Prediction of the measured temperature after the last finishing stand using artificial neural networks

Willem Vermeulen, Andre Bodin and Sybrand van der Zwaag

In this report the development of an artificial neural network, capable of predicting the temperature after the last finishing stand of a hot strip mill for a certain class of steels, is described. Three neural networks with different numbers of hidden nodes (3, 5 and 7) were trained. The relative standard deviation in finish temperature as predicted by the best performing neural network model (7 hidden nodes) was just over 25% smaller than that of the linear Hoogovens model. This improved accuracy can be explained by the incorrect assumption in the Hoogovens model of linear dependence of the finishing temperature on some input parameters.
With the trained neural network, the influence of the various input parameters on the finishing temperature could be examined. The dependencies predicted by the neural network can be approximated by a linear fit and are a factor 2 lower for all input parameters.
It is conceivable that operation of the mill using an artificial neural network for the prediction of the finishing temperature would have resulted in smaller operational fluctuations.

Vorausberechnung des Temperaturmeßwertes hinter dem letzten Fertiggerüst mit neuronalen Netzen. In dieser Arbeit wird die Entwicklung eines künstlichen neuronalen Netzes beschrieben, mit dem sich die Temperatur hinter dem letzten Fertiggerüst einer Warmbandstraße für eine Reihe von Stahlsorten vorausberechnen läßt. Drei verschiedene Netzkonfigurationen mit unterschiedlicher Anzahl von Knoten in der mittleren Ebene (3, 5 und 7) wurden trainiert. Die Standardabweichung für die genaueste Temperaturabschätzung, die mit dem neuronalen Netz mit 7 Knoten in der verdeckten Ebene zustande kam, lag um 25% niedriger als der mit dem linearen Hoogovens-Modell berechnete Wert. Diese bessere Genauigkeit läßt sich damit erklären, daß das Hoogovens-Modell von der falschen Annahme ausgeht, daß die Abhängigkeit zwischen der Fertigwalztemperatur und einigen Eingangsparametern linear sei.
Mit dem trainierten neuronalen Netz läßt sich der Einfluß der verschiedenen Eingangsparameter auf die Fertigwalztemperatur untersuchen. Die mit dem Netz vorausberechneten Abhängigkeiten können angepaßt werden und liegen für alle Eingangsparameter um den Faktor 2 niedriger. Es ist vorstellbar, daß der Einsatz künstlicher neuronaler Netze zur Vorausberechnung der Fertigwalztemperatur zu einem gleichmäßigeren Walzwerksbetrieb führen könnte.

Hot rolling of steel is a very complex process. In order to produce a strip of metal with uniform mechanical properties and hence a uniform microstructure both hot rolling and the cooling conditions on the run-out table must be carefully controlled. This control is only possible if the temperature is known at all stages of the process. Of particular interest is the temperature after the finishing train of the mill as this determines the settings of the cooling conditions on the run-out table and the temperature control from reheating furnace to finishing mill. It also holds information on whether the strip was rolled in the austenitic state or not. In this report a neural network is developed to predict the measured finishing temperature as a function of the process parameters in the rolling mill. The predictions of the neural network are compared with those of an existing linearised model involving the same parameters currently in use at the hot strip mill of Hoogovens, IJmuiden (from now on referred to as the linear model).

First, the complete rolling and cooling process will be discussed briefly, using hot strip mill no. 2 at Hoogovens, **figure 1**, as reference mill.

Following continuous casting the slabs are allowed to cool to room temperature and are then fed into the reheating furnace (or occasionally hot charged). To reduce the work that is necessary to deform the steel slab and to dissolve precipitated elements such as niobium and aluminium, the slabs are (re)heated to a temperature of 1250 °C. An optimum has to be found between (micro)alloying elements going into solution, excessive oxide layer formation, reheating time, and energy costs. The oxide layer formed in the furnace is removed before the actual rolling by high-pressure water jets. The next process is roughing, where a transfer bar is produced that can be handled by the finishing train. The temperature of the slab after roughing is approximately 1050 °C. The next step is the deformation of the transfer bar to a strip in a coupled finishing train with 7 stands. In the finishing train the final thickness and shape of the strip are determined but the width can be adjusted slightly during roughing. During the complete hot rolling process, the steel must be kept in the austenitic state

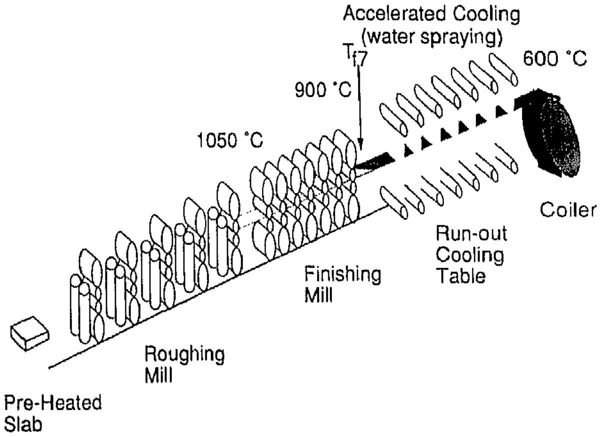

Figure 1. Schematic representation of the hot strip mill no. 2 at Hoogovens IJmuiden

