A model for predicting the Ms temperatures of steels.

T. Sourmail*, C. Garcia-Mateo**

Department of Materials Science and Metallurgy, University of Cambridge

Pembroke Street, Cambridge CB2 3QZ, U.K.

* *corresponding author, email: ts228@hermes.cam.ac.uk*

** CENIM, National Center for Metallurgical Research

Av. Gregorio del Amo, 8, 28040 MADRID, Spain

—

Abstract

Using neural networks in a bayesian framework, a model has been derived for the M_s temperature of steels over a wide range of compositions. By its design and by use of a more extensive database, this model improves over existing ones, by its accuracy and its ability to avoid wild predictions.

Keywords: martensite , thermodynamics, bayesian neural networks, linear regression

1 Introduction

There is considerable industrial interest in being able to predict reliably the temperature at which austenite transforms to martensite (M_s). For this reason, a significant amount of work has been devoted to obtaining quantitatively accurate models for predicting M_s. This temperature is typically a function of a number of variables which may include stress or magnetic field. From a material point of view, the M_s temperature is essentially controlled by the composition of the steel.

In a recent assessment of the existing models for predicting the M_s temperature of steels as a function of their compositions, the authors showed that, from an

1

applied point of view, the neural network model due to Capdevilla *et al.* performed at least as well as the thermodynamic models proposed by Ghosh and Olson [1–4]. Furthermore, the former is freely available as a standalone computer program.

However, the assessment revealed a number of weaknesses of the neural network model proposed by Capdevilla *et al.* [5] (further referred to as model A), which is the widest in scope available to date. In particular, a large amount of published data had not been used in training this model, which was shown to perform poorly on most of these [6]. The model also had a tendency to make very wild 'predictions', with some values of M_s reaching many thousands of Kelvin on rather ordinary compositions. Finally, we found a significant number of errors in the database used by Capdevilla *et al.* (further referred to as database A), some of them by up to 273 K as a result or incorrect conversions.

In the present work, a new model is created for the M_s temperature of steels as a function of composition, after verifying that the austenitisation temperature can reasonably be neglected in most cases. We then validate it against unseen data and compare its performance to that of model A.

2 Method

Neural network modelling is an empirical modelling method in which a very flexible function is fitted to a set of data by adjusting the parameters of the network, also known as the weights.

The neural network method used in the present investigation has been previously reviewed in the literature (details can be found in [7–10]) and only its most important features are presented.

2.1 Adaptative functions

Neural networks, as opposed to traditional linear or polynomial regression methods, do not impose a shape of function on the data. The structure of a typical feedforward network as used in the present work is illustrated in figure 1. Each hidden unit calculates a weighted sum of the inputs and return its hyperbolic tangent. The output of the hidden units are then linearly combined by the output neuron. The

function corresponding to the 4 hidden-units network shown in figure 1 is:

$$
\begin{aligned}
y \;=\;\; & \omega_1 \tanh(w_1 x + h_1) + \omega_2 \tanh(w_2 x + h_2) \\
& + \omega_3 \tanh(w_3 x + h_3) + \omega_4 \tanh(w_4 x + h_4) + \theta
\end{aligned}
\tag{1}
$$

where the w, ω and h are the parameters to adjust, often referred to as weights and biases.

As illustrated in figure 1, simply varying the weights of such a network allows vastly different functions to be represented.
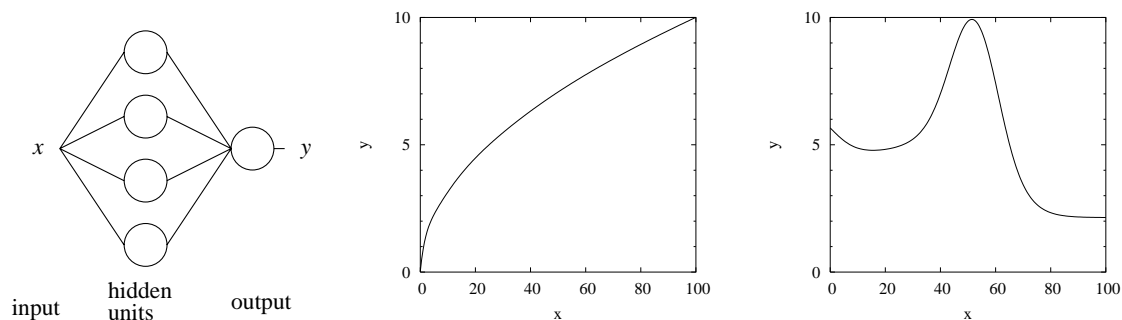

