METRO
MEtallurgical TRaining On-line

Artificial neural networks in analysis of foundry processes

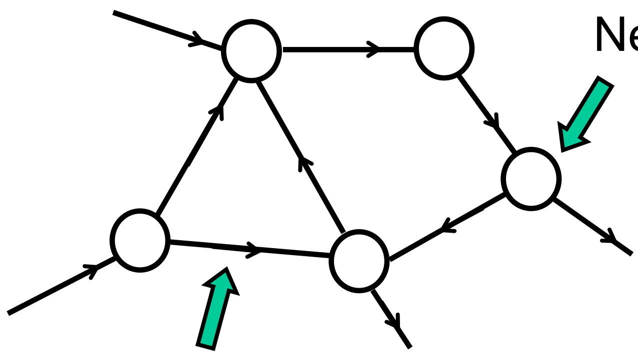
Marcin Perzyk
WUT

Education and Culture

Definition of artificial neural network (ANN)

Artificial neural network (ANN) is a complex mathematical relationship, the structure of which imitates structure and data processing in cerebral cortex of mammals, including humans.

Neuron (network knot)



Synapse
(connection of knots,
sometimes network output)

Synapses transfer values of variables and contain model parameters – synapses weights.

Neurons perform mathematical operations on variables and weights.

Artificial neural networks
Basic advantages

- Ability to learn and to generalise the acquired knowledge. ANNs are able to find regularities in situations of large number of variables of various types. Such regularities often cannot be detected by senses of analysts or other mathematical models.

- ANN is resistant to noise in data as well as errors appearing in some weights, i.e. incorrectly determined individual model parameters.

- Fast data processing, sometimes online.

Artificial neural networks
General characteristics

ANNs are learning systems type models. Values of model parameters (network weights) are determined form results of observations (training examples) by successive corrections in such a way that the network outputs (network responses) approach the real (observed) values. This type of training (learning) is called a supervised learning, most frequently applied.

Example of a relationship determined by ANNs:

$$Y1 = f_1\,(X1, X2, X3, \ldots)$$

$$Y2 = f_2\,(X1, X2, X3, \ldots)$$

X – input signals (independent variables),
Y – output signals (dependent variables)

The coefficients of those equations W (weights) are found (corrected) in a training process from differences between output values predicted by the network Y, and learning ones R (real, i.e. observed):

$$W' = F\,\{W, (Y - R)\}$$

Artificial neural networks
General characteristics

ANNs can perform several *types of tasks*. In modelling of manufacturing processes, including metallurgical and foundry processes, the following are utilised:

- **Regression** or approximation of an unknown multivariable function.
- **Prediction** of future system behaviour on the basis of sequence of values from the past, combined with on-line adaptation of weights.
- **Detection of regularity** (Kohonen type networks). An unsupervised learning is applied, which does *not* make use of any observed (known) output values.

Artificial neural networks
General characteristics

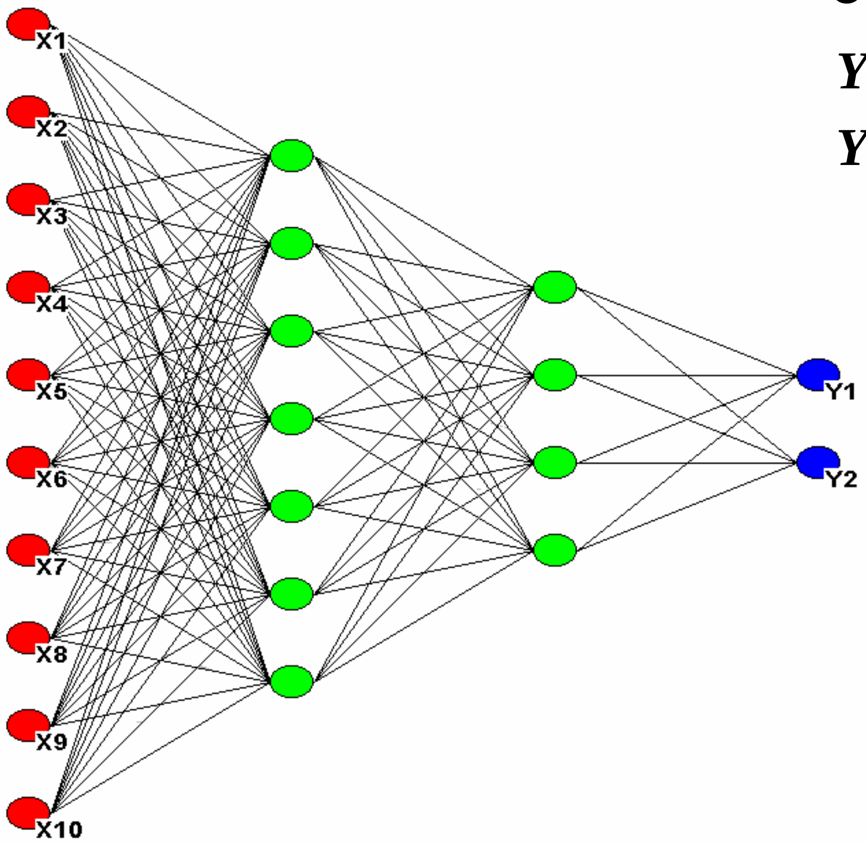
ANNs can have various types of structures and particular configurations within a given type. The most important types are:

- **Multi Layer Perceptron** (MLP) most often used for modelling manufacturing processes.
- **Recurrent network**, characterised by feedbacks between output and input neurons.

Artificial neural networks
Example of MLP type network

This is a four layer MLP used for approximation of function of the type:

$$Y1 = f_1\,(X1, X2, X3, ..., X10)$$
$$Y2 = f_2\,(X1, X2, X3, ..., X10)$$



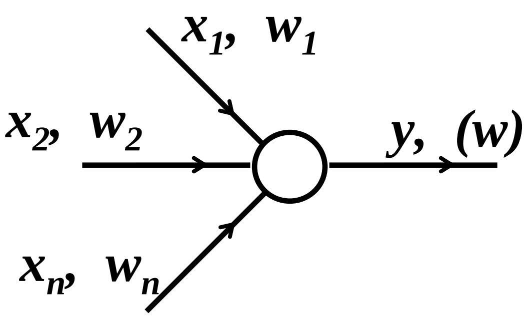
Green colour denotes hidden layers (there are two of them in this example)

Blue colour denotes output layer

Red colour denotes input layer (these neurons do not perform any mathematical operation)

