the variables at a time, all the others being assumed constant. However, there is an increasing trend towards the use of numerical optimisation methods that seek to find all the parameters in a set of equations simultaneously. These numerical methods work by iteratively altering model parameter values in order to minimise the difference between prediction and experimental results, as computed through an 'objective function'. Careful selection of the most appropriate optimisation method and initial parameter starting point is necessary, otherwise convergence to a global minimum is unlikely to be achieved, and poor parameter estimation will result.

Here we compare several popular optimisation techniques and propose a hybrid methodology for initiating and completing a multi-parameter search in non-linear constitutive equations describing material deformation. A state-variable hot-deformation model for Ti-6Sn-2Zr-4Mn-6Mo, a popular commercial α-β titanium alloy widely used in forged aero-engine components, is adopted for this purpose.

2 CONSTITUTIVE MODEL

The constitutive model used here takes account of combined flow hardening and softening by assigning the weighted sum of deformation mechanisms to a single internal state variable, λ, as discussed in detail elsewhere [1]. λ is assumed to evolve as an exponential function of strain, ε, towards a steady-

state condition, λ_{ss}, where $\lambda_{ss}=\lambda_0 Z^q$ and $Z=(d\varepsilon/dt)exp(Q/RT)$:

$$\lambda = \lambda_{ss} + (1-\lambda_{ss})\exp(-\alpha\varepsilon) \quad (1)$$

The flow stress σ is then defined as

$$\sigma = K\lambda Z^m(1-\exp(-\beta\varepsilon)) \quad (2)$$

where λ_0 and K are scaling constants, m is the rate sensitivity, q is a structure sensitivity parameter, Q is the apparent activation energy and R the universal gas constant. K, λ_0, Q/R, m, q, α and β, are material constants to be determined from experimental data.

3 MODEL PARAMETER DETERMINATION

3.1 Determination of Starting Values

The starting values of material constants can be determined based on experimental data by using a combination of graphical and analytical methods. The experimental data examined here were obtained from nine isothermal compression tests on Ti-6Sn-2Zr-4Mn-6Mo at temperatures of 825°C, 875°C and 925°C at each of three strain rates: 0.001, 0.01 and 0.1/s [2]. The flow stress data were first analysed using conventional methods to obtain 'as-calculated' starting values for the parameters in the constitutive model. From curves of initial stress, σ_I, and steady-state stress, σ_{ss}, *versus* temperature, T, intermediate temperature values corresponding to a set of arbitrary constant stresses were determined. These temperature values were then used to derive an average value of Q/R from an Arrhenius plot of $1/T$ *versus* strain rate. The derived activation energy was then used to calculate Z, in order to obtain the strain rate sensitivity, m. A log-log plot of λ_{ss} against Z then allows q and λ_0 to be determined where $\lambda_{ss}=\sigma_i/\sigma_{ss}$. K was estimated from the average value of $\sigma_{ss}/\lambda_o Z^{m+q}$. α and β were assigned arbitrary positive real value. A summary of material constants derived in this way is presented in Table 1.

Table 1. Constitutive data for Ti-6Al-2Sn-4Zr-6Mo

K	λ	m	q	α	β	Q/R
0.00568	4.900	0.251	-0.0498	5	148	49952.36

The above constitutive data was used as a starting point for several optimization algorithms in order to refine the fit between model and experiment.

3.2 Optimisation of Material Parameters

Unknown design variables $\mathbf{x} = [x_1\ x_2\ x_3\ \ldots x_i]$ are sought by fitting model results to experimental data y_j ($j = 1, 2, 3,\ldots, p$), where i and p are the number of unknown parameters and data points respectively. Such parameter optimisation is best carried out by minimising an objective function, usually the least squares difference between experimental data and modelled output *i.e.*

$$F(\boldsymbol{x}) = \sum_{j=1}^{p}(\sigma_j - \sigma_j(\boldsymbol{x}))^2 \quad (3)$$

where σ_j is the experimental stress and $\sigma_j(\mathbf{x})$ is the stress value obtained from the model. Individual algorithms used for the iterative multidimensional non-linear optimisation are described below.

