Applied Soft Computing 46 (2016) 260–266

Contents lists available at ScienceDirect

Applied Soft Computing

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

New training strategies for neural networks with application to quaternary Al–Mg–Sc–Cr alloy design problems

S. Ganguly [a], A. Patra [b], P.P. Chattopadhyay [c], S. Datta [d,*]

[a] Department of Metallurgical Engineering, National Institute of Technology Raipur, Raipur, Chhattisgarh 492010, India
[b] Department of Mechanical Engineering, Hooghly Engineering & Technology College, Hooghly 711103, India
[c] Department of Metallurgy and Materials Engineering, Indian Institute of Engineering Science and Technology, Shibpur, Howrah 711103, India
[d] Calcutta Institute of Engineering and Management, Kolkata 700040, India

ARTICLE INFO

Article history:
Received 24 November 2014
Received in revised form 25 March 2016
Accepted 12 May 2016
Available online 18 May 2016

Keywords:
Artificial neural network
Multi-stage training
Back propagation
Genetic algorithm
Multi-objective optimization
Alloy design

ABSTRACT

This study concerns the training of a neural network in multiple stages considering minimization of errors from multiple data/pattern resources. The paper proposed a dual stage multi-resource data training scheme using multi-objective genetic algorithm. The training scheme has been used for the design and development of efficient neural network model focusing on missing, but most informative domains of the data set by means of introducing only a few patterns from missing domain treated separately during the later stage of training. The trained model has been used to design a quaternary Al–Mg–Cr–Sc alloy system, from the information subsets of binary Al–Cr and the ternary Al–Mg–Sc alloys. The validity of the proposed algorithm has been discussed in light of the evolution of the ageing characteristics of the new aluminium alloy system.

1. Introduction

The experimental design and development of aluminum alloys are founded on the basis of some significant alloying addition leading to the development of specific alloy series viz. 2xxx, 5xxx and so on Ref. [1]. The increasing uses of the alloys in automotive and aerospace applications [2,3], demand high and reliable performance resulting in search of new improved aluminum alloy systems. The unique feature of such alloys is low density with adequate strength. The chief mechanism for increasing the strength of the aluminium alloys is the heat treatment called age hardening. During this heat treatment precipitates of intermetallic compounds form within the matrix. In case of aluminium–magnesium alloy, such phenomenon could not be observed. Addition of other elements like chromium and/or scandium may impart this phenomenon, and thus improve the strength of the alloy. Therefore, the design of the new aluminum alloys based on the response knowledge space from the known binary to ternary systems and the known ternary to quaternary systems is important.

Recently, neural network (NN) tool has received increasing attention for data analysis and explore the functional properties space by modeling the material systems [4–8]. Usually, NN based materials design techniques aims to train the system by learning algorithms, where some data are developed from experimental studies. Then these models are used to design the chemistry, process and structure space of the materials prior to the experimental development of the alloys. NN along with other computational intelligence tools find promising applications in materials informatics and alloy design approaches [9–16]. Among the different varieties of neural networks, the feed forward NNs (FNNs) features the most general approach to represent the network structure, simplicity in representing a problem, and availability of number of training algorithms [17]. Probably because of these reasons, FNNs are widely popular in many engineering applications, and alloy design is certainly no exception [18–25]. This study also focuses on the application of FNNs in the context of the design of age-hardenable grade of aluminum alloys.

When a FNN is applied to function approximation task, like mapping of alloy properties as output from the chemistry and process parameter space as the inputs, the primary interests are to develop a predictive model and to explore the pattern of underlying phenomenon in the alloy system(s) [23]. Thus, a major objective is generalization of predictability of NN to the unknown data regime in order to design new alloys with improved properties and wider

* Corresponding author.
E-mail address: shu.datt@gmail.com (S. Datta).

http://dx.doi.org/10.1016/j.asoc.2016.05.017
1568-4946/

engineering applications. Nevertheless, given the properly selected network size and training algorithm, the selection of training data set substantially regulate the performance of the trained network. This is a problem of pragmatic interest of the neural network users. Only a limited amount of research has been conducted on this issue [26] and asks for more attention of researchers in future. Acute problem appears when the data have missing or sparse information of any particular pattern, which is of major importance for the purpose the model is developed for. In such condition, it becomes hard to incorporate the particular hidden pattern during future prediction or design using the NN model. As in the present context, the in-house generation of training data being beyond the scope, the data is taken from literature, which is generated with time in different laboratories. Hence, the data pattern for training is limited by the traditional choice of the researcher in contemporary experimental interest. The objective of the NN analysis of the data in this work was to recognize the age hardening pattern of the quaternary Al–Mg–Sc–Cr alloy system. In the present application, a new training strategy for neural networks has been proposed on a data that inherits the age hardening knowledge of ternary Al–Mg–Sc alloy system as well as binary Al–Cr alloy system, but the age hardening pattern of the quaternary Al–Mg–Sc–Cr alloy remain missing. The missing pattern of aging behavior of the alloy has been recognized by the multi stage training through the additional resource of micro-data experimentally developed for the purpose. In short, published experimental data of two different alloy systems are used to develop the model, which failed to predict a peculiar behavior of a third alloy system, which was developed experimentally by the authors mixing the first two alloy systems. The problem of modeling such systems revealed here and a solution is also proposed.