Ir. Willem Vermeulen; Professor Dr. Ir. Sybrand van der Zwaag, Materials Science Laboratory, Delft University of Technology; *Ir. Andre Bodin*, Section Manager R&D, Hoogovens Research and Development, Koninklijke Hoogovens N.V.

and the oxide layer thickness must be kept as low as possible. This limits the temperature domain of the rolling process from 1050 to approximately 890 °C for a low carbon steel. The minimum temperature is determined by the steel composition. Below this minimum temperature, the austenite will transform into a different structure (ferrite, pearlite, bainite or martensite, depending on steel composition, temperature, and cooling rate). After the last finishing stand, the strip is water cooled on the run-out table to a predetermined coiling temperature. The cooling conditions are of vital importance for the final properties (microstructure) of the rolled strip. A very important temperature which is required for calculating the cooling procedure on the run-out table is the strip temperature after the last finishing stand, T_{f7}.

Production costs in a hot strip mill can be reduced if its efficiency is increased, e.g. if the amount of rejected material is reduced. A strip is considered as rejected material if it does not meet the requirements of the customer and thus has to be sold as lower quality or has to be re-melted. This last option implies a tremendous amount of extra materials handling and energy costs. One way of increasing the efficiency of the hot strip mill is to optimise the cooling on the run-out table by a more accurate control of the strip temperature after the last finishing stand. A better control of the coiling temperature will also increase efficiency, although this control is already very good [1] at Hoogovens. Other possible improvements can be obtained by better control of the rolling forces [2...4]. Better prediction of the rolling forces may reduce variations in head-end thickness of the hot strip. For a more accurate rolling force prediction, an accurate temperature model is a prerequisite.

This report describes the modelling of the measured finishing temperature (T_{f7}) using an artificial neural network. At present a simplified (linear) model, derived from an off-line physically based mathematical (finite difference) model is used at Hoogovens Steel Stripmill Products to predict this finish temperature. This model is continuously adapted to the changing operational conditions in the mill by an adaptive procedure. The accuracy of this model is good, but it is assumed that further improvements can be reached if the T_{f7} is predicted using a non-linear model. The goal of this research is to design such a model and to verify the assumption. Several neural networks with different architecture were trained and the accuracy of their predictions was compared with that of the linear model. The neural network with the highest accuracy was investigated in more detail; the influences of the input parameters on the predicted T_{f7} were identified and compared to the input parameter dependencies in the linear model.

The Hoogovens model

The linear model for predicting the measured finishing temperature after the last finishing stand is a model that consists of eight linear independent factors and one partially dependent parameter. The linear model in its basic form is given in equation (1):

$$T_{f7} = f(T_{f7base}, V_{f7}, H_{f7}, T_{f1}, \Delta H_{R6}, \text{MIF}, \text{CF}, \text{RSH}) \quad (1)$$

with T_{f7} - re-predicted finishing temperature after the last stand, T_{f7base} - base finish rolling temperature, adapted from strip to strip, V_{f7} - rolling speed of the last stand, H_{f7} - strip thickness after the last finishing stand, T_{f1} - calculated temperature at first finishing stand. No contribution of water-cooling at descaling and interstand-cooling. So, just radiation is accounted for. H_{R6} - transfer bar thickness, MIF - mill intensity factor, CF - cooling factor (water-cooling at descaling and interstand cooling) and RSH - relative strip hardness.

For any statistically based model, the strips have to be divided into groups with similar characteristics. For the linear Hoogovens model, the steel grades are divided into quality groups which in turn are divided into thickness groups. The constants in the model can have different values for different thickness groups. The same applies to the standard rolling strip thickness and the transfer bar thickness. The standard rolling speed and the standard temperature at the first finishing stand are constant within a specific thickness and quality group. The T_{f7base} is adapted from strip to strip and accounts for changes in installation condition.

Data

In total 5763 strips from a specific thickness group and steel grade were used to train and validate the neural network. From this data set 1441 strips (25% of the total number of strips) were selected randomly for the validation set. The remaining 4322 strips for the training set were stored in random order to prevent positional bias.

The input parameters of the neural network were identical to those of the Hoogovens model to allow a fair comparison between the two models. In total eight variables served as input to predict the measured finish temperature after the last finishing stand. These input parameters are: T_{f7base}, V_{f7}, H_{f7}, T_{f1}, H_{R6}, MIF, CF, and RSH.