3.2.a Nelder-Mead Simplex Method

The Nelder-Mead direct search algorithm [3] evaluates an objective function at several points without calculating derivatives. The substantial element is a geometric figure called a simplex that is defined as a set of $i + 1$ points in i-dimensional space. At each iteration new points are calculated, along with their function values, to create a new simplex. The simplex can move, shrink and expand toward a minimum. The algorithm terminates when the function values at the vertices of the simplex satisfy a predetermined condition.

3.2.b Quasi-Newton

A large class of algorithms, collectively called Newton methods, make use of gradient information to minimise a function of several variables. A quadratic approximation (second-order Taylor series expansion) is used to model a step change, s, in objective function $F(\boldsymbol{x})$. At the kth iteration the model function is given by:

$$m_{(k)}(\boldsymbol{s}) = a + \boldsymbol{g}_{(k)}^T\boldsymbol{s} + \frac{1}{2}\boldsymbol{s}^T H_{(k)}\boldsymbol{s} \quad (4)$$

where $a = F(\boldsymbol{x}_{(k)})$, $\mathbf{g} = \nabla F(\boldsymbol{x})$ is the $(1\times i)$ gradient vector of first partial derivatives of $F(\boldsymbol{x})$ (the well-known Jacobian matrix), and $H = \nabla^2 F(\boldsymbol{x})$ is the $(i\times i)$ Hessian matrix of second partial derivatives. At its minimum Eqn-(4) becomes $\boldsymbol{s}_{(k)}=H_{(k)}^{-1}\boldsymbol{g}_{(k)}$ and step $\boldsymbol{s}_{(k)}$ leads directly to the minimum of the model function, but not necessarily the real function – in which case the next iteration is $\boldsymbol{x}_{(k+1)} = \boldsymbol{x}_{(k)} + \boldsymbol{s}_{(k)}$. In many optimisation problems, including the one considered here, it is not possible or desirable to

supply the Hessian matrix explicitly. Rather, an approximation to $H_{(k)}$ is built up using gradient information from some or all of the previous steps – this is the quasi-Newton approach. We have used the Broyden-Fletcher-Goldfarb-Shanno (BFGS) method for updating H, widely believed to be the most effective technique for general use, as implemented in MATLAB Optimisation toolbox function 'fminunc' [4].

3.2.c Trust Region Newton Method

The quadratic approximation of $F(\boldsymbol{x}_{(k)}+\boldsymbol{s})$ given in Eqn-(4) is not valid when $\boldsymbol{s}$ becomes too large. Trust region methods place a restriction on the size of the step such that $\|D_{(k)}.\boldsymbol{s}\| \leq \Delta_{(k)}$ where $D_{(k)}$ is a diagonal scaling matrix and $\Delta_{(k)}$ is a positive scalar representing the size of the region over which the approximation can be 'trusted'. The magnitude of $\Delta_{(k)}$ is based on the agreement between the model function and the objective function at each step, as measured by the ratio:

$$p_{(k)} = \frac{F(\boldsymbol{x}_{(k)}) - F(\boldsymbol{x}_{(k)} + \boldsymbol{s}_{(k)})}{F(\boldsymbol{x}_{(k)}) - m_{(k)}(\boldsymbol{s}_{(k)})} \tag{5}$$

If $p_{(k)} \approx 1$ there is good agreement and the trust region can be expanded for the next iteration; if $p_{(k)}$ is close to zero or negative the trust region must be made smaller. Trust region methods are generally regarded to be faster and more stable than quasi-Newton methods. For this study we have used a bounded trust region method implemented in MATLAB optimisation toolbox function, 'lsqnonlin' [4]. The bounds of design variables were fixed within the range: $0<K<1$; $0<\lambda_0<1$; $-1<q<1$;$0<m<1$; $1<\alpha<20$; $20<\beta<200$.