Figure 1: The structure of a feedforward neural network with one input, 4 hidden-units and one output. Two networks with the same structure (4 hidden units) but different weights can represent totally different functions.

2.2 The bayesian framework

A neural network is traditionally trained by optimising its parameters with regard to a given error function. This results in an optimum set of weights which are in turn used to make predictions.

In a bayesian approach however, a probability distribution of weight values is fitted to the data [8, 9]. Where data are sparse, this distribution will be wide, indicating that a number of solutions have similar probabilities. If, on the contrary, there are sufficient data, the probability distribution for the network parameters will be narrow, indicating that one solution is significantly more probable than others. This uncertainty can be translated into an 'error-bar' on predictions, which indicates the uncertainty of fitting where the calculation is made.

This is illustrated in figure 2. The assessment undertaken by the present authors [6] has illustrated how powerful the technique is in limiting the danger of 'wild' predictions.
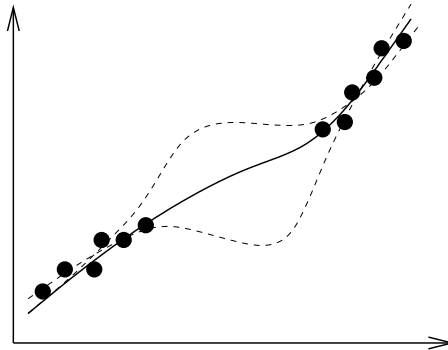

Figure 2: Illustration of the possibilities offered by Bayesian neural networks: the prediction can be accompanied by an error bar related to the uncertainty of fitting. When data are sparse, the uncertainty of fitting is larger than in region with sufficient data.

For further details on the method, we point to the review by Mackay [11].

3   Database

Data were obtained from a variety of sources. The database used by Capdevilla *et al.* [5] was kindly provided by this author. It is based on data published in references [12–16]. During our assessment of existing models for M_s predictions [6], it became apparent that some mistakes were present in this database. It was therefore decided to check all data against the original references. This resulted in a significant number of corrections, sometimes by as much as 273 K when unit conversion has obviously not been done. Additional data were also gathered from the literature [1, 17–31]. This resulted in a database containing about 1200 entries and covering a wide variety of compositions as illustrated in table 1.

Element	Min(wt%)	Max(wt%)	Element	Min(wt%)	Max(wt%)
C	0.0	2.25	Co	0.0	30.0
Mn	0.0	10.24	Al	0.0	3.01
Si	0.0	3.8	W	0.0	18.59
Cr	0.0	17.98	Cu	0.0.	3.04
Ni	0.0	31.54	Nb	0.0	1.98
Mo	0.0	8.0	Ti	0.0	2.52
V	0.0	4.55	B	0.0	0.06
			N	0.0	2.65

Table 1: Minima and maxima for each input variable included in the database.

4 Choice of inputs and output

4.1 Inclusion of strong carbonitride formers

In a number of previous attempts, it has generally been assumed that the austenitising temperature (further denoted T_γ) has only a small effect on the M_s temperature. Experiments have shown that most variations in M_s caused by changes in T_γ should be contained within ±25K [32]. Although this is possibly true for the compositions then investigated, it may not hold for steels with additions of Ti, Nb or V, in which one expects to find carbides or nitrides whose quantity depends on T_γ.

