Materials and Manufacturing Processes 24 (2009) 53-58

Domains of Steels with Identical Properties

Minsung Joo, Joohyun Ryu and H. K. D. H. Bhadeshia
Graduate Institute of Ferrous Technology (GIFT)
Pohang University of Science and Technology (POSTECH), Korea

Abstract

There have been many attempts in the past to reduce the variety of steels produced without compromising the range of properties available. The objective in doing so is to reduce costs. These attempts have largely focused on experiments in which thermomechanical processing parameters are varied in order to obtain a range of properties without changing the chemical composition.

We have approached this problem differently by examining a large database on hot-rolled steels, all with a final microstructure which is a mixture of ferrite and pearlite. Alloys like these form the bulk of all the steel production in the world. The database was used to create a neural network model in a Bayesian framework with inputs consisting of the chemical composition (C, Mn, Si, P, S, Cr, Ni, Mo, Ti, Nb, V, Al, N, B, Cu), the finish deformation temperature (T_F) and the coiling temperature (T_C); the output was either the ultimate tensile strength or tensile elongation to failure. The neural network model was then combined with a genetic algorithm to seek domains of input parameters which correspond to different target properties. The results revealed ambiguity in the inputs with respect to the required property, and the trends observed within these sets of inputs were found to be metallurgically meaningful..

Keywords: ambiguity, steel, mechanical properties, genetic algorithms, neural network

1. Introduction

Hot-rolled steels with a microstructure consisting of a mixture of ferrite and pearlite form the backbone of the steel construction industry because of the unique combination of strength, toughness and cost that they offer. They are also easy to fabricate using processes such as welding. It is possible that a reduction in the variety

of steels available could result in greater economies in the production process [1].

One approach to investigating this problem would be to conduct a well-designed series of experiments, but this is in principle unnecessary because a vast quantity of data already exists in the form of research, production and quality-control experiments; these data have been accumulated over the years as a function of many variables and could be exploited both to develop quantitative relationships and to identify domains where further experiments are necessary.

The most general way of developing quantitative relationships when dealing with complex problems is the artificial neural network [2-7]. It is ideal for the present problem as a means to express the relationship between the large number of variables which determine the properties of the steel, and the properties themselves. The method has been applied to hot-rolled steels [8-10]. Indeed, it has been used widely to model various aspects of commercial steels [11-17].

On the other hand, once the phenomenon has been modelled, the neural network like many other modelling techniques, requires the user to reach the desired solution by a trial and error choice of inputs. The user has to make educated guesses on what the inputs to use in order to reach the target value of the output parameter. This is akin to experimental programmes of research and suffers from the same issues, that an element of bias is introduced by the user in exploring the input space.

The problem can be resolved by combining the method with a genetic algorithm which in its simplest form efficiently searches the input space [18,19]. The power of doing this has recently been demonstrated in reaching novel solutions out of the field of established knowledge, leading to the discovery of the δ-TRIP steel [20,21]. The combination of a neural network model with a genetic algorithm can allow the user to set the goals and let the algorithm discover the domain of inputs which can satisfy the conditions (output and constraints) set at the outset. The use of a Bayesian framework [2-5] for the network also allows the possibility of setting limits to the modelling uncertainty when searching the input space. In this way, outrageous solutions can be omitted.

In this paper, we attempt to follow the procedure described above to identify the range of variables which lead to steels which have the required tensile strength or elongation;

the modeling uncertainty is used to control the search process. We begin with a brief description of the neural network – details can be found elsewhere [2-6], followed by a detailed description of the genetic algorithm and then the outcomes of the analysis for selected cases.

2. Neural Network Model

A neural network is a general method of non-linear regression analysis [2-6]. It is composed of input nodes, hidden units and an output node. The inputs x_j such element concentrations define the input nodes and the output such as tensile strength defines the output node. Each input is multiplied by a random weight w_j and the products are summed together with a constant θ to give the output $z = \sum_j w_j x_j + \theta$. And z will be the argument of a hyperbolic tangent transfer function $h_i = \tanh(z)$. Each h_i is itself multiplied by a weight w_i. The sum of these hyperbolic tangent with a second constant $\theta^{(2)}$ gives the output y as a non-linear function of x_j as shown in following equation.

$$y = \sum_i w_i h_i + \theta^{(2)} \tag{2-1}$$

A potential difficulty with the use of powerful non–linear regression methods is the possibility of overfitting data. To avert this difficulty, the experimental data can be divided into two sets, a training dataset and a test dataset. The model is produced using only the training data. And then the test data are used to check that the model behaves well.