3.2.d Simplex-Genetic Algorithm Hybrid Function

Rather than use the initial calculated parameter values given in Table 1, we have developed a hybrid approach whereby genetic algorithms are used to find a staring point, and then the simplex method is applied to obtain a global minimum. Genetic algorithms are based on probabilistic methods that use a population of design variables derived using concepts of biological natural selection [5]. Initial population members are generated randomly and are represented by binary strings. To evaluate each member, a fitness function (the objective function) is implemented. Members for the next generation are chosen based on these fitness values. A member with a greater fitness value will have a higher probability of being chosen for the next generation than one with a lower fitness value. Genetic algorithms uses crossover and mutation techniques. Crossover is the exchange of bits of binary string between parents to create children. Mutation occurs when a bit in a string changes randomly from 0 to 1 or 1 to 0.

4 CONFIDENCE INTERVALS

Parameter confidence intervals can be calculated from the Hessian, H of $F(\boldsymbol{x})$ at the solution. The least-squares problem of Eqn-(3) can be rewritten as:

$$F(\boldsymbol{x}) = \sum_{j=1}^{p} f_j^2(\boldsymbol{x}) \;\; ; \; f_j(\boldsymbol{x}) = (\sigma_j - \sigma_j(\boldsymbol{x})) \tag{6}$$

$$H = 2(\boldsymbol{g}^T\boldsymbol{g} + \sum_{j=1}^{p} f_j(\boldsymbol{x}) H_j) \tag{7}$$

where H_j is the Hessian matrix of each $f_j(x)$. In the neighbourhood of the solution, $\|f(\boldsymbol{x})\|$ is often small compared to $\|\boldsymbol{g}^T\boldsymbol{g}\|$. In such cases, H is sufficiently approximated by $H \approx 2\boldsymbol{g}^{\mathrm{T}}\boldsymbol{g}$. The Jacobian, $\boldsymbol{g}$ is returned by all the comprehensive least-squares routines and, in addition, a routine is available in the NAG Fortran Library [6]. An unbiased estimate of the variance of the ith parameter x_i is

$$\operatorname{var} x_i = \frac{2F(\boldsymbol{x})}{v-w} diag(H)^{-1} \tag{8}$$

If x_i^* is the true solution, then the 100(1-φ)% confidence interval on x_i is

$$x_i - \sqrt{\operatorname{var} x_i}.t_{(1-\varphi/2,v-w)} < x_i^* < x_i + \sqrt{\operatorname{var} x_i}.t_{(1-\varphi/2,v-w)} \tag{9}$$

where $t_{(1-\varphi/2,\mathrm{v\text{-}w})}$ is the 100(1-φ)/2 percentage point of the t-distribution with v-w degrees of freedom.

5 RESULTS AND DISCUSSION

The optimised parameters obtained using the methods outlined in Sections 3.2.a-d are presented in Table 2. The performance of each algorithm can be assessed from the table in a number of ways. The quality of fit is represented by (i) the final value of the objective function and (ii) the percentage of mean variation, *%MV*, per data point (defined as the absolute difference between modelled $\sigma_j(\boldsymbol{x})$ and experimental stress σ_j, divided by the number of data points p). Overall uncertainty is measured by the stated confidence intervals and the efficiency and

rapidity of convergence can be assessed from the number of iterations and function evaluations required to reach the optimised value.

Table 2 indicates that the Nelder-Mead and hybrid algorithms obtain the lowest values of the objective function and are also the the most computationally efficient methods. The reflective-Newton method also reached an acceptable minimum but required more iterations; the quasi-Newton algorithm failed to converge. Figure 1 shows examples of experimental and modelled flow curves, optimized using the Nelder-Mead method.