Artificial neural networks

Functioning of a single neuron

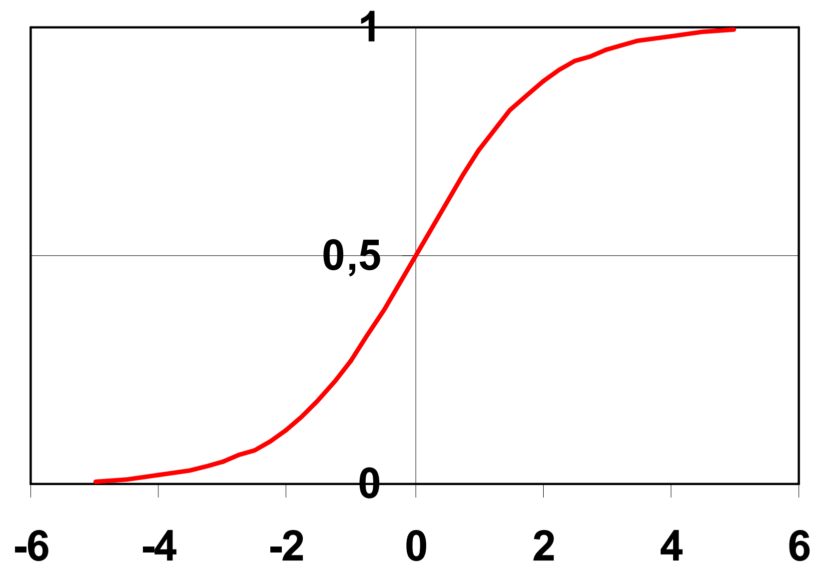


Linear neuron: $$y = \sum_{i=1}^{n} x_i \cdot w_i + w_0$$

Non-linear neuron: $$y = f\left(\sum_{i=1}^{n} x_i \cdot w_i + w_0 \right)$$

f denotes so called activation function, usually of a sigmoidal shape, e.g. given by the formula:

$$f(s) = [1 + exp(-\alpha \cdot s)]^{-1}$$



Supervised learning of ANN's

Training of ANN is a solving of an optimisation task of multivariable function (number of the variables is equal to the number of all weights plus bias terms, present in the whole network).

The goal is to find such values of weights for which the mean square error *E* of all network responses for all experimental data reaches minimum

$$E = \frac{1}{p} \cdot \sum_{k=1}^{p} \left(\frac{1}{m} \cdot \sum_{j=1}^{m} \left(d_{kj} - Y_{kj} \right)^2 \right)$$

m – number of outputs,

p – number of data records (observations),

d – experimental (observation) values

Y – values predicted by network (network responses)

Supervised learning of ANNs
Principles and practice

Available set of experimental data (observations) is usually divided into two parts:

- Basic **training set** used for the corrections of the network weights
- **Verifying (testing) set,** usually smaller, which is used for current calculation of network error for other data than that used for the weights corrections.

The corrections of network weights are made repeatedly, for all records in the training set. One cycle: error calculation – modification of weights is called an *epoch*.

The *end of training* usually takes place when the error for verifying data starts to increase. This is aimed at prevention of the overfitting of the network to the training data, without loosing its ability to make correct prediction for other (new) data.

Methods of supervised learning of ANNs

There are many methods of finding the minimum of the network's error, which can be classified as:

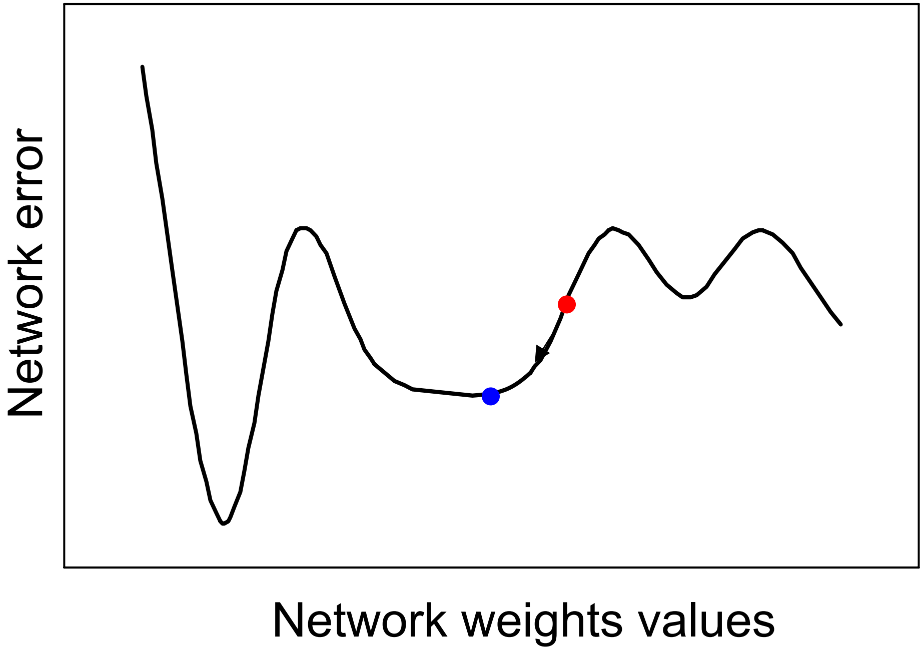
- Gradient methods (most often used)

  The initial values of weights are assumed by random sampling and then the corrections are made in the direction of decreasing error. This often leads to finding a *local minimum* of the network error.

- Methods searching the global minimum of networks error (rarely used)

  Include *simulated annealing* method (discussed later) and *genetic algorithms* based methods (discussed in another lecture).

Supervised learning of ANNs
Illustration of gradient methods

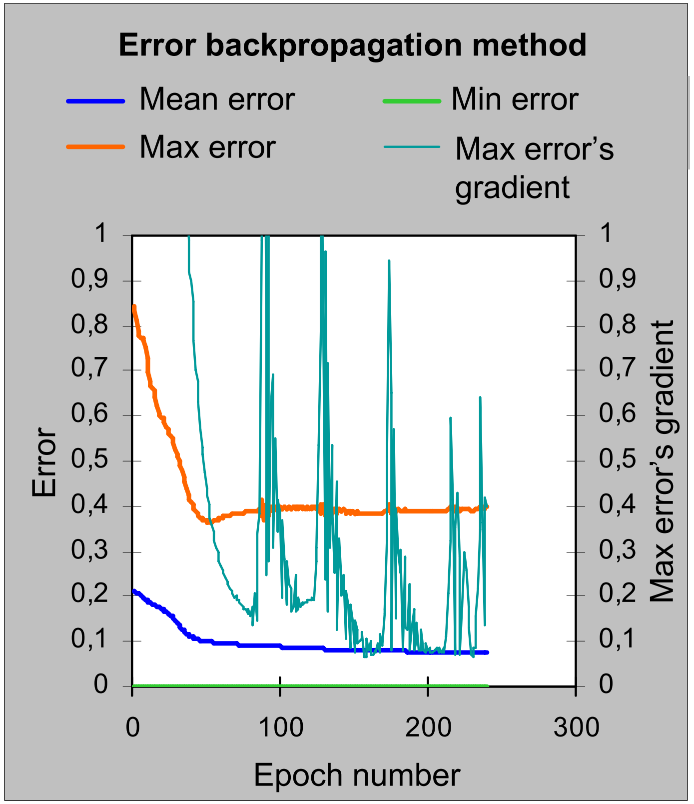


- Randomly selected starting point
- Minimum found by moving in the direction of decreasing error