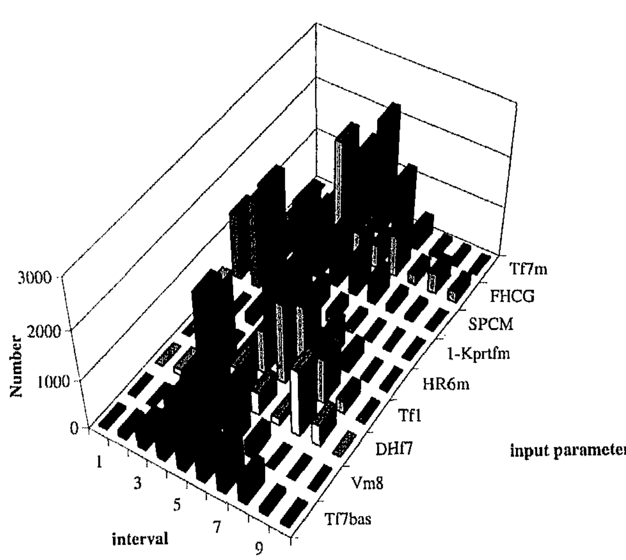

Figure 2. Graphical data distribution of the input and output parameters. The parameter domain is divided in 10 equal intervals and the number of strips lying within a specific interval is identified

The output of the neural network is the temperature of the strip just measured after the last finishing stand (T_{f7}). The distribution of the input and output parameters is given in graphical format in **figure 2.** Each parameter domain is divided into 10 equal intervals and the number of strips lying within a specific interval is identified.

Training of the neural network

The neural network used in this research is a hierarchical feed forward neural network and is trained with back-propagation of error, the momentum version of this training rule. A detailed description of this training rule is given by Hecht-Nielsen [5]. In this section the basic features of the neural network used and training with backpropagation are explained. More information on the neural network theory in general, can be found in references [5...7].

A diagram of the type of neural network used is shown in **figure 3.** The basic unit in a neural network is its processing element, called a node or a neuron. In a hierarchical neural network these nodes are ordered in layers. The network is called feed-forward, because the nodes process the information in one direction only, from input to output. Each node in a layer is connected, via a weight factor, with each node in the preceding layer; the network is fully connected. The number of nodes in the input layer equals the number of input parameters. In this research the total of input parameters is 8. The number of nodes in the output layer equals the number of output parameters. In this research the number of output parameters equals 1, the finish temperature after the last finishing stand. The optimum number of nodes in the intermediate -hidden- layer depends on the complexity of the problem, and it is up to the researcher to determine this. This intermediate layer is called the hidden layer since both inputs and outputs from this layer come and go from other network layers and hence this layer is not seen when looking from the outside to the network.

Each node computes the scalar product of its input values and their weight factors and passes this value to a sigmoid transfer function, which produces the output signal of the node. To determine the weight factors -the actual modelling- the neural network has to be trained with back-propagation of error, the momentum version [5].

An iteration in the training cycle is outlined in the flowchart of **figure 4.** To start, the weight factors are initialised randomly. Then, the input and output data of one sample are presented to the network, which, with its randomly initialised weight factors, calculates the output of the sample; the predicted value. This predicted value is compared with the actual or target value of the sample. The difference between target and predicted value -the error in the prediction- is a measure for the weight factor correction. This correction takes place in the reverse direction -back propagation of error- so first, the weight factors of the output layer are corrected, and then the weight factors of the nodes in the hidden layer. Once the weight factors have been corrected for all samples, the training cycle is repeated until the differences between calculated and target output values are minimised sufficiently.

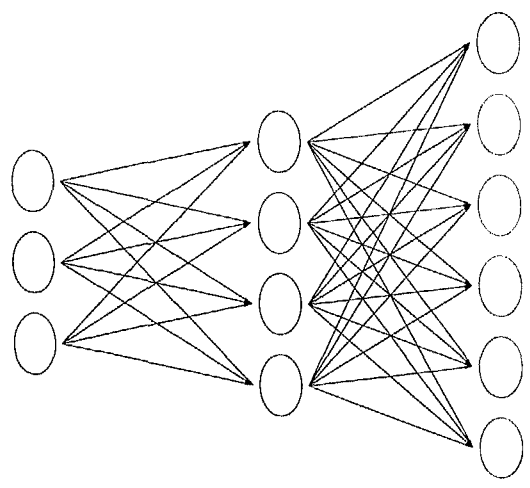

Figure 3. Representation of a feed-forward hierarchical neural network. Data is transferred from left to right along the arrows; the circles represent the nodes or neurones

At large number of training cycles -iterations- the network starts to model not only the functional dependencies between input and output parameters but also the noise in the data set. This is called overtraining. To prevent the network from overtraining the data is split into a relatively large training set and a smaller test or validation set. The weight factors in the network are adjusted using the data in the training set only. In case of overtraining the error for the training set decreases while that for the test set increases with further iteration. At this point training should be stopped. The neural network software package used is designed in such a way that at the onset of overtraining the training is stopped automatically. Once the network is trained, the weight factors are fixed and the neural network may be used to calculate the output for any arbitrary set of input data between the boundaries of the input parameter domain.

In total 3 neural networks were trained, all on the same data, but with different architecture, i.e. with a different number of hidden nodes. A network with 3 hidden nodes (network 1), a network with 5 hidden nodes (network 2), and finally a network with 7 hidden nodes (network 3) were trained. The number of iterations for the training was pre-set at 10000, resulting in a computing time of approximately 12 to 14 hours for each of the three networks.

