Materials Science & Engineering A 567 (2013) 80–83

Contents lists available at SciVerse ScienceDirect

Materials Science & Engineering A

journal homepage: www.elsevier.com/locate/msea

# Modeling of aging process for supersaturated solution treated of Al–3wt%Mg alloy

R.H. Nada [a], D.M. Habashy [a], F. Abd El-Salam [a], Mahmoud Y. El-Bakry [a,b], A.M. Abd El-Khalek [a,b,*], E. Abd El-Reheim [c]

[a] Department of Physics, Faculty of Education, Ain Shams University, Cairo, Egypt
[b] Department of Physics, Faculty of Science, University of Tabuk, Saudi Arabia
[c] Tabbin Institute for Metallurgical Studies (TMS), Helwan, Egypt

ARTICLE INFO

*Article history:*
Received 22 December 2012
Received in revised form
5 January 2013
Accepted 8 January 2013
Available online 12 January 2013

*Keywords:*
Supervised learning
Artificial Neural Network
Rprop
Micro-hardness
Age-hardening

ABSTRACT

The age-hardening curves of micro-hardness measurements obtained for sheets of Al–3 wt%Mg alloy under different temperatures, applied loads and dwell times showed leveling and pronounced oscillations, indicating instability and reflecting a competition between the effect of dynamic recovery or sub-structure coarsening and the effect of solute drag and precipitation hardening. An artificial neural network (ANN) and the Rprop training algorithm were used to model the nonlinear relationship between the parameters of the aging process and the corresponding micro-hardness measurements. The predicted values of the ANN are in accordance with the experimental data. A basic repository on the domain knowledge of the age-hardening process verified the expected effect of micro-hardness decrease by increasing any of the applied parameters.



## 1. Introduction

Weight reduction, reducing fuel consumption and improving fuel efficiency have become a key focus area because of the demand for light-weight alloys for structural applications [1]. In this respect, Aluminum (Al) alloys, of which alloys with Magnesium (Mg) as the major alloying element, have been considered for use in a wide variety of applications [2]. Wrought as non-heat treatable alloys, their strength is derived mainly from solid solution strengthening by Mg, which has a substantial solid solubility in Al, and strain hardening [3].

Micro-hardness testing, as a complex property related to the strength of inter-atomic forces, can be the easiest way to determine the mechanical properties of the different phases of the structure and follow aging behavior during the phase decomposition sequence [4] even at high temperatures [5]. It is therefore important and indispensable to simulate the aging processes by numerical methods in order to control and predict the properties of the Al–Mg alloy.

Artificial neural network as a kind of data mining and artificial intelligence technique is a massively parallel-distributed processor that has a neural propensity for storing experimental knowledge, making it available for future use [6]. Unlike conventional, explicitly programmed computer programs, neural networks are trained through the use of previous example data and then the weights of the neurons are iteratively adjusted until the output for a specific network is close to the desired one. Furthermore, neural networks possess many excellent properties such as outstanding non-linear approximation, auto-adaptation and association capability. As a complex non-linear system, NN models have been widely employed to map the indeterminate relationship between cause and effect variables. [7–12].

During solution treatment and aging process the hardness, $H_v$, of Al–3 wt%Mg alloy depends on three independent aging parameters; temperature ($T$), time ($t$) and load ($L$); which represent the input layer. Hence, the output layer in the present network consists of one neuron representing hardness, $H_v$.

This paper uses an ANN program and back propagation algorithm (Rprop) to model the nonlinear relationship between hardness and temperature. The following sections provide an experimental procedure, brief introduction to ANN, describe the selected ANN structure, provide training data and discuss the results.

## 2. Experimental procedure

An Aluminum–Magnesium alloy containing 3 wt%Mg (Al–3 wt%Mg) was prepared from elements of 99.9% purity (Al and

* Corresponding author at: Ain Shams University, Department of Physics, Faculty of Education, Roxy, Holepoles, Cairo, Egypt. Tel.: +20 1223649582; fax: +20 225500415.
*E-mail addresses:* asaad_abdelkahlk@yahoo.com, asaad_abdelkahlk@hotmail.com (A.M.A. El-Khalek).

0921-5093/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.msea.2013.01.009

Mg). Aluminum was melted in a graphite crucible placed in the stable zone of a muffle furnace adjusted at 1123 K. Mg was added to the Al until it completely melted. Finally, the casting was carried out in hot iron molds. The melt was cooled in iced water. The ingot was given an initial annealing for 8 h, at 773 K, and then quenched in cold water. The ingot was rolled with intermediate annealing at 773 K to obtain sheets of 7 cm length, 3 cm width and 0.1 cm thickness. The micro-hardness measurements were obtained by using a Leco micro-hardness tester (LM700). The tested sheets were heated for 2 h in the working temperature range from 443 K to 503 K in steps of 10 K, then quenched in cold water kept at room temperature (RT). The surface of any tested sample was polished using a polishing machine (KNUTH-ROTOR 2 STRUERS), and then examined by an optical microscope having magnifications of 20 × and 50 ×. Hardness indentation was obtained by applying the loads 10, 50, 100 and 300 g for the dwell times 10, 20, 30 and 40 s. The end button was pressed to show the mean diagonal value and the corresponding hardness value.