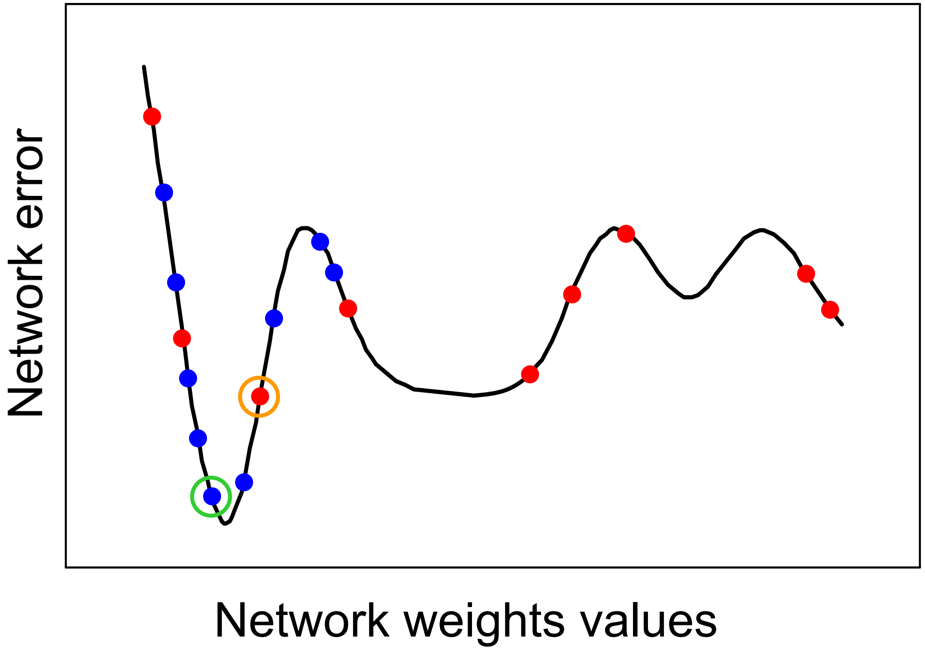
Supervised learning of ANNs
Illustration of gradient methods

A number of gradient methods are known, of which the most often used is the classic *error's backpropagation* method.

The graph presents exemplary changes of the network error in consecutive iterations (epochs). The curves may be completely different for another starting point (chosen by the random selection).



Supervised learning of ANNs
Illustration of simulated annealing method



- Random selections in first or preceding (wider) range (higher ‘temperature’)
- Random selections in next (narrower) range (lower ‘temperature’)

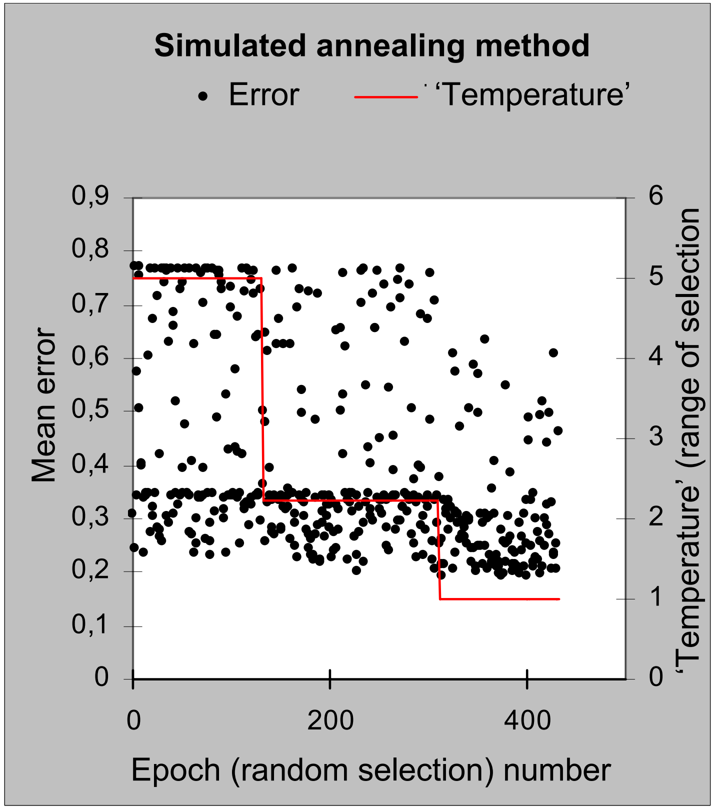
○ Best result of the first random selections (centre of the next selections range)

○ Best result of the next series of random selections

Supervised learning of ANNs
Illustration of simulated annealing method

The graph presents exemplary changes of the network error after repeated random selections of weights, for three, successively decreasing ranges (‘temperatures’).

The simulated annealing method can be used as the single one or as a preliminary one, for selection of the best starting point for a gradient method.



Artificial neural networks
Preliminary data analysis

Hints for selection of input and output model variables:

- The choice of input (independent) variables should be preceded by an importance analysis of possible quantities from the point of view of the output (dependent) variables, utilising statistical methods (analysis of variance). The least significant inputs should be ignored, which will facilitate network's training and analysis of results.
- In case of more then one output variable, it is recommended to consider the use of several networks with single outputs, which leads to reduction of number of weights which have to be found.

For reliable results the *number of training examples* should be *significantly larger* then the number of weights to be found.

Artificial neural networks
Principles of construction of MLP type networks

- *Number of hidden layers* is usually 1, seldom 2 and very seldom 3.
- Larger *number of hidden neurons*, giving larger number of weights, can give more accurate predictions (better flexibility of the model). However, it requires larger training sets or can lead to overfitting of the model to the training data as well as an extension of the calculation time.

  Good practice, as a starting point, is setting the number of neurons in the consecutive hidden layers according to the geometric progression between inputs and outputs numbers.

Artificial neural networks
Training process

- Computer software is used, some of them are available as shareware.
- Different network architectures and also consecutive training sessions of the same network can lead to different results. A good practice is:
  - testing various versions of the network configuration (starting from single hidden layer)
  - testing various number of neurons in hidden layer(s)
  - for each configuration a number of training sessions should be done (e.g. 10).

MLP type networks
Utilisation and analysis of results

- *Quality of a trained network* can be evaluated by calculation of the network's mean error for a testing set not used in the training procedure, i.e. independent form the training and verifying (used for the end of training criterion) data sets.

- If various network configurations end/or a number of training sessions have been applied for solving a regression – type problem, then either:
  - *averaging* of the networks predictions can be utilised, or
  - a single prediction calculated from network with the *smallest error* obtained during training can be taken

MLP type networks
Utilisation and analysis of results

- Elementary way to make use of the trained network is *interrogating*, i.e. calculation of output values (network's responses) for given set of input values.

- An important result of trained network can be also *relative importance factors of input variables*, which facilitate detection of significance of particular parameters for the process.

  There are various methods of calculation of these coefficients. Examples of their application will be presented later.