If it is the case that the constitution of such alloys changes significantly over the range of typical austenitisation temperatures, strong variations of M_s should be expected as this temperature is changed.

To verify this, we first calculated the austenite composition of a Fe-0.3C-0.6Si-1.5Mn-0.2Ti (wt%) as a function of temperature. This was done using MTDATA [33] and the SGTE SSOL and SSUB databases [34], allowing austenite and titanium carbide to coexist. The composition of the austenite in equilibrium with TiC was then fed into a computer program that calculates the M_s temperature as a function of composition, following the method of Ghosh and Olson [1].

As illustrated in figure 3, it would be erroneous, when using thermodynamic models, to use the bulk composition to estimate the M_s temperature. However, it is fair to say that variations of T_γ have little impact on the expected M_s temperature once the presence of TiC accounted for. Therefore, it is reasonable not to include
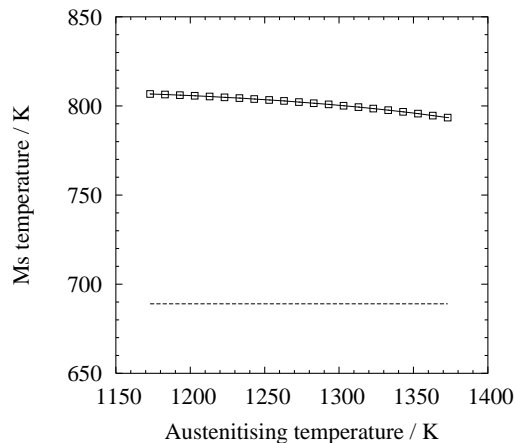
T_γ in a fully empirical model.

Figure 3: The M_s temperature as predicted for Fe-0.3C-0.6Si-1.5Mn-0.2Ti (wt%) using the model of Ghosh and Olson [1]. The dotted line represents the predictions if the bulk composition is used as an input (therefore neglecting the presence of TiC), the plain line represents the M_s temperatures calculated from the composition of the austenite in equilibrium with TiC at the given austenitising temperature.

4.2   A bounded output

As emphasised by Mackay [10], it is important to ensure any knowledge about the system is somehow present in the database, or in the network structure.

The assessment recently published by the present authors [6] illustrated the fact that the M_s temperature should be bounded between 0 and 1000 K. While this was naturally present in the thermodynamic approach of Ghosh and Olson [1–4], existing neural network models are not necessarily bounded [5, 35], although as shown by Yescas *et al.* [36], it is possible to formulate the output in such a way that it has lower and upper limits. In the case of model A, this lead to wild predictions of plus or minus thousands of Kelvin on unseen data.

One way to incorporate this knowledge is to train the model using a function of the target which is naturally bounded in the desired interval [8–10, 36].

The present network was trained using $y = \ln\left(-\ln\left(M_s/1000\right)\right)$, and therefore $M_s = 1000 * \exp\left(-\exp\left(y\right)\right)$ which is bounded between 0 and 1000.

4.3 Training

In a first instance, 124 sets were randomly selected from the database to serve as a test. None of these sets were used in training the present network (while model A is likely to have been trained on a number of these lines, since half of the database is identical to that used to train that model).

The remaining data were then divided in two sets, also randomly selected. The first, one, containing 80% of the lines, was used to train a number of models, while the second, containing the rest of the database, was used to validate the training and select an optimum committee of models. As mentioned earlier, this procedure has been described numerous times in the literature (for example, [7]). In the present study, a commercial package [37] was used which implements the algorithm written by Mackay [8].

5 Results

The performance of the network was assessed on the 124 sets of data unseen during training. Predictions were also obtained for this set of data using model A. As noted earlier, while it is likely that the latter will have seen some of these data during training, the present model will not have seen any of these lines. Table 2 gives some examples of compositions found in this testing set.