The overall error in the neural network model, E_D, is calculated by comparing the predicted values y_j of the output against those measured t_j

$$E_D \propto \sum_j (t_j - y_j)^2 \tag{2-2}$$

E_D is expected to increase if important input variables have been excluded from the analysis. An alternative measure of fitting error is the log predictive error (LPE) which helps to minimise the influence of outliers:

$$LPE = \sum_j [\frac{1}{2}\frac{(t^{(j)} - y^{(j)})^2}{\sigma_y{}^{(j)^2}} + \log(\sqrt{2\pi}\sigma_y{}^{(j)}] \tag{2-3}$$

One of the most important advantages of MacKay's method is that in addition to the

measures of noise described above, the method also identifies a modelling uncertainty [3]. This is a recognition of the fact that many functions may adequately represent known data and yet extrapolate differently. The variation in the extrapolation behaviour is an indication of modelling uncertainty and is a key indicator in two respects. First, it gives an immediate indication of regions of the input space where knowledge is sparse and secondly, a large modelling uncertainty gives a clear indication of the input parameters for which new experiments are desirable. Thus, instead of using just a best-fit set of weights, a distribution of weights is calculated.

Finally, several different of the best-performing models are combined together to produce an optimised committee of models to allow more reliable predictions.

3. Genetic Algorithm

A genetic algorithm is a search technique used in computing to find true or approximate solutions to optimization and search problems. It is a particular class of evolutionary algorithms that use techniques inspired by evolutionary biology such as inheritance, mutation, selection, and crossover [18,22-24]. It can be implemented as a computer simulation. The general process of the algorithm is illustrated in Fig. 1. The initial population represents a possible solution group for the given problem. Each possible solution is called an *individual*. The problem is encoded in a series of bit strings which are then manipulated by the algorithm. These bit strings are coded representations of input variables.

While in general optimization algorithms, the searching is done from a single initial set of inputs, genetic algorithms operate from many initial sets which are created before the algorithm operates. The choice of the initial sets can be important in influencing the computational efficiency of the optimization process.

To decide whether a solution is meaningful in the context of the goal, it is necessary to define a *fitness function* which helps evaluated the suitability of each individual set. It determines how each individual is suited to the objective, and hence its ability to survive in the next iteration of the algorithm.

Each model i created by the neural network gives a result $y^{(i)}$ and the associated error $\sigma_{y^{(i)}}$. The average prediction of a committee of L models is:

$$p = \frac{1}{L}\sum_{i} y^{(i)} \tag{3-1}$$

The standard deviation error (σ) of p is as follows:

$$\sigma^2 = \frac{1}{L}\sum_{i} \sigma_y{}^{(i)2} + \frac{1}{L}\sum_{i} (t-p)^2 \tag{3-2}$$

where t is the desired output. The score of an individual could be σ.

The process of selecting the fittest input sets is inspired by the theory of natural selection. An evolutionary algorithm performs a selection process in which the most fit individuals of the population survive, and the least fit individuals are eliminate. There are many ways to select the survivors. In this paper, the fitness proportionate selection, also known as roulette-wheel selection is used.

In fitness proportionate selection, the probability of selection of an individual i is determined as:

$$P_i = \frac{f_i}{\sum_{j=1}^{N} f_j} \tag{3-3}$$

where f_i is the fitness of the ith individual and N is the number of individuals in the population. Thus, the best solutions or chromosomes will be selected from those with the greater probability. As this method prevents the rapid lowering of diversity, it helps avoid a premature convergence of solutions. *Reproduction* is used to copy fitter populations to the locations left empty when unfit individuals are eliminated.

Further procedures are needed to ensure the avoidance of local minima and permit attempts which deviate from the norm. *Crossover* is used to vary the nature of individual between generations, by taking genes from two parents, mixing them and producing an offspring. There are many ways to do crossover. In this paper, the uniform-crossover method is used in which individual bits in the string comprising one individual are compared between two parents. The bits are swapped with a fixed probability, typically 0.5.

Mutation is used to maintain genetic diversity from one generation of a population to the next. It periodically makes random changes in one or more members of the current population, yielding a new candidate solution. It particularly helps in a wider exploration

of the input space, thus avoiding local minima.

4. Results

We have used here models based on a total of 3508 experiments, for the ultimate tensile strength and for the total tensile elongation, as a function of the concentrations of C, Mn, Si, P, S, Cr, Ni, Mo, Ti, Nb, V, Al, N, B, Cu, the finish deformation temperature T_F and the coiling temperature T_C.

Given the large number of variables, it was found necessary to implement at most 10,000 generations to reach reasonable answers. Because in the previous research [19, 24], the fitness scores converged within 5,000 generations, 10,000 was assessed to be sufficient to reveal the domains corresponding to specific properties. The population size was fixed at 20 [19, 24,25].