Artificial neural networks
General industrial applications

- Prediction of properties of products or materials on the basis of the parameters of the manufacturing process involved.
- Replacement of online numerical simulations of physical processes (often time-consuming) by ANN - generalised results of previously made ‘numerical experiments’.
- Materials properties prediction (empirical equations).
- Designing based on the specific data which was collected in the industry and generalised by ANNs.
- Prediction of equipment failures on the basis of selected signals, e.g. load, temperature.
- Neural controllers in automated systems.

Artificial neural networks
Applications in foundry technology

- Breakout forecasting system for continuous casting
- Control of cupola and arc furnace melting
- Power input control in foundry
- Design of castings and their rigging systems
- Design of vents in core boxes
- Green moulding sand control
- Predicting material properties in castings
- Determination of pressure die casting parameters

Applications of ANNs in foundry
Prediction of ductile cast iron properties

ANNs have been used to predict tensile strength, Brinell hardness and elongation of ductile cast iron, based on industrial measurements of the chemical composition of the melt (defined by 9 components: C, Si, Mn, P, S, Cr, Ni, Cu and Mg). Over 800 melts were recorded in an iron foundry.

The MLP type networks of different architectures have been trained 10 times each, using the combined simulated annealing and backpropagation methods.

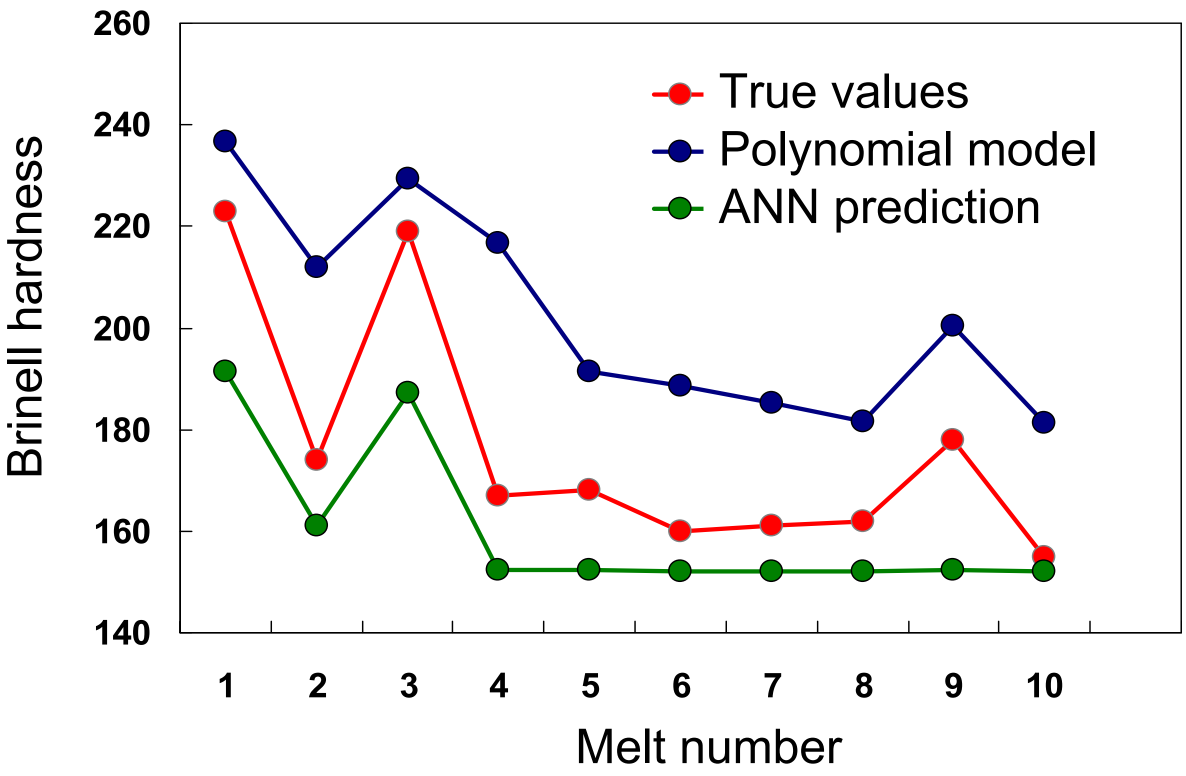
Following comparisons with other models were made:

- Hardness predictions of the network with those obtained from a polynomial type model applied in another foundry in Finland.
- Quality of fitting to the training and verifying data of the network with that obtained for naive bayesian classifier.

Furthermore, relative importance factors of input variables were calculated, which indicated significance of individual chemical elements.

Applications of ANNs in foundry
Prediction of ductile cast iron properties

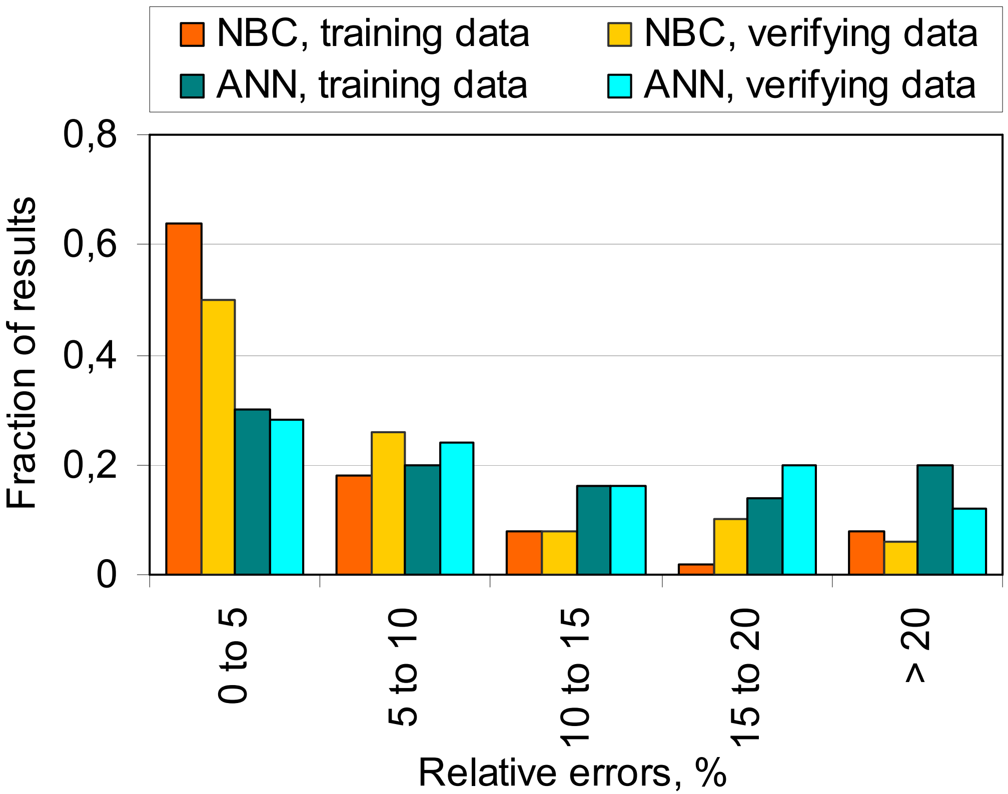
Comparison of hardness predictions of ductile iron on the basis of its chemical composition