Figure 4 compares the performance of both models on this dataset. As in our
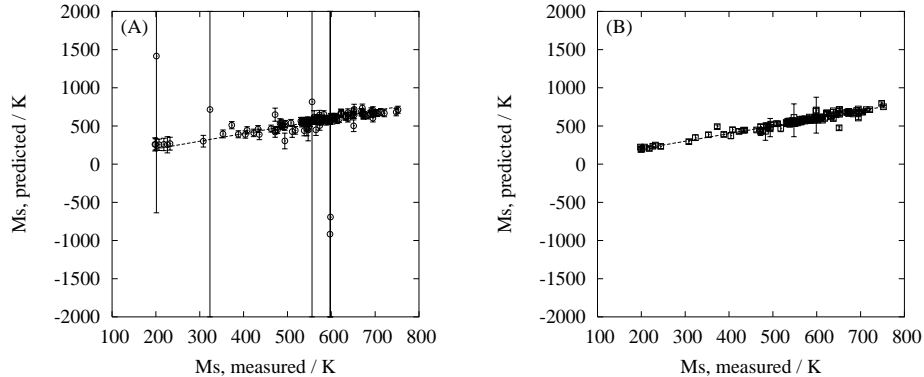

Figure 4: Comparison between the model proposed by Capdevilla *et al.* (A) and the present model (B) on a test dataset containing a variety of compositions.

previous assessment of existing models [6], we propose to compare models using the average of the absolute value of the error between target and prediction (denoted $\overline{\varepsilon}$) and the associated standard deviation (σ_{err}). These gave $\overline{\overline{\varepsilon}}= 94$ ($\sigma_{err} = 334$) using the model by Capdevilla *et al.* and $\overline{\overline{\varepsilon}}= 22$ ($\sigma_{err} = 25$) for the present model.

To take into account the 'warning' given by the large error bars accompanying the wild predictions made by model A, these values were recalculated only for results accompanied by uncertainties of fitting less than 100 K. This eliminates the wild predictions made by the model A (as visible in figure 4).

C	Mn	Si	Cr	Ni	Mo	V	Co	Al	W	Cu	Nb	Ti	B	N
0.22	1.1	0.21	0.6	0.18	0.08	0	0	0	0.3	0	0	0	0	0
0.58	0.08	0.89	1.27	0.06	0.02	0.11	0	0	0	0.14	0	0	0	0
0.72	0.27	0.39	4.09	0	0	1.25	0	0	18.59	0	0	0	0	0
0	0.03	0.08	0	21.66	0	0	3.75	0	0	0	0	0	0	0
0.2	0.64	0.08	2.12	0.76	0.83	0.32	0.01	0	0.63	0	0.02	0	0	0.01
0.24	0	0	1.4	4.98	1.52	0	16.06	0	0	0	0	0	0	0

Table 2: Some examples of compositions found in the randomly selected test set. This set was not used in any part of the training of the new model. All compositions in wt%.

The procedure somewhat reflects the fact that a user should discard such values because of the amplitude of the accompanying error bars. In this case, values of $\overline{\overline{\varepsilon}}= 32$ ($\sigma_{err} = 32$) and $\overline{\overline{\varepsilon}}= 20$ ($\sigma_{err} = 24$) were obtained for model A and the present model respectively, which indicates significantly better predicting performance from the new model, in spite of the fact that some of the test data had been seen by model A during training.

6   Conclusions

Using a large amount of published data, a neural network model has been trained to predict the M_s temperature of steels of a wide range of compositions. By using of a carefully selected function of M_s rather than M_s as the target, it was possible to put bounds on the output, therefore eliminating the risk of wild predictions such as those generated in a previous neural network model. The new model was shown to perform significantly better than the latter.

The bayesian framework means that not only the knowledge present in the database is reflected in the model, but also the absence of it, as the model will produce large error bars for predictions where data were sparse during training.

7 Availability

This neural network model can be used on the wold-wide-web (www-map-online. msm.cam.ac.uk).
The database is also distributed on the internet (www.msm.cam.ac.uk/map).

8 Acknowledgements

References

[1] G. Ghosh, G. B. Olson, Acta Mat. 42 (1994) 3361–3370.