The crossover rate is typically 80-95% [25] but a lower value of 60% was used in order to keep well-fitting solutions for as many generations as possible. The mutation rate was set between 0 and 0.2%. All the results are for reasons of clarity stated to a precision in four significant figures after the decimal point; this should not imply that measurements can be made to that accuracy and uncertainties must be considered alongside the estimated values.

As mentioned in section 3, the neural network committee model has two outputs; the result and the uncertainty. The fitness function should therefore consider both of these parameters. Therefore the fitness *f* is:

$$f = \frac{1}{\sigma} \tag{4-1}$$

Tensile Strength Model

Fig. 2 shows the significance of each of the inputs, as perceived by the first five models in the neural network committee model which is combination of 11 different of the best performing individual models for the ultimate tensile strength. As expected, C, Mn and Si are most significant in influencing the strength.

None of the input parameters were constrained in running the composite neural network – genetic algorithm model, though limits were placed on the size of the

modelling uncertainty in order to avoid results which simply may not make metallurgical sense.

The first analysis conducted involved a search for the domain of inputs which would lead to an ultimate tensile strength of 400 MPa, allowing for a ±15% uncertainty including computational errors. Some 94 combinations of inputs were found to be consistent with this criterion. A selection of these data is plotted in Fig. 3, illustrating particularly the carbon, manganese, silicon, finish-rolling and coiling temperatures, together with the strength. It is interesting to see that low carbon concentrations (e.g., samples 4,5,8 and 9) are automatically associated by the model with low coiling temperature T_C in order to maintain a value of strength close to 400 MPa. This is expected since the increase in strength due to carbon is compensated for by a reduction in cooling rate due to the use of a high T_C. Similarly, large manganese concentrations are compensated for by low carbon concentrations [e.g., 26,27]. The range of the key inputs which lead to the required strength are as follows:

400 MPa target	C / wt%	Si / wt%	Mn / wt%	T_C / °C	T_F / °C
Minimum	0.11	0.02	0.70	596	828
Maximum	0.20	0.11	0.79	709	983

There are of course many more variables involved in the analysis but they have a relatively minor effect and hence are not listed here. The important point is that a significant range of inputs can lead to the required strength of 400 MPa, with the analysis coming up with a total of 94 solutions.

The number of solutions decreased to 75 and 18 when the required strength was increased to 600 and 800 MPa respectively. For brevity we illustrate the change using just the 800 MPa simulation. It is clear from the data listed below and from Fig. 4, that not only has the number of available solutions decreased, but the permitted range of the inputs has decreased significantly. The permitted ranges for C, Si, T_C and T_F decrease from 0.09 wt% to 0.03 wt%, 0.09 wt% to 0.00 wt%, 113 °C to 26 °C and 155 °C to 39 °C respectively. It is indicating that to achieve high strength in this alloy system would require careful control, perhaps beyond the accuracy available in practice:

800 MPa target	C / wt%	Si / wt%	Mn / wt%	T_C / °C	T_F / °C
Minimum	0.44	0.11	0.70	596	771

Maximum	0.47	0.11	0.79	622	810

Another interpretation is that the alloy system considered in creating the model is too simple to allow flexibility in the choice of inputs when the target strength is high. In contrast, the 400 MPa is typical of ferrite-pearlite steels [28,29].

Elongation Model

Fig. 5 indicates the significance of each of the inputs, as perceived by the all individual models in the neural network committee model which is combination of 3 different models for the elongation to failure. This shows that carbon and manganese are the most significant determinants of the elongation in the neural network model which included the same as the tensile strength analysis. The model was combined with a genetic algorithm. Some 129 solutions were found when the target elongation was set at 35% and 175 for 55% elongation.

Target of 35% is used to check the behavior of the combining of neural network and genetic algorithm. This target is easy to achieve. It is about average of dataset which was used to make models. Target of 45% is to check the effect of combining of neural network and genetic algorithm. This value is higher than the average elongation of ferrite-pearlite steels [9][10]. Target of 55% is used to test the high values of elongation which we can't reach easily.

In 35% case, we get 129 varieties of inputs. It shows the model is reliable. In 45% case, we get 171 varieties of inputs and in 55% case, we get 175 varieties of inputs. It means we can get even high elongation steels. Figure 6 shows that in both cases, the range of compositions capable of producing the required elongation is very limited. This implies that the domain of compositions becomes much smaller if an attempt is made to simultaneously satisfy both strength and elongation targets.