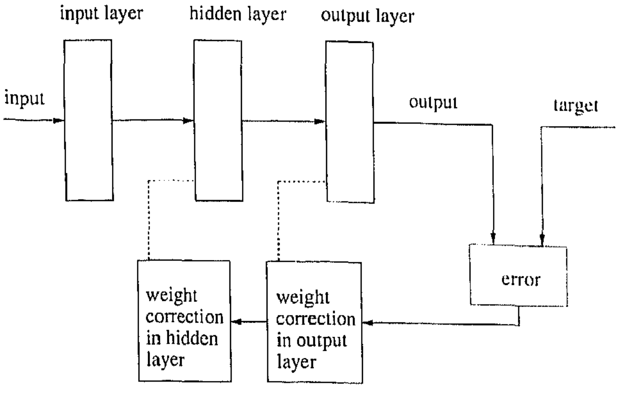

Figure 4. A flowchart of the training cycle in a feed-forward hierarchical neural network

Table 1. The s (°C) of the prediction of the neural network models and the linear model for both the complete and the validation data set

	s_{all}	$s_{validation}$
linear model	5.9	6.0
network 1	4.4	4.6
network 2	4.4	4.5
network 3	4.3	4.4

Training was performed on a IBM compatible PC, 486 DX II, 66 MHz.

The neural networks converged to a minimum after approximately 2000 - 3000 iterations. Overtraining started at this number of iterations, and the weights of the neural network are saved just before the onset of overtraining.

Results

The performance of the neural network is evaluated using the relative standard deviation (s). The s is defined as:

$$s = \sqrt{\frac{\sum_{n=1}^{k} \left(T_{f7m} - T_{f7}\right)^2}{(k-1)}} \qquad (2)$$

with s - relative standard deviation in °C, T_{f7m} - measured finishing temperature in °C, T_{f7} - re-predicted finishing temperature in °C and k - number of samples (-).

The s of the prediction of the linear model and network 1, 2 and 3 networks for the complete data set and the validation set are given in **table 1.**

The finish temperature predicted by network 3 is plotted in **figure 5a** versus the measured finish temperature. A similar plot for the linear model predictions is shown in **figure 5b.** The measured data are equal for both plots. Generally a good agreement between measured and predicted is obtained forboth linear model and network 3. However, the performance of the network 3 is just over 25% better than the linear statistical model.

Discussion

All three neural networks give a better prediction of the T_{f7} than the linear model. From table 1 it is concluded that the linear model is an accurate model with a low s in the prediction (~ 6.0 °C). However, the neural network models can improve on this with an s in prediction of 4.4 °C for network 3, which is an improvement in prediction of just over 25%.

Beside the s in prediction, the performances of all models (linear model and network 1, 2 and 3) were evaluated using the average mean error. The average mean error is defined as the average value for the measured minus the predicted finish temperature. So a positive average mean error, means an average prediction that is lower than the measured temperature. For all models this mean was very close to zero, indicating that none of them exhibit a systematic prediction below or above the actual re-predicted finishing temperature, T_{f7}.

Network 3 (7 hidden nodes) is the best artificial neural network model and will therefore be considered during the rest of this report. The difference between the s and average mean error of network 1, 2 and 3 indicates that the number of hidden nodes does not have a large influence on the accuracy in the prediction. It is considered to be beyond the scope of this report to go into further detail on this aspect of neural networks.

The outliers in the final set of predicted finishing temperatures (those predictions of network 3 with an error in prediction larger than 3 s), were identified in both the training and the validation set. On the assumption of a

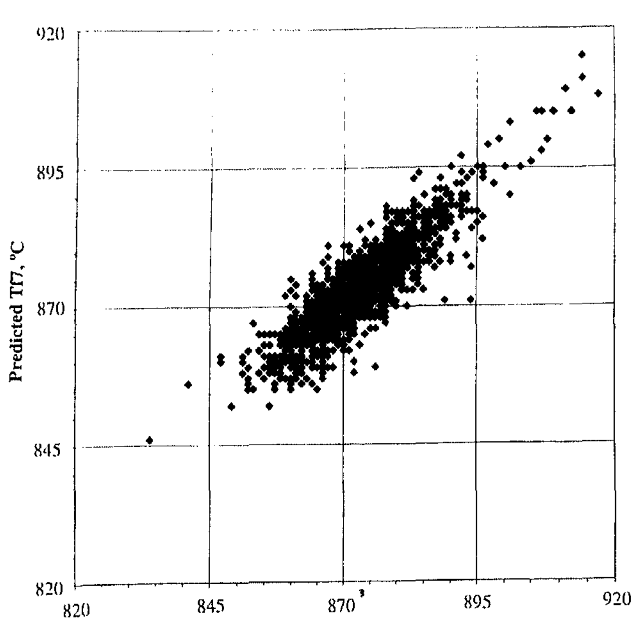

Figure 5a. Finishing temperature predicted by network 3 versus the measured finishing temperature