Ductile cast iron made in a foundry in Finland

ANN trained on results collected in another foundry

Applications of ANNs in foundry
Prediction of ductile cast iron properties



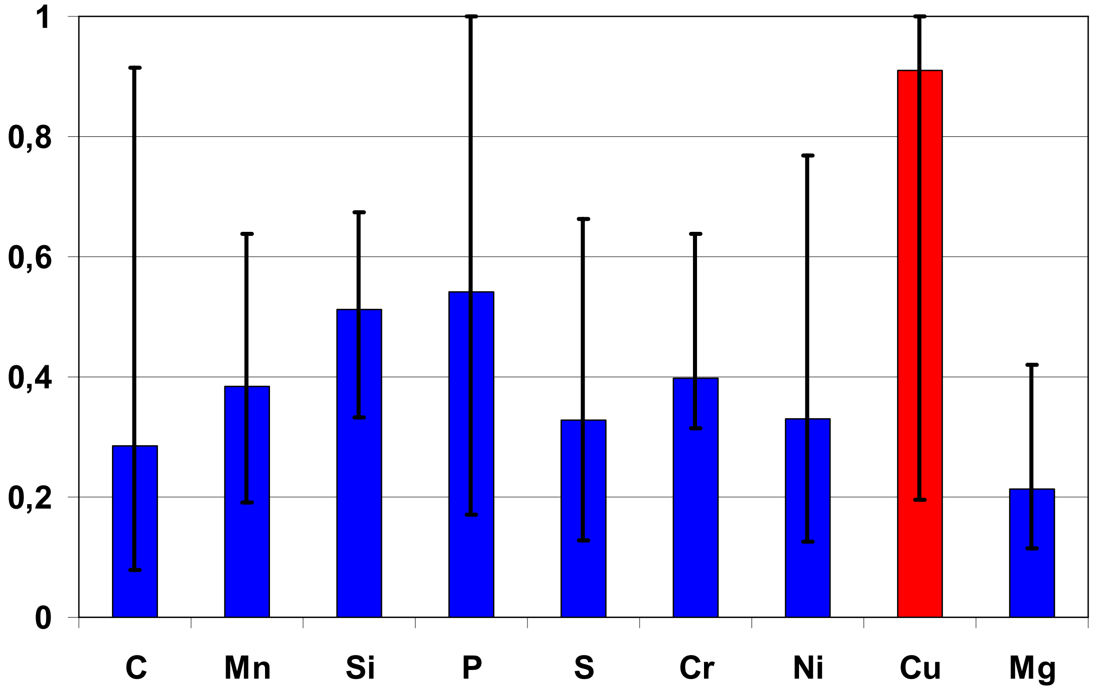
Comparison of prediction errors for tensile strength of ductile iron for ANN and NBC

Obtained error distributions are characteristic for industrial noised data

Notations:
NBC – naive bayesian classifier,
ANN – neural network

Applications of ANNs in foundry
Prediction of ductile cast iron properties

Relative importance factors of chemical elements for strength of ductile cast iron, obtained from a trained ANN. Distinguishing significance of copper is consistent with metallurgical knowledge.



The heights of bars are averaged results from 10 training sessions of the same network.

The black lines denote their ranges (scatter resulting from different training sessions).

Applications of ANNs in foundry
Optimisation of heat treatment parameters of ADI type cast iron

Austempered ductile cast iron (ADI) is one of the most advanced cast structural materials. Its mechanical properties depend on:

- parameters of the raw ductile cast iron heat treatment, i.e.:
  - austentisation temperature
  - austentisation time
  - isothermal tempering temperature
  - isothermal tempering time
- chemical composition of cast iron
- amount and shape of graphite precipitations
- geometry and manufacturing conditions of casting

Applications of ANNs in foundry
Optimisation of heat treatment parameters of ADI type cast iron

A database including over 300 cases was completed, using especially made experiments as well as published data about:

- heat treatment parameters
- chemical composition
- modules of castings
- tensile strength R_m and elongation A_5

Trained MLP type network enables prediction of strength and elongation for a particular casting as a result of as assumed heat treatment parameters.

Applications of ANNs in foundry
Optimisation of heat treatment parameters of ADI type cast iron

Also, a special software has been developed which makes possible to optimise arbitrarily selected input variables (heat treatment parameters in this case) to obtain a desired result: maximum, minimum or an assumed value of any output parameter (e.g. strength).

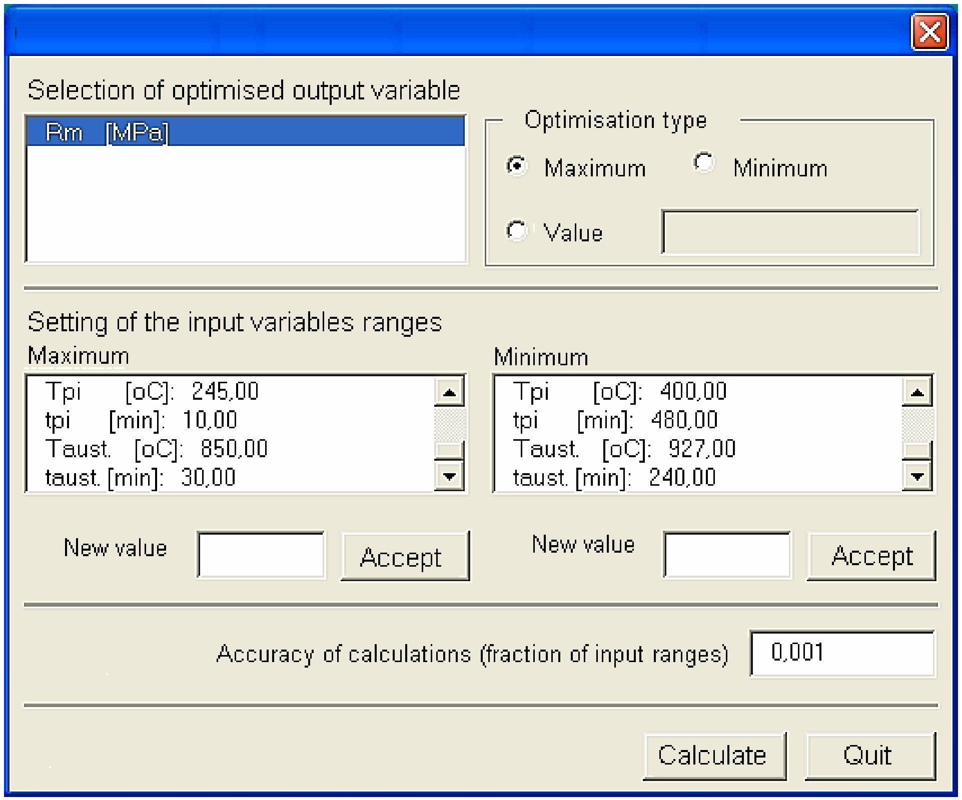
The optimisation algorithm uses the simulated annealing method discussed earlier. In this application it is based on repeated interrogations of the network for randomly assumed (optimised) input values within starting ranges assumed by a user.