Figure 6 confirms the trend identified with the significance chart (Fig. 5) that carbon and manganese have the biggest influence on the absolute value of elongation. A high elongation can correspond to a high carbon and low manganese concentration and vice versa. A high concentration of carbon would lead to a greater fraction of pearlite and hence a larger work hardening rate; the latter in turn would delay plastic instability and hence enhance elongation.

5. Conclusions

The identification of domains of independent parameters which lead to a particular strength or ductility can be systematically defined by using a combination of a neural network model with a genetic algorithm. The computational cost of doing so is large, but a trial and error search of the input space probably should incur a greater expense.

The results of the search process have been found to be physically meaningful in terms of the metallurgy of hot-rolled steels with a mixed microstructure of ferrite and pearlite. It would now be interesting to make a more restrictive model in which a number of property targets are set.

Acknowledgements

The authors are grateful to Professor Hae-Geon Lee for the provision of laboratory facilities at the Graduate Institute of Ferrous Technology, POSTECH, Korea, and to Drs Jae Kon Lee, Young Roc Im and Jung Hyeung Lee of POSCO for their help with the neural network analysis.

6. References

1. B. Denton, D. Gupta and K. Jawahir, Managing increasing product variety at integrated steel mills Interfaces, Vol. 33, 2003, 41-53.
2. D. J. C. MacKay, Bayesian interpolation, Neural Computation, Vol. 4, 1992, 415-447.
3. D. J. C. MacKay, A practical Bayesian framework for backpropagation networks, Neural Computation, Vol. 4, 1992, 448-472.
4. D. J. C. MacKay, Evidence framework applied to classification networks, Neural Computation, Vol. 4, 1992, 720-736.
5. D. J. C. MacKay, Information theory, inference and learning algorithms, Cambridge University Press, U. K., 2003.
6. H. K. D. H. Bhadeshia, Neural networks in materials science, ISIJ International, Vol. 39, 1999, 966-979
7. C. M. Bishop, Neural Networks for Pattern Recognition, Oxford University Press,

1995.

8. S. B. Singh, H. K. D. H. Bhadeshia and D. J. C. MacKay, Neural network analysis of steel plate processing, Ironmaking and Steelmaking, Vol. 25, 1998, 355-365.
9. A. Kermanpur, A. Ebnonnasir and M. Hedayati, A novel analytical-artificial neural network model to improve efficiency of high pressure descaling nozzles in hot strip rolling of steels, Materials Science and Technology, Vol. 23, 2007, DOI: 10.1179/174328407X213099.
10. J.-S. Son and D.-M. Lee and I.-S. Kim and S.-G. Choi, A study on on-line learning neural network for prediction for rolling force in hot-rolling mill, Journal of Materials Processing Technology, Vol. 164-165, 2005, 1612-1617.
11. S. Datta and M. K. Banerjee, Optimizing parameters of supervised learning techniques (ANN) for precise mapping of the input--output relationship in TMCP steels, Scandanavian Journal of Metallurgy, Vol. 33, 2004, 310-315.
12. P. Dutta and D. K. Pratihar, Modeling of TIG welding process using conventional regression analysis and neural network-based approaches, Journal of Materials Processing Technology, Vol. 184, 2007, 56-68.
13. R. Kemp and G. A. Cottrell and H. K. D. H. Bhadeshia and G. R. Odette and T. Yamamoto and H. Kishimoto, Neural-network analysis of irradiation hardening in low-activation steels, Journal of Nuclear Materials, Vol. 348, 2006, 311-328.
14. C. Capdevilla and C. Garcia-Mateo and F. G. Caballero and C. Garcia de Andres, Neural network analysis of the influence of processing on strength and ductility of automotive low carbon sheet steels, Journal of Materials Science, Vol. 38, 2006, 192-201.
15. H. K. D. H. Bhadeshia, D. J. C. MacKay and L.-E. Svensson, Materials Science and Technology, Vol. 11, 1995, 1046-1051.
16. S. B. Singh and H. K. D. H. Bhadeshia, Estimation of bainite plate thickness, Materials Science and Engineering A, Vol. 245, 1998, 72-79.
17. T. Cool and H. K. D. H. Bhadeshia, The yield and ultimate tensile strength of steel welds, Materials Science and Engineering, Vol. 223, 1997, 186-200.
18. N. Chakraborti, Genetic algorithms in materials design and processing, International Materials Reviews, Vol. 49, 2004, 246-260.
19. A. Delorme, Genetic algorithm for optimization of mechanical properties, Technical report, University of Cambridge, June 2003.
20. S. Chatterjee and H. K. D. H. Bhadeshia, δ-TRIP steel, Materials Science and Technology, Vol. 23, 2007, 819-827.
21. S. Chatterjee, Transformations in TRIP-assisted steels: microstructure and

properties, Ph.D. Thesis, University of Cambridge, 2006. http://www.msm.cam.ac.uk/phase-trans/2000/phd.html