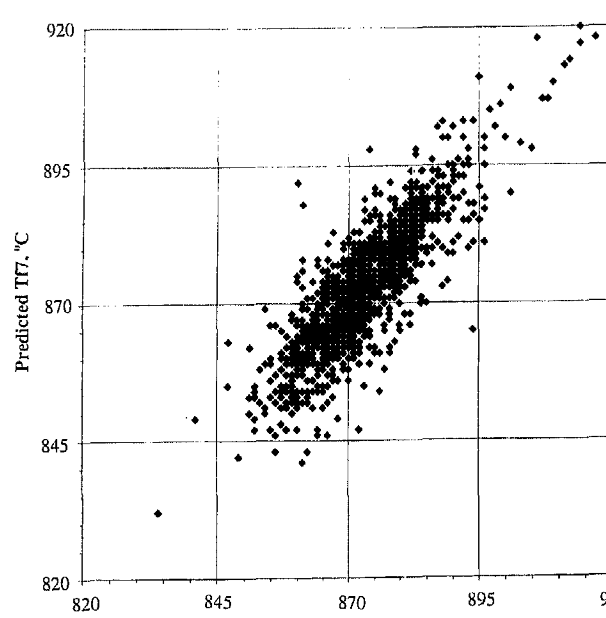

Figure 5b. Finishing temperature predicted by the linear model versus the measured finishing temperature

Table 2. Input parameter of the 4 reference strips

Input parameter	reference 1	reference 2	reference 3	reference 4
T_{f7base}	low	high	medium	medium
V_{f7}	medium	medium	medium	low
H_{f7}	low	medium	medium	high
T_{f1}	high	medium	medium	low
H_{R6}	low	high	low	medium
MIF	high	low	medium	high
CF	low	high	medium	medium
RSH	medium	medium	medium	medium

normal distribution, 99.74% of all data should fall within the limits of 3 s. So, 4 outliers in the validation and 11 in the training set are expected. However, 14 (0.97%) and 30 outliers (0.69%) are identified in, respectively, validation and training set. Almost half of the erroneous predictions can be explained simply by looking at the data, e.g. some strips have a number of input parameters at the edges of the input domain, i.e. very thick or thin strips, strips that are rolled at relatively high speed, strips that have cooled a relatively long time at the roller table before finish rolling etc. No information could be found in the data set to explain the other outlier predictions.

Even for the outliers, the predictions of the neural network were a little better than those of the linear model ($s_{network\,3}$ = 15.1 °C and s_{linear} = 17.6 °C). The predictions of both linear model and network 3 deviate in the same direction, i.e. for a specific strip, both models predict a finishing temperature that is either too low or too high.

The neural network is capable of handling different (non-linear) dependencies in different areas of the input space. Therefore it is not possible to illustrate the dependence of the finish temperature on a particular input parameter in a simple manner. To circumvent this problem, and to illustrate that the neural network is indeed capable of handling different dependencies, 4 typical reference strips were chosen and for these strips the influence of the input parameters on the predicted finish temperature was determined. The input characteristics (in terms of low, medium and high) of these reference strips are given in **table 2.**

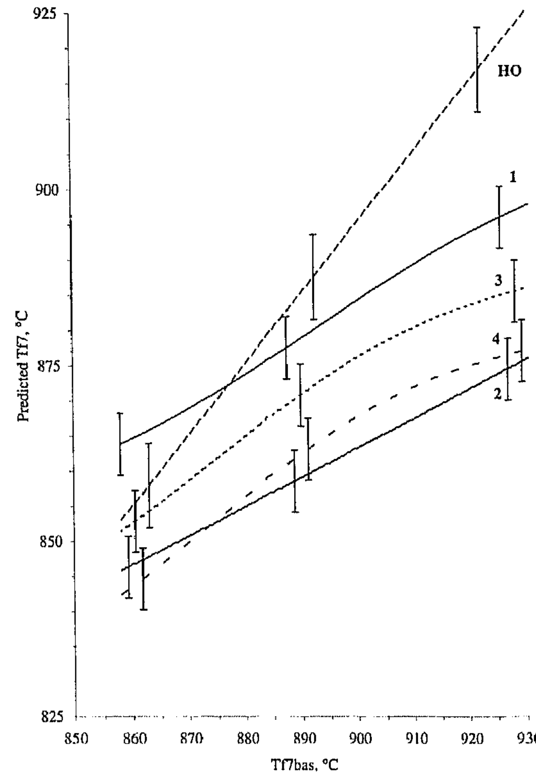

Figure 6. Influence of T_{f7base} on the predicted T_{f7}

In the next section the influence of five of the input parameters on the predicted finishing temperature will be discussed in more detail. These five input parameters are: T_{f7base}, H_{f7}, T_{f1}, CF and RSH. The influence of the remaining three input parameters (V_{f7}, H_{R6} and MIF) is summarised in **table 3.**

For each reference strip, just one input parameter will be varied, keeping the others at their set value. In the linear model, all eight input parameters (T_{f7base}, V_{f7}, H_{f7}, T_{f1}, H_{R6}, MIF, CF and RSH) are approximated as linear. This is not the case for the neural network models. However the dependencies found can be approximated as linear. The resulting linear least squares curve fit equations are summarised in table 3 to allow direct comparison with the linear model. The accuracy of the fit expressed with the R-value is also given.