Accuracy of calculations can be also set as fractions of the whole ranges of the variability of the inputs. This accuracy is equal to the lowest ‘annealing temperature’ range in the optimisation procedure.

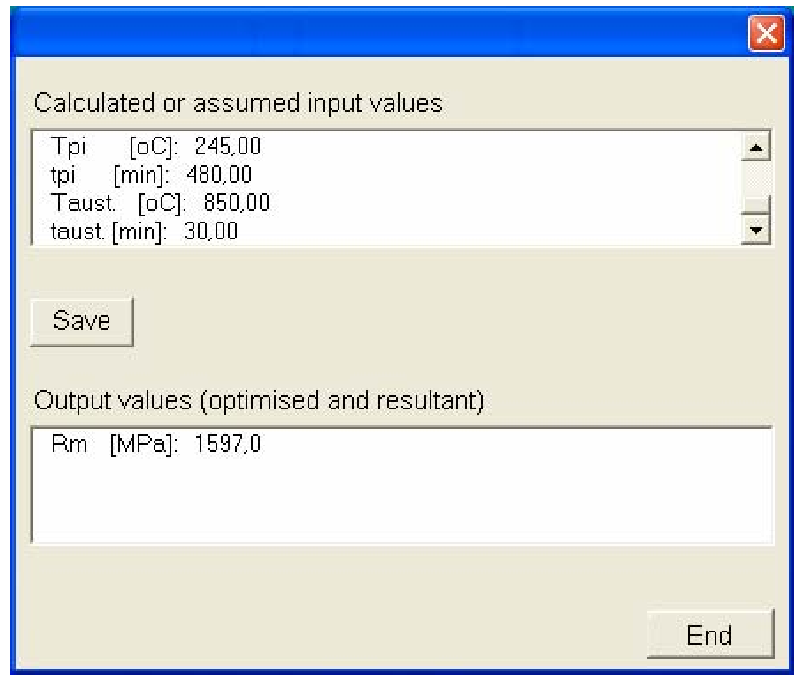
Next slide shows reproduction of two dialog windows captured from the software.

Applications of ANNs in foundry
Optimisation of heat treatment parameters of ADI type cast iron

Setting of optimisation parameters:



Results of optimisation:



Applications of ANNs in foundry
Detection of causes of gas porosity in steel castings

Premises of the project

- An identification of the actual cause of casting defects, especially like gas porosity, is particularly difficult because of a large number of varying parameters which could influence occurrence of the defects.
- It was expected that the trained network would be capable of detecting regularities among those factors and thus indicate the most probable source of the defect.

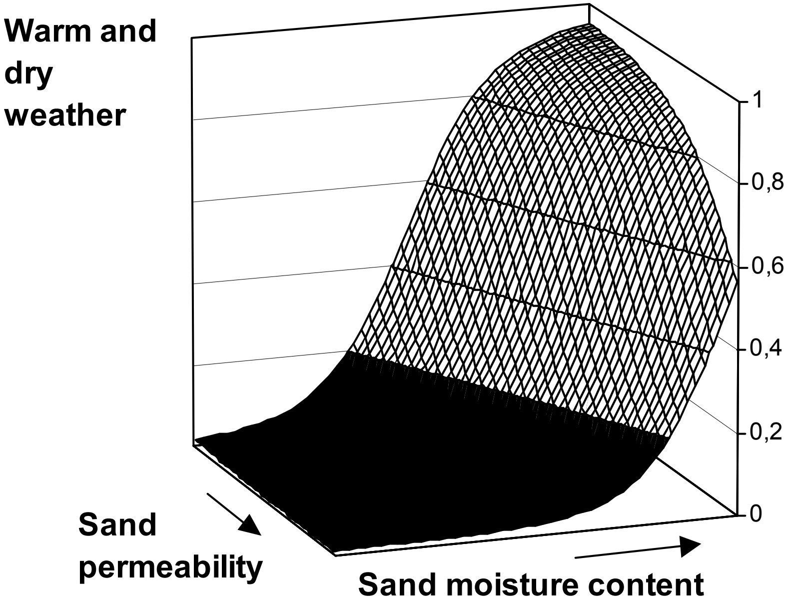
Applications of ANNs in foundry
Detection of causes of gas porosity in steel castings

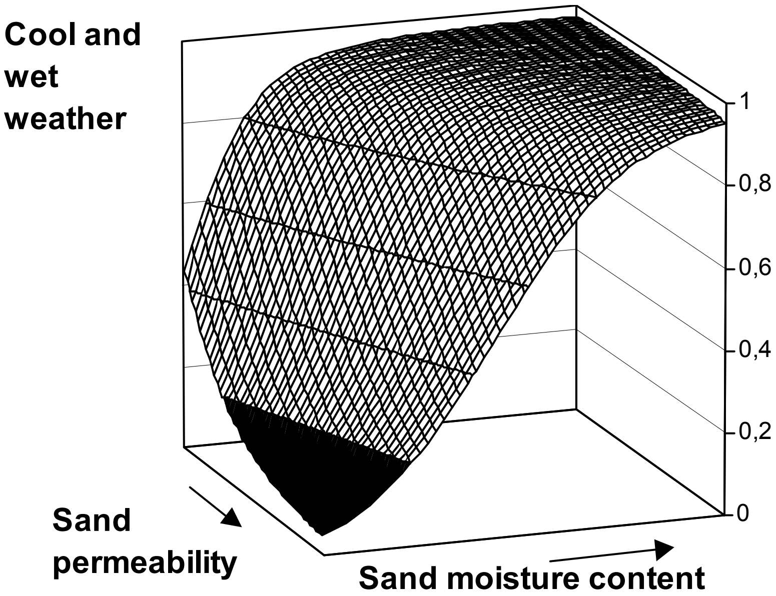
Main stages of the project

- Identification of all possible production parameters which could be related to gas porosity appearance
- Collecting of two types of information: about process parameters, materials used, environment conditions and even employees involved in the production (as the network inputs) and the appearance of a given defect (as the network output)
- Preparation of the training sets for ANN
- Construction, training and testing of ANN
- Analysis of responses of trained network and diagnosis of the most probable sources of the defects.