22. D. E. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley, 1989.
23. Z. Michalewicz, Genetic Algorithms + Data Structures = Evolution Programs, Springer, 1996.
24. Grefenstette, J. J., 1986, "Optimization of Control Parameters for Genetic Algorithms", IEEE Syst. Man Cybern, Vol. 16, pp.122~128.
25. M. Obitko and P. Slavik, Visualization of Genetic Algorithms in a Learning Environment. Pages 101-106, Bratislava: Comenius University, 1999. Spring Conference on Computer Graphics, SCCG'99.
26. R. Reed and H. K. D. H. Bhadeshia, Reconstructive austenite-ferrite transformation in low alloy steel, Materials Science and Technology, Vol. 8, 1992, 421-435.
27. J. S. Kirkaldy and D. Venugopalan, Prediction of microstructure and hardenability in low alloy steel, Phase Transformations in Ferrous Alloys, eds A R Marder & J I Goldstein, TMS-AIME, Warrendale, Ohio, 1984, 125-148.
28. H. K. D. H. Bhadeshia, Alternatives to the ferrite-pearlite microstructures, Materials Science Forum 284-286, 29, 1998
29. W. D. Callister, Material Science and Engineering: An Introduction 7th edition, Wiley, 2007.

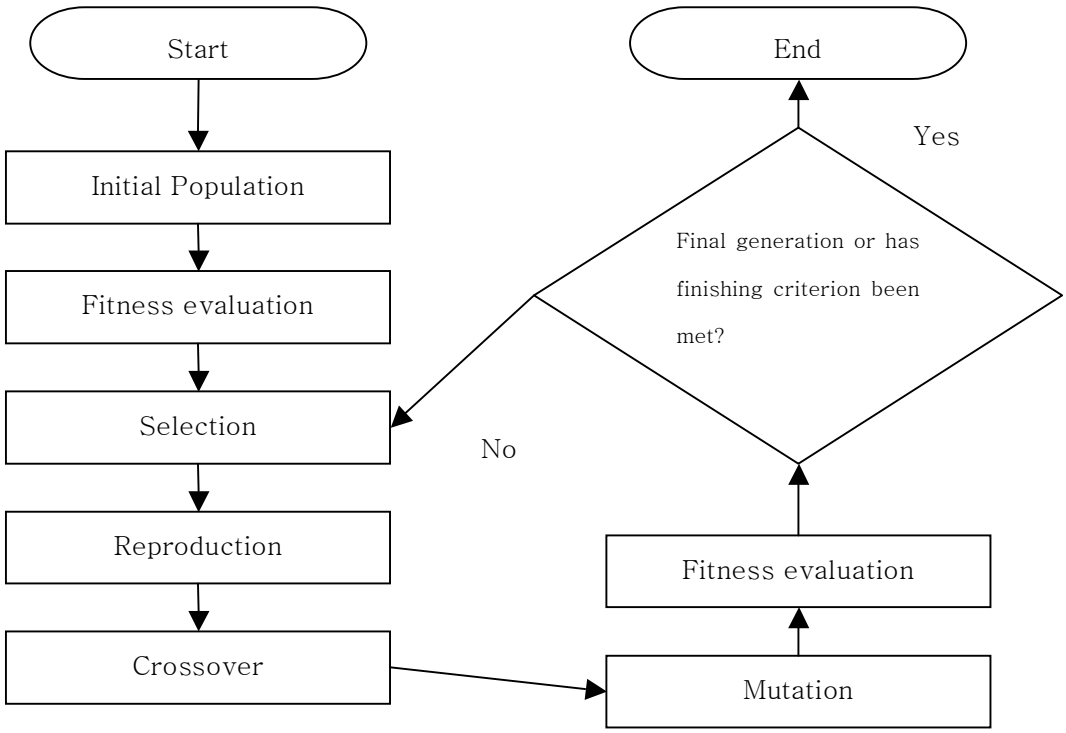


Fig. 1. Illustration of the working of the genetic algorithm

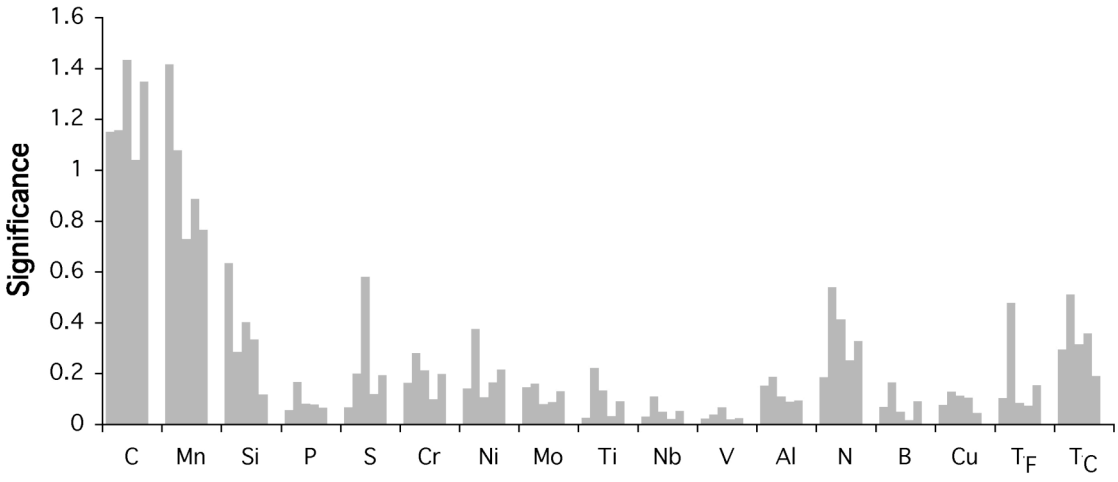
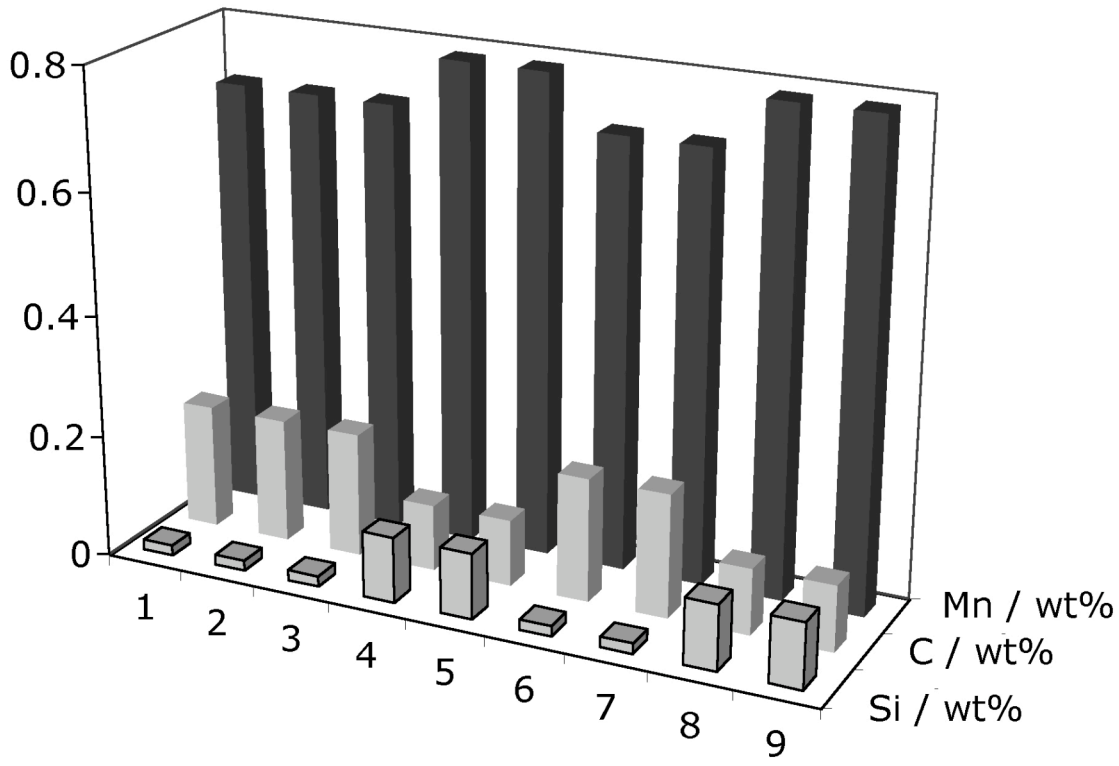