In the proposed strategy of the present work, the initial stage of the training follows the concept of FF backpropagation with scale conjugate gradient descent training algorithm. However, second stage of training focuses on the micro-data of the missing pattern domain along with the original data of the stage one in multi-objective data training scheme using a multi objective genetic algorithm (MOGA) aiming to minimize the both the error simultaneously as discussed in the following sections.

2. Structure of the new training approach

2.1. Conventional single layer backpropagation neural network training

A **back propagation** NN comprises an input layer, one or more hidden layer and an output layer where each of the layers consists of number of neurons or nodes. In the general case, the number of nodes in the input and output layer is determined by the application problem under study, while that of the hidden layer is needed to be heuristically devised by the user. On the other hand, each node of a layer is connected to every node of the next layer through synaptic connections. In the present application, we use a single layer 9-*N*-1 type network structure, where 9 is the number of input neurons constituted by the alloy chemistry (Mg, Sc, Cr, Si, Zr and Ti), percent cold deformation, ageing temperature (A.Temp.) and ageing time (A.Time). *N* is the number of hidden neurons in the only hidden layer determined through trial and error method. Hardness is the only output in our study. The connectivity between the layers are developed using the tan-hyperbolic ($\tan h$) transfer function at the hidden layer and linear transfer function at the output layer. The computed output of the node in the output layer can be represented as follows [12]:

$$y = \left[\sum_{1}^{j} W_j \left\{ \tanh\left(\sum_{1}^{i} w_{ji}x_i + b_j\right)\right\} + b\right] \quad (1)$$

where y is the output of the node in the output layer; x_i is the input of the ith node in the input layer and w_{ij} is the connective weights between nodes in the inputs and hidden layer. W_j represents the connection weights between the nodes in the hidden layer and the outputs; and b_j and b are bias terms that represent the threshold of the transfer function.

The learning error e, can be calculated by the following formula:

$$e = \sum_{q=1}^{p} \frac{\left(y_q - a_q\right)^2}{p} \quad (2)$$

where e is the mean square error of the actual output and desired output of the training samples used for training. This error e is defined as the fitness function in multi criteria training using GA as described later.

2.2. Need for multi-objective training of neural network

It is well known that the selection of data/patterns for network training is a significant issue. This is a problem that has tangible implications and always been the attention of researcher. The training data set selection can have significant impact on the performance of the trained network. Particularly, when we use a trained network for prediction in the interpolation regime or in the regime for which the data pattern density in the training data is poor, the performance is surprisingly poor. During training of the network, the algorithms adjust the weights focused on the overall prediction error on the training or testing data. However, the example pattern with poor data pattern density in the training or testing data fail to gain the necessary importance in the training process. As a result, the trained network prediction performance becomes poor and highly uncertain in this regime. If beside the main stream of data, the data for the poorly populated regime is treated separately and the error information computed from both the data set separately in the parameter adjustment process during training, the problem can resolved. Nevertheless, the training algorithm has to take care of more than one error information and the weight adjustment will take place under a multi-objective optimization scenario. In multiple objective scenarios, the conventional mathematical optimization techniques based training algorithms such as scale conjugate gradient or Levenberg–Marquardt based algorithms are not efficient. On the other hand, multi objective genetic algorithm (MOGA) is a fascinating heuristic tool to solve such problem [27]. Therefore, a training scheme efficient for multi-source date using MOGA has been proposed and presented here.

2.3. Multi-objective genetic algorithm (MOGA)

The genetic algorithms (GAs) are optimizers influenced by the principles of natural selection and natural genetics [28,29] mimics the Darwin's theory of evolution "survival of the fittest". In contrast to the many gradient-based traditional optimization techniques, GA does not start with a single guess solution [28]. Nevertheless, it generates number of random individuals each signifying a possible solution. A population of such individuals undergoes a process of simulated Darwinian evolution through application of genetic operators typically known as selection, cross over and mutation. In this manner, a new generation is obtained, which may offer better individuals as optimal solution.

In case of multi-objective problems, the genetic search is performed following the concept of Pareto-optimality [30]. In case of multi-objective problems, a search is generally performed following the concept of Pareto optimality. In contrast to single-objective optimization, where fitness assessment can easily be done from the objective function use of section operator is simple, in multi-objective optimization fitness assignment and selection is more