[2] G. Ghosh, G. B. Olson, Acta Mat. 42 (1994) 3371–3379.

[3] G. Ghosh, G. B. Olson, J. Phase Eq. 22 (3) (2001) 199–207.

[4] G. Ghosh, G. B. Olson, Acta Mat. 50 (2002) 2655–2675.

[5] C. Capdevilla, F. G. caballero, C. G. de Andrés, I.S.I.J. 42 (2002) 894–902.

[6] T. Sourmail, C. Garcia-Mateo, unpublished .

[7] H. K. D. H. Bhadeshia, ISIJ Int. 39 (1999) 966–979.

[8] D. J. C. Mackay, Neural Computation 4 (1992) 448–472.

[9] D. J. C. Mackay, Neural Computation 4 (1992) 698–714.

[10] D. J. C. Mackay, Bayesian non-linear modelling with neural networks, http://www.inference.phy.cam.ac.uk/mackay/cpi_short.pdf (1995).

[11] D. J. C. Mackay, Network: Comput. Neural Syst. 6 (1995) 469–505.

[12] M. Atkins, Atlas of continuous cooling transformation diagrams for engineering steels, Tech. rep., British Steel Corporation.

[13] M. Economopoulos, N. Lambert, L. Habraken, Diagrammes de transformation des aciers fabriqués dans le benelux, Tech. rep., Centre National de Recherches Métallurgiques (1967).

[14] Atlas of isothermal transformation diagrams of b.s. en steels. special report no 40, Tech. rep., The British Iron and Steel research association (1949).

[15] Atlas of isothermal transformation diagrams of b.s. en steels.(2nded) special report no 56, Tech. rep., The Iron and Steel Institute (1956).

[16] Atlas of isothermal transformation and cooling transformation diagrams, Tech. rep., American Society for Metals (1977).

[17] A. B. Greninger, Trans. ASM 30 (1942) 1–26.

[18] T. G. Digges, Trans. ASM 28 (1940) 575–600.

[19] T. Bell, W. S. Owen, JISI 205 (1967) 1777–1786.

[20] K. Ishida, T. Nishizawa, Trans. JIM 15 (1974) 218–224.

[21] M. Oka, H. Okamoto, Metall. Trans. A 19 (1988) 447–452.

[22] J. S. Pascover, S. V. Radcliffe, Trans. AIME 242 (1968) 673–682.

[23] R. B. G. Yeo, Trans AIME 227 (1963) 884–890.

[24] A. S. Sastri, D. R. F. West, JISI 203 (1965) 138–145.

[25] U. R. Lenel, B. R. Knott, Metal. Trans. A 18 (1987) 767–775.

[26] W. Steven, A. G. Haynes, JISI 183 (1956) 349–359.

[27] R. H. Goodenow, R. F. Heheman, Trans. AIME 233 (1965) 1777–1786.

[28] R. A. Grange, H. M. Stewart, Trans. AIME 167 (1945) 467–494.

[29] M. M. Rao, P. G. Winchell, Trans. AIME 239 (1967) 956–960.

[30] P. Payson, C. H. Savage, Trans. ASM 33 (1944) 261–281.

[31] E. S. Rowland, S. R. Lyle, Trans. ASM 37 (1946) 27–47.

[32] C. Y. Kung, J. J. Rayment, Metall. Trans. A 13 (1982) 328–331.

[33] MT-DATA, National Physical Laboratory, Teddington, Middlesex, U.K. (1989).

[34] Scientific Group Thermodata Europe, www.sgte.org (1983).

[35] W. G. Vermeulen, P. F. Morris, A. P. de Weijer, S. van der Zwaag, Ironmaking and Steelmaking 23 (1996) 433–437.

[36] M. A. Yescas-Gonzales, H. K. D. H. Bhadeshia, Mater. Sci. Eng. A 333 (2002) 60–66.

[37] Model Manager, Neuromat Ltd, www.neuromat.com (2003).