Fig. 2. Significance of inputs in tensile strength model

Materials and Manufacturing Processes 24 (2009) 53-58



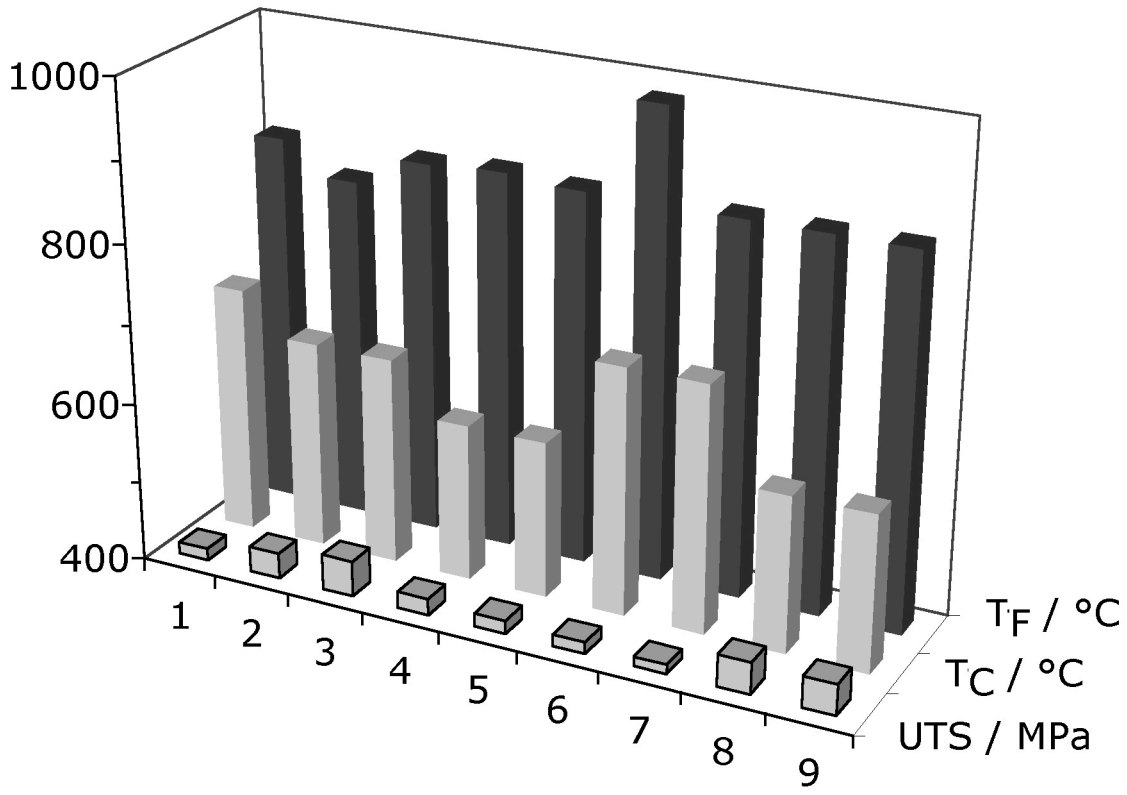


Fig. 3: Combinations of variables, all of which lead to an ultimate tensile strength of approximately 400 MPa.

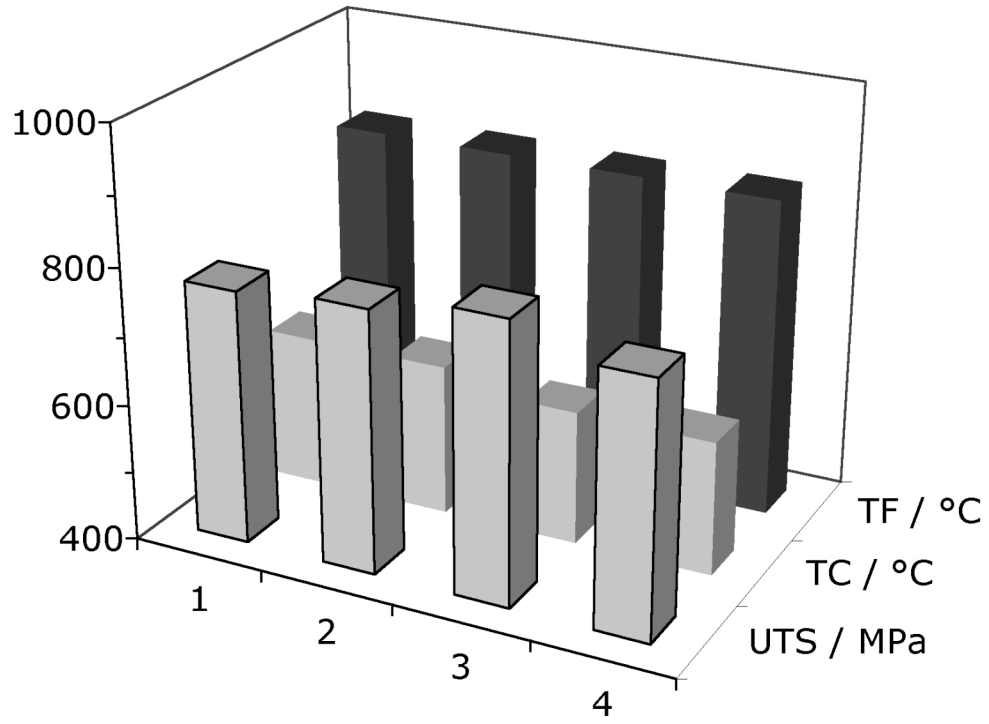


Fig. 4: Combinations of variables which lead to an ultimate tensile strength of 800 MPa.