## 3. Artificial neural network

Neural networks are basically a connective system [13–19] in which various nodes, called neurons, are interconnected. A typical neuron receives one or more input signals and provides an output signal depending on the processing function of the neuron. The most popular neural networks are feed forward networks. During the training process, the network adjusts its weights to minimize the error between the predicted and actual outputs. The most common algorithm for adjusting the weights is the back propagation algorithm. In the forward pass, the input signals propagate from the network input to the output. In the reverse pass, the calculated error signals propagate backwards through the network, where they are used to adjust the weights. Any efficient optimization method can be used for minimizing the error through weight adjustment. An example of a neuron with a sigmoidal transfer function is shown in Fig. 1. This simple processing unit is known as an elementary perceptron.

### 3.1. Improvement on BP training algorithms

Training of the neural network was done in matlab, using the trainrp function. Trainrp is a network training function that updates weights and bias values according to the resilient back-propagation algorithm (Rprop). The Rprop algorithm proposed by Riedmiller and Braun is one of the best performing first-order learning methods for neural networks. In the Rprop learning algorithm, the direction of each weight update is based on the sign of the partial derivative $\partial E/\partial w_{ij}$ (let $w_{ij}$ denote the weight in a neural network from neuron $j$ to neuron $i$, and $E$ an arbitrary error measure that is differentiable with respect to the weights). A step-size, i.e., the update amount of a weight, is adapted for each weight individually. The main difference to other techniques is that the step-sizes are independent of the absolute value of the partial derivative. The benefits of this update scheme are described in Ref. [20], one iteration of the original Rprop algorithm can be divided into two parts. The first part, the adjustment of the step-sizes, is basically the same for all algorithms employed in this study. For each weight, $w_{ij}$, an individual step-size, $\Delta_{ij}$, is adjusted using the following rule:

$$\Delta_{ij}^{(t)} = \begin{Bmatrix} \eta^{+}\Delta_{ij}^{(t-1)}, \text{ if } \frac{\partial E^{(t-1)}}{\partial w_{ij}} \times \frac{\partial E^{(t-1)}}{\partial E_{ij}} > 0 \\ \eta^{-}\Delta_{ij}^{(t-1)}, \text{ if } \frac{\partial E^{(t-1)}}{\partial w_{ij}} \times \frac{\partial E^{(t-1)}}{\partial E_{ij}} < 0 \\ \Delta_{ij}^{(t-1)}, \text{ else,} \end{Bmatrix}$$

where $0<\eta^{-}<1<\eta^{+}$. If the partial derivative $\partial E/\partial w_{ij}$ possesses the same sign for consecutive steps, the step-size is increased, whereas if it changes sign, the step-size is decreased (the same principle is also used in other learning methods, e.g., in Refs. [21,22]). The step-sizes are bounded by the parameters $\Delta_{\min}$ and $\Delta_{\max}$. The second part of the algorithm is the update of weights.

### 3.2. Modeling the hardness using ANN

The proposed ANN model of hardness can be viewed as a two inputs–one output model. The inputs are temperature and loads (10, 50, 100 and 300 g) at different times (10, 20, 30 and 40 s) while the output is hardness $H_v$. As the nature of the inputs (different times) is completely different from each other, authors choose to internally model the problem with four individual neural networks trained separately using experimental data.

The first ANN was configured to have temperature and time(10 s) as inputs at different loads (10, 50 and 300 g). The output is the hardness (see Fig. 2). Using this input–output arrangement, different network configurations were tried to achieve good mean sum square errors (MSSE) and good performance for the network. The four-layer configuration shown in Fig. 3 is chosen. These layers are: three hidden layers of 80, 90 and 60 neurons respectively, and the output layer consisting of one neuron. The transfer functions were chosen to be a logsig function for the first hidden layer and poslin for the second and third hidden layers, while the output layer was chosen to be the linear pureline function.

The second ANN (at $t=20$ s) with three hidden layers of 70, 60 and 60 neurons respectively, the third ANN (at $t=30$ s) with three hidden layers of 80, 70 and 67 neurons respectively, and the fourth ANN (at $t=40$ s), three hidden layers of 50, 60 and 70 neurons respectively, were found to give the minimum mean square errors and exact modeling.

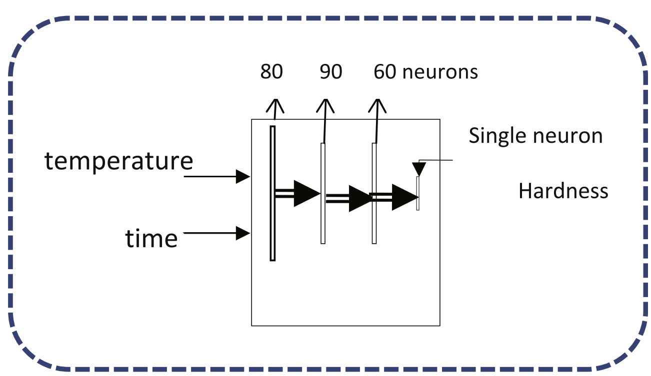


**Fig. 2.** Block diagram of the first ANN based modeling.

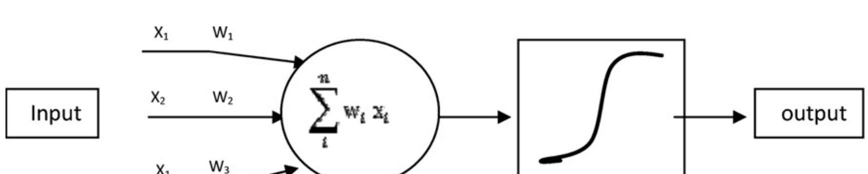


**Fig. 1.** Neuron with a sigmoidal transfer function.