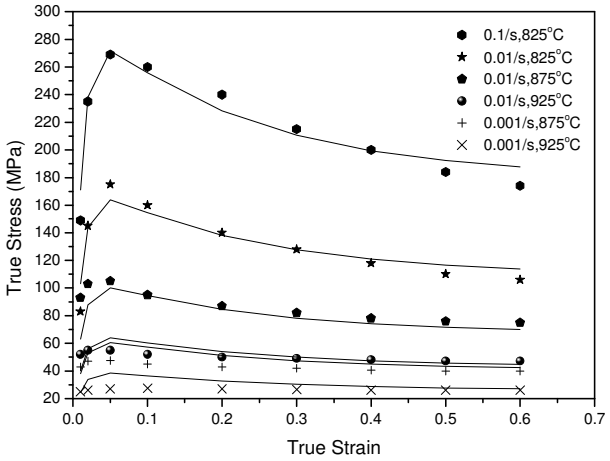

Fig. 1. Experimental and modelled data based on Nelder-Mead

Table 2. Optimised parameters and their confidence limits

Material Constants	Nelder-Mead	Reflective-Newton	Quasi-Newton	Hybrid Function
K ($\times 10^{-3}$)	6.14 ± 4.87	6.87 ± 5.29	12.96 ± 15.50	6.14 ± 4.87
λ_0 ($\times 10^{-1}$)	6.86 ± 7.98	6.86 ± 7.89	7.05 ± 2.26	6.86 ± 7.98
Q/R ($\times 10^{4}$)	5.65 ± 0.41	5.55 ± 0.40	4.99 ± 0.00011	5.65 ± 0.41
M ($\times 10^{-1}$)	2.19 ± 0.16	2.21 ± 0.25	2.33 ± 0.26	2.19 ± 0.16
q ($\times 10^{-3}$)	-2.70 ± 24.83	-2.67 ± 25.15	-3.89 ± 7.95	-2.70 ± 24.83
α	4.50 ± 1.96	4.47 ± 1.96	4.25 ± 3.73	4.49 ± 1.96
β (x10)	8.72 ± 1.29	8.73 ± 1.29	8.81 ± 1.73	8.72 ± 1.29
$F(x)$($\times 10^{3}$)	6.50	6.52	7.35	6.50
%MV	8.42	8.4	8.73	8.42
Iterations	1849	9074	> 100000	1121
Fun. Eval	2466	19905	> 900000	1724

A weakness of many traditional algorithms, including Nelder-Mead, is the requirement of initial parameter values. The proposed genetic algorithm-simplex hybrid system overcomes this problem by allowing a starting point to be chosen from a random population. Table 2 shows that performance is similar to that for the stand-alone simplex method.

Table 2 also indicates that, given the small data set, confidence limits of most of material constants are acceptable. However, large uncertainties (>100%) were found for λ_0 and q. This could be due to the weakness of the $\lambda_{ss}=\lambda_0 Z^q$ term in the constitutive model (Section 2). Note that the derived value of q is close to zero, so if we assume that $q \approx 0$, then $\lambda_{ss}=\lambda_0$ and the number of material parameters is reduced, thus simplifying the model. In fact, on applying this assumption it was found that the confidence limits of all the parameters moved to within a tolerable range, thus justifyng a refinement in the constitutive equations.

6 CONCLUSIONS

Methods for parameter optimisation in a non-linear constitutive model for hot deformation have been compared. It can be concluded that:

- the Nelder-Mead simplex method shows the best performance as indicated by the final objective function value, and by the number of iterations and function evaluations required for optimisation,
- a hybrid method combining genetic algorithms and a simplex search has promising applications where initial value estimates are difficult or time consuming
- confidence limit estimates can be used to reasonably assess the form and validity of the applied constitutive model, as well as the degree of fit.

REFERENCES

1. J.C.F.Millet, J.W.Brooks and I.P.Jones, Assessment and modelling of isothermal forging of intermetallic compounds Part 1-TiAl, *Materials Science and Technology*.15 (1999), 697-704
2. M.Jackson, PhD Thesis, Department of Materials, Imperial College of Science, Technology and Medicine, University of London, UK, 2002
3. J.A.Nelder, R.Mead, A simplex method for function minimization, *Computer Journal*.7(1965),308-313
4. MATLAB software, version 6. (Natick, MA : The MathWorks Inc., 2002).
5. D.E.Goldberg, Genetic Algoritms in Search, Optimization and Machine Learning, chapter 1. Addison-Wesley Longman Publishing, Boston, MA, 1989
6. The Fortran77 Library, The Numerical Algorithms Group Ltd, Oxford, UK