Applications of ANNs in foundry
Detection of causes of gas porosity in steel castings

Graphs obtained from the trained network, used as a basis of identification of the most probable causes of gas porosity, showing a chance of the occurrence of gas porosity





Applications of ANNs in foundry
Detection of causes of gas porosity in steel castings

Analysis of the graphs has led to the conclusion, that the direct cause of gas porosity was water vapour pressure in the vicinity of the mould cavity, increasing as a result of number of factors:

Vapour pressure in sand mould ↗

↓ ↓

Sand moisture content when pouring ↗ Sand permeability ↘

↓

Sand moisture content during moulding ↗
Time from moulding to pouring ↘
Environment temperature ↘
Air humidity ↗

Black arrows indicate positive or negative effect on the porosity predicted by graphs obtained from trained ANN

Applications of ANNs in foundry
Green moulding sand control aid

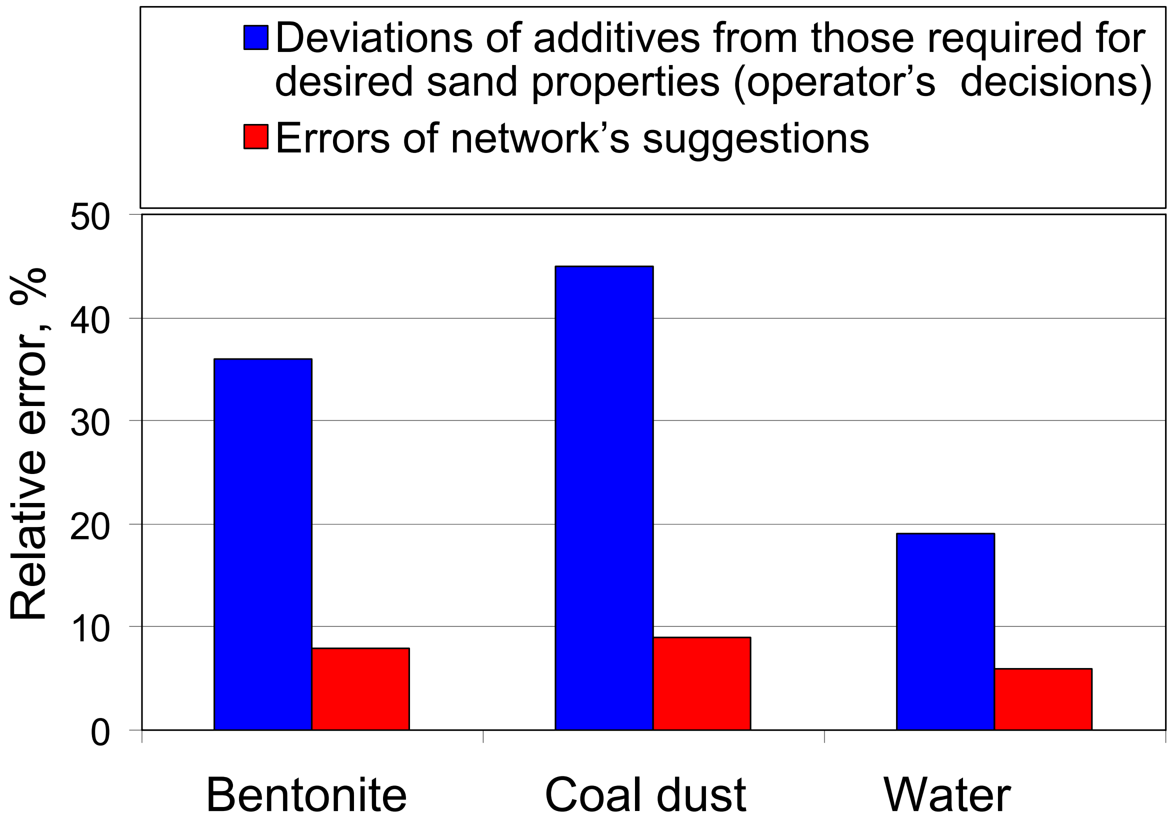
ANN was used as an aid for decision making regarding the new additives to the bentonite moulding sand (water, clay, fresh sand, coal dust).

In non-automated systems an operator decides about the amounts additives, on the basis of the moulding sand tests (moisture, strength, compactibility).

ANN was trained using data collected in a typical, medium size iron foundry.

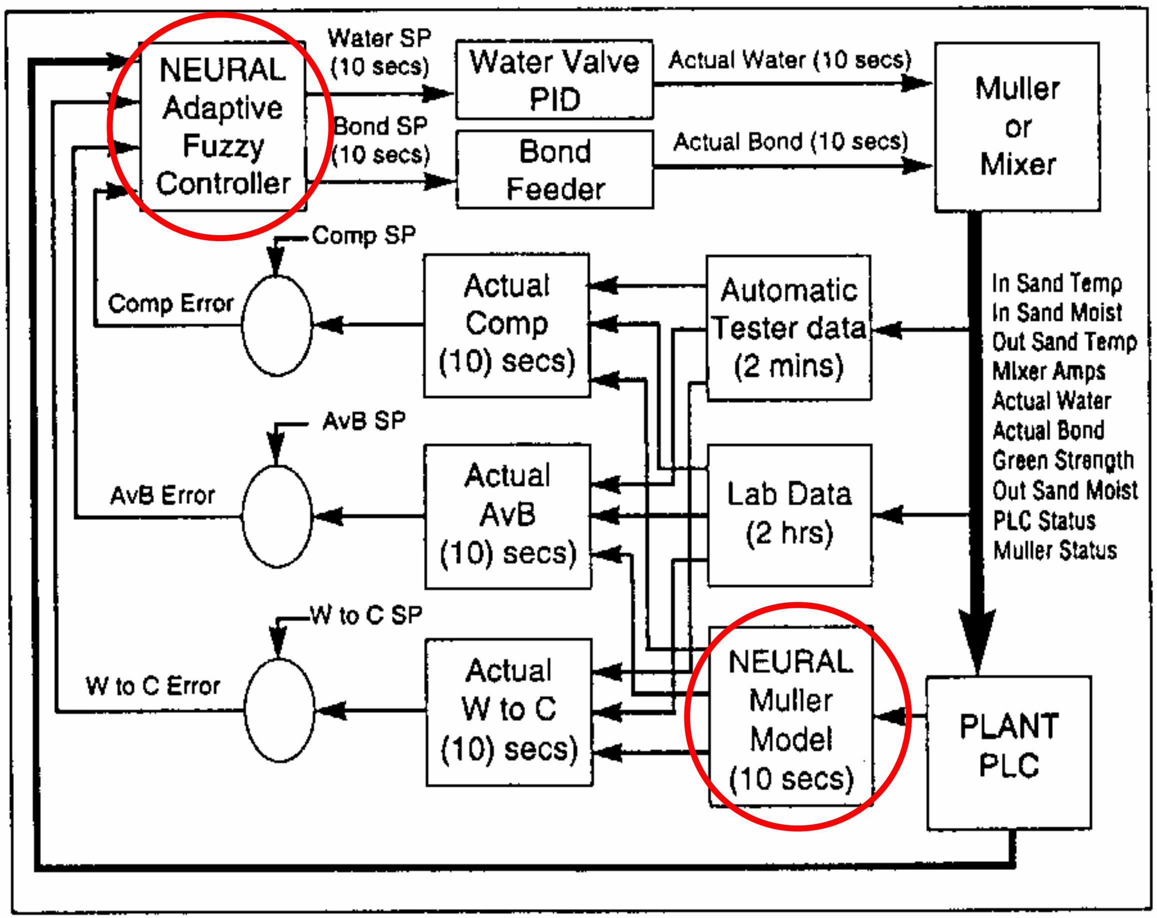
Applications of ANNs in foundry
Green moulding sand control aid