Table 3. Curve fit equations for the linear approximation of the influence of the input parameters on the T_{f7} predicted by network 3

	reference 1	reference 2	reference 3	reference 4
T_{f7base}	$Y = 442.5 + 0.5 \cdot X$	$Y = 489.3 + 0.4 \cdot X$	$Y = 428.4 + 0.5 \cdot X$	$Y = 417.2 + 0.5 \cdot X$
	$R = 0.9994$	$R = 1.0000$	$R = 0.9910$	$R = 0.9888$
V_{f7}	$Y = 860.0 + 12.3 \cdot X$	$Y = 854.4 + 9.9 \cdot X$	$Y = 859.0 + 11.2 \cdot X$	$Y = 857.5 + 10.1 \cdot X$
	$R = 0.9961$	$R = 0.9998$	$R = 0.9997$	$R = 0.9985$
H_{f7}	$Y = 878.5 + 16.9 \cdot X$	$Y = 868.1 + 29.1 \cdot X$	$Y = 877.3 + 13.7 \cdot X$	$Y = 863.9 + 6.8 \cdot X$
	$R = 0.9194$	$R = 0.9594$	$R = 0.9006$	$R = 0.3491$
T_{f1}	$Y = 869.8 + 0.5 \cdot X$	$Y = 862.2 + 0.5 \cdot X$	$Y = 868.6 + 0.6 \cdot X$	$Y = 869.1 + 0.6 \cdot X$
	$R = 0.9700$	$R = 0.9997$	$R = 0.9969$	$R = 0.9989$
H_{R6}	$Y = 879.3 + 2.4 \cdot X$	$Y = 867.6 + 0.7 \cdot X$	$Y = 876.7 + 1.6 \cdot X$	$Y = 867.0 + 1.7 \cdot X$
	$R = 0.9992$	$R = 0.9977$	$R = 0.9999$	$R = 1.0000$
CF	$Y = 892.1 - 10.7 \cdot X$	$Y = 903.1 - 13.8 \cdot X$	$Y = 901.0 - 14.2 \cdot X$	$Y = 891.4 - 13.5 \cdot X$
	$R = 0.9736$	$R = 0.9867$	$R = 0.9920$	$R = 0.9945$
MIF	$Y = 883.1 - 17.8 \cdot X$	$Y = 881.3 - 16.2 \cdot X$	$Y = 882.5 - 13.2 \cdot X$	$Y = 873.9 - 3.2 \cdot X$
	$R = 0.9993$	$R = 0.9934$	$R = 0.9935$	$R = 0.8180$
RSH	$Y = 876.1 + 24.9 \cdot X$	$Y = 867.1 - 8.0 \cdot X$	$Y = 876.6 + 37.3 \cdot X$	$Y = 866.9 + 38.8 \cdot X$
	$R = 0.8898$	$R = 0.5489$	$R = 0.7497$	R = 0.54459

Figures 6 to 10 show the influence of the five input parameters (T_{f7base}, H_{f7}, T_{f1}, CF and RSH) on the finishing temperature T_{f7} predicted by network 3, for the 4 reference strips. Also included in these figures is the input parameter dependence used in the linear model.

Figure 6 shows the influence of T_{f7base}. The curves are fairly linear for all reference strips. Those of strip 3 and 4 are slightly non-linear, however the curvature of these curves is statistically insignificant. The slopes of the curves are more or less equal, with an average value of 0.5, half the value of the linear model.

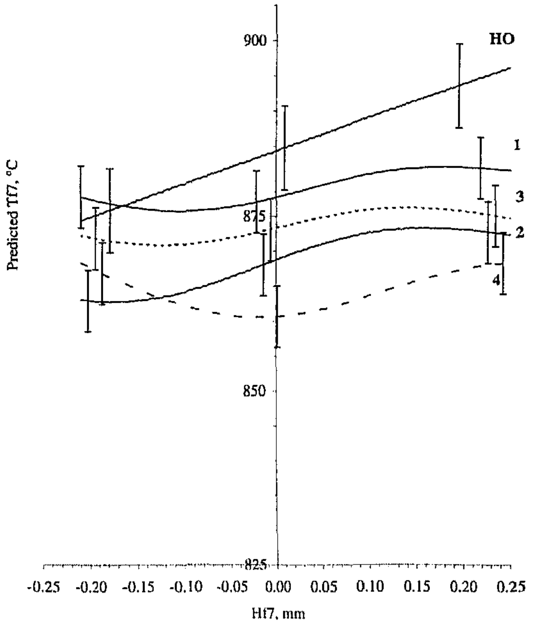

Figure 7. Influence of H_{f7} on the predicted T_{f7}

A higher base temperature will result in a higher strip temperature after the last finishing stand. Both linear and neural network model predictions are in agreement with this statement.