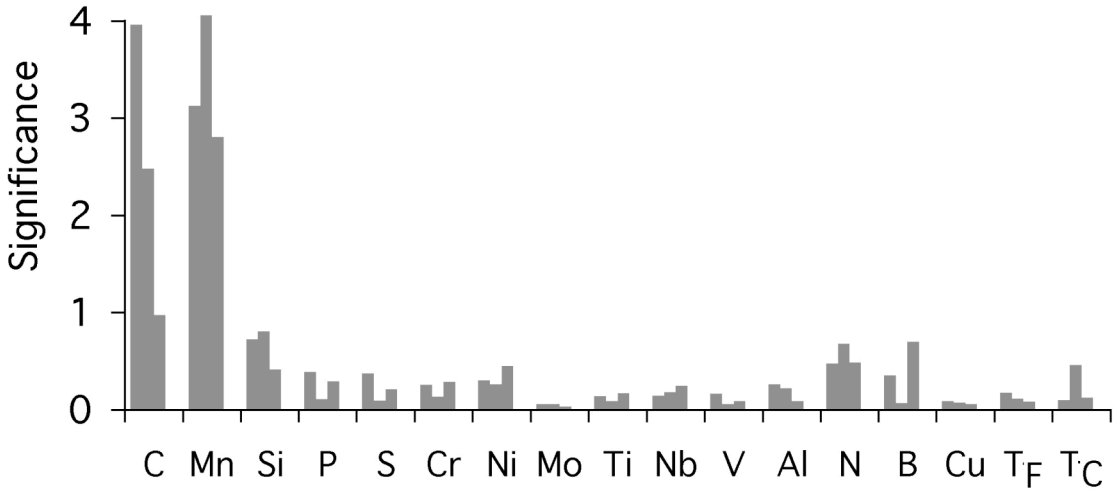


Fig. 5: Perceived significance of input variables on elongation.

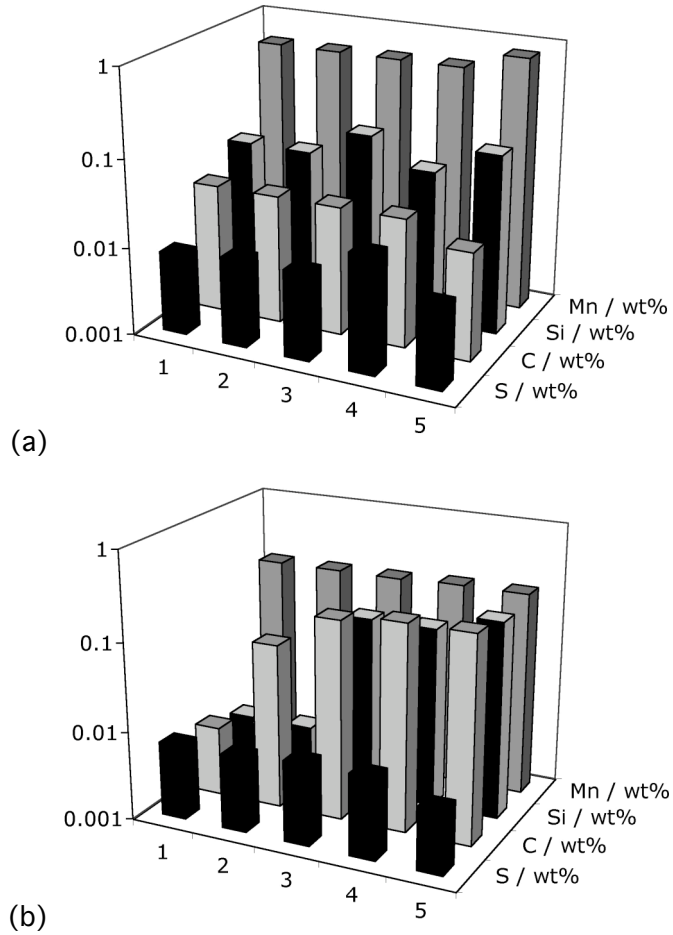


Fig. 6: Combinations of variables which lead to an elongation of (a) 35%, (b) 55%.