Selected results:



Errors of necessary amounts of additives made by a human were significantly larger from those made by ANN

Applications of ANNs in foundry
Automated system for green sand control



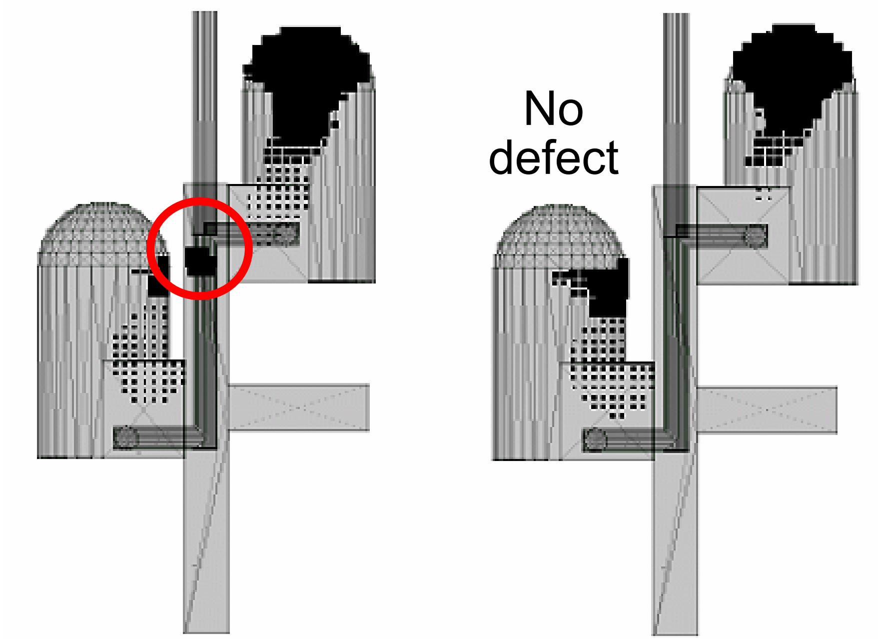
The system operates in John Deer Foundry, Iowa, USA

System elements where ANNs are used are marked by red circles

The neural model for the muller is trained several times per hour using data obtained automatically from the plant PLC system

Applications of ANNs in foundry
Design of feeding systems

The goal of the project was to obtain relationships facilitating determination of correct dimensions of side feeders, which make frequent problems related to local overheating of mould. Simulations of solidification were made for especially designed castings using commercial CAE type software.



Shrinkage defects distributions were obtained, dependent on: feeder's size, height-to-diameter ratio and distance from casting wall as well as the neck dimensions.

On the exemplary picture (left) a typical defect due to too short distance between feeder and casting is marked by a red circle

Applications of ANNs in foundry
Design of feeding systems

ANN generalised the results of numerical simulations of solidification. An appropriate interrogating of the network made it possible to formulate the following recommendations for the feeding systems design:

- Optimal height-to-diameter of a side feeder is 1.5. This value ensures negligible overheating of the casting and relatively favourable modulus to volume ratio.
- Minimum neck's length (distance between feeder and casting wall) is 20 mm.
- For aluminium alloys the sufficient ratio of the feeder-to-casting moduli is 1 for pouring through the feeder and it is equal to 1.3 for a uniform initial temperature distribution in poured metal. For steel castings this ratio is 1.2, for both cases.

Applications of ANNs in foundry
Diagnosis of continuous processes

Many foundry processes can be treated as continuous ones, i.e. with parameters that should be kept constant in a longer time interval. Examples are all mass or large batch processes like melting of one grade alloy, moulding from pattern of the same shape etc.

Diagnosis of continuous process faults or irregularities includes two phases:

- Detection and estimation of degree of process irregularity, on the basis of increase or decrease some of its parameters (e.g. temperature, product properties etc). Typical mathematical tools are statistical process control (SPC) methods. However, in the presented solution this kind of task is performed by ANN (denoted as *neural network I*).
- Identification of the causes of detected irregularities, which traditionally is a result of analysis made by a company staff. In the new approach this is made by ANN (denoted as *neural network II*).

Applications of ANNs in foundry
Diagnosis of continuous processes

L measurements of process parameters from a time window

Parameter 1
Parameter 2
......
Parameter N

Neural network I
Determines character and degree of process irregularity

Degrees of increase of N parameters
Degrees of stability of N parameters
Degrees of decrease of N parameters

Degrees of increase of N parameters
Degrees of stability of N parameters
Degrees of decrease of N parameters

Neural network II
Models relationships between irregularity and occurrence of product defects or device failure

Fault 1
Fault 2
......
Fault M

Applications of ANNs in foundry
Diagnosis of continuous processes

Training of network I

The training set includes of 3 records only, which are made as follows:

INput

L elements = number of measurements in a time window

E.g. for L= 10:

OUTput

3 variables, defining increase, stability and decrease

-1	-0.8	-0.6	....	+0.6	+0.8	+1	1	0	0
0	0	0	....	0	0	0	0	1	0
+1	+0.8	+0.6	....	-0.6	-0.8	-1	0	0	1

Results: for an observed sequence of input values (successive measurements in the time window), e.g. -1, -1, -1, ... 0, +0.5, +1, one obtains triplets of numbers characterising degrees of increase, stability and decrease of given process parameter,
e.g.: 0.72 , 0.01, 0.12 means a remarkable rise.

Training of network II

The training set includes recorded industrial cases, for which degrees and types of process parameters irregularities are known, together with the corresponding occurrences of defects or failures.

In particular, the input variables are the irregularities observed
in a time window, converted to the triplets.

INput

3 x N elements

N – number of process parameters

OUTput

M elements – number of analysed types of defects or failures

Applications of radial base functions neural networks

- Radial base functions neural networks (RBF) can perform regression or classification type tasks, similar to MLP networks with a sigmoidal activation function.
- Typical RBF network includes an input layer, one hidden layer of neurons with radial functions and an output layer, usually with one linear neuron.
- RBF type networks find increasing interest in modelling of manufacturing processes, often exhibiting better representations of such processes, compared to MLP type networks.

Applications of self-organising neural networks

Grouping of signals exhibiting similar characteristics made by Kohonen type networks (unsupervised training) can also have remarkable applications in analysis of manufacturing processes.

Examples of potential applications:

- If a group in which certain combinations of process parameters are included, is characterised also by a larger defectiveness of products, it could mean that this combination is a source of a lower quality.
- If a group in which extreme values of parameters (close to the specified limits) are included is associated with a particular operator it is likely that he does not his work properly.
- If the networks tends to group the process parameters in a number of significantly distinct groups which is greater from the number of different product types, it could mean, that the process suffers from some undesired variations.

METRO
MEtallurgical TRaining On-line

Artificial neural networks in analysis of foundry processes

End of lecture

Education and Culture