Figure 7 shows the influence of the strip thickness after the last finishing stand on the predicted T_{f7}. In this figure, also some experimental T_{f7} data are plotted for steel strips with similar characteristics of strip 1. Clearly, the dependence of T_{f7} on H_{f7} is sometimes non-linear. The curves of reference strips 1, 2 and 3 have the same shape and can be approximated as linear with reasonably accuracy. This is

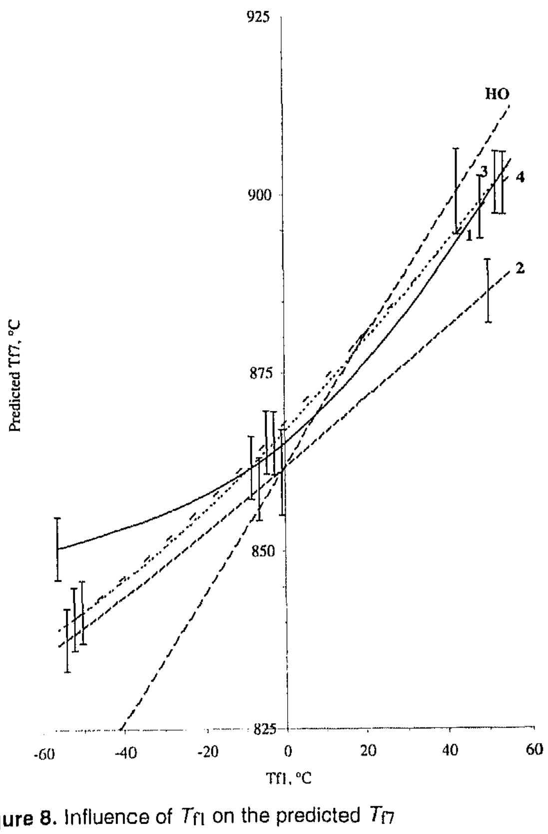

Figure 8. Influence of T_{f1} on the predicted T_{f7}

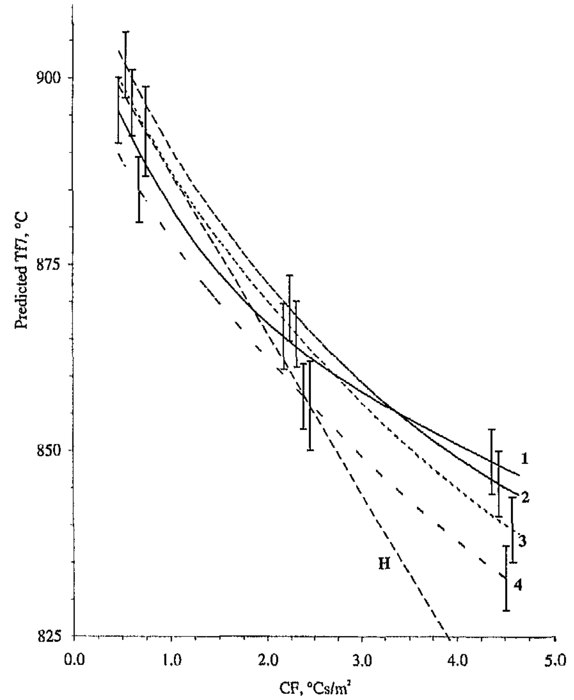

Figure 9. Influence of CF on the predicted T_{f7}

not the case for reference strip 4. However, the measured T_{f7} is almost independent of the strip thickness after the last finishing stand for the range of input data. The influence of the H_{f7} predicted by the linear model is somewhat higher. A useful comparison of the linear approximations is not possible, because influences predicted by network 3 were non-linear.

It is expected that the predicted T_{f7} will increase with increasing H_{f7}. Therefore, the prediction for reference strip 4 can not be correct. A possible explanation for the decrease in T_{f7} with increasing H_{f7} could be the correlation between H_{f7}, T_{f1}, and CF.

The measured T_{f7} for strips with similar characteristics as strip 1 were also plotted to validate both linear model

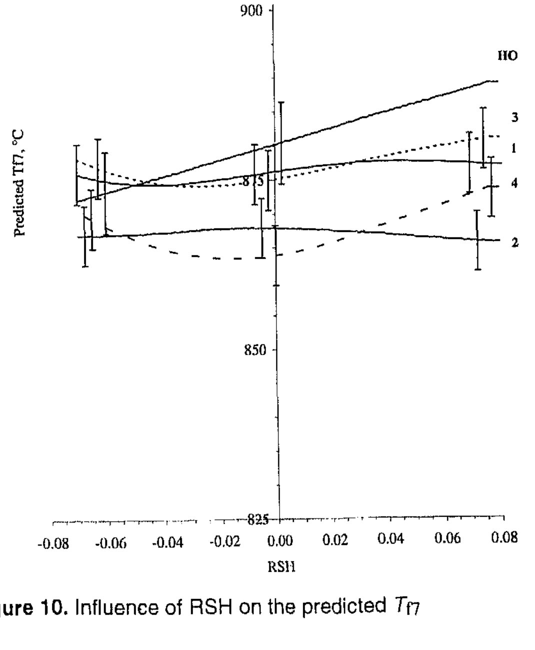

Figure 10. Influence of RSH on the predicted T_{f7}

and network 3. The extensive database however did not contain strips with exactly the same characteristics except for H_{f7}. Therefore, no general trend could be deduced from the plotted points. It is, on the basis of this figure, impossible to compare the predictions of both models with the measured data and to conclude which model is best.

The dependence of finishing temperature on the calculated temperature at the first finishing stand (T_{f1}), can be approximated as linear with high accuracy (figure 8, table 3). The difference between the slopes for the different reference strips is minimal, with an average value of 0.5. Again this is approximately a factor 2 lower than that of the linear model.

A higher temperature at the entrance of the finishing train (T_{f1}) should result in a higher temperature at the end (T_{f7}). Both the linear model and network 3 comply with this expectation.

The cooling factor (CF) has a non-linear influence for all reference strips, figure 9. The shape of the curves for strip 2, 3 and 4 is the same; the non-linearity is higher for strip 1. All influences can be approximated as linear with reasonably high accuracy with approximately the same slope. The average value of -13 is almost a factor 2 lower than that of the linear model.

At CF values lower than 2.5, both models predict the temperature at equal levels. At higher values of CF, the influence predicted by network 3 levels off and consequently the influence predicted by the linear model will be lower. Both models agree with the expectation that a higher temperature loss will result in a lower temperature after the last stand, T_{f7}.

Figure 10 shows the influence of the relative strip hardness. For all reference strips network 3 predicted a non-linear dependence of T_{f7}. Especially the R-values for the linear approximation for reference strips 2 and 4 is very low (0.54). The linear coefficient for reference strip 2 is negative while that of the others is positive. However, the variations of T_{f7} on RSH were all within the statistical error. The RSH did not have a very strong influence on the finish temperature T_{f7}. Clearly, a comparison with the linear coefficient of the linear model does not make any sense.

Conclusions

The relative standard deviation in the prediction of the temperature after the last finishing stand in a hot strip mill, T_{f7} by a neural network with 7 hidden nodes is just over 25% better than that in the prediction of the linear model. Even for its outliers, the predictions of the neural network model are better than those of the linear model for the same strips.

The linear approximations in the linear model seem acceptable for most input parameters. However, the neural network indicated that the dependence of T_{f7} on H_{f7}, CF, and RSH might in fact be non-linear. The influence of the other 5 input parameters ($T_{f7\mathrm{base}}$, V_{f7}, T_{f1}, H_{R6}, MIF) on T_{f7} can be approximated as linear with high accuracy. However, even for these input parameters, a small deviation from the linear approximation is accounted for by the neural network. The difference in the accuracy (s and average mean error) in prediction of the finishing temperature between the neural network model and the linear model may be explained by the incorrect assumption of linearity between finishing temperature and H_{f7}, CF and RSH respectively.

On average, the linear approximation of the functional dependencies found by the neural network is a factor 2 lower for all input parameters.

The overall conclusion is that neural networks are suitable for the prediction of the temperature after the last finishing stand and result in a smaller standard deviation of the recalculated finishing temperature than the present linear model.

Application of artificial neural networks for a set-up calculation can be done by applying a suitable adaptation procedure. In view of the fact that the standard deviation in the recalculated finishing temperature can be reduced significantly using artificial neural networks, smaller operational fluctuations may be possible.

Acknowledgements

The authors wish to thank Koninklijke Hoogovens N.V. and Hoogovens Steel Stripmill Products for supplying the necessary data and their cooperation in publishing this report. Thanks are also expressed to Ton Snijders and Jochem Groot of Hoogovens Research and Development for the clarifying discussions.

(A 01 157; received: 17. February 1996;
in completed form: 24. April 1996)

References

[1] *Heesen, G.; Hollander, F.; La Maire, M.; Snijders, T.*: Modernization of Hoogovens' Run-out Table Cooling to meet Tomorrow's Quality Requirements, METEC Congr. 1994, 6th Intern. Rolling Conf., June 20-22, 1994, Düsseldorf, Vol. 2, p. 184/91.

[2] *Martinetz, T.; Protzel, P.; Gramckow, O.; Sörgel, G.*: Proc. 3rd Annual SNN Symp. on Neural Networks, 14-15 September 1995, Nijmegen, Netherlands, p. 280/86.

[3] *Portmann, N.; Linhoff, D.; Sörgel, G.; Otto, G.*: Iron Steel Eng. (1995), p. 33/36.

[4] *Lindgoff, D.; Sörgel, G.; Gramckow, O.; Klode, K.*: Stahl u. Eisen 114 (1994) No. 4, p. 49/53.

[5] *Hecht-Nielsen, R.*: Neurocomputing, Addison-Wesley Publishing Company, Reading, Massachusetts, 1990 .

[6] *Aleksander, I.; Morton, H.*: Neural Computing, An introduction to, Chapmann & Hall, London, 1991.

[7] *Zupan, J.; Gasteiger, J.*: Neural Networks for Chemists, An Introduction, VCH Verlagsgesellschaft, Weinheim, 1993.