Journal of the Korean Ceramic Society
Vol. 41, No. 5, pp. 353~358, 2004.

Artificial Neural Network Models in Prediction of the Moisture Content of a Spray Drying Process

Osman Taylan† and Ali Haydar*

Department of Industrial Engineering, College of Engineering, King Abdulaziz University, P.O. Box 80204, Jeddah 21589, Saudi Arabia
**Department of Computer Engineering, Girne American University, North Cyprus, via Mersin 10, Turkey*
(Received December 11, 2003; Accepted January 7, 2004)

ABSTRACT

Spray drying is a unique drying process for powder production. Spray dried product must be free-flowing in order to fill the pressing dies rapidly, especially in the ceramic production. The important powder characteristics are; the particle size distribution and moisture content of the finished product that can be estimated and adjusted by the spray dryer operation, within limits, through regulation of atomizer and drying conditions. In order to estimate the moisture content of the resultant dried product, we modeled the control system of the drying process using two different Artificial Neural Network (ANN) approaches, namely the Back-Propagation Multiplayer Perceptron (BPMLP) algorithm and the Radial Basis Function (RBF) network. It was found out that the performance of both of the artificial neural network models were quite significant and the total testing error for the 100 data was 0.8 and 0.7 for the BPMLP algorithm and the RBF network respectively.

Key words : Spray drying, Artificial neural network, Modeling, Process control

1. Introduction

Spray drying is a drying process, since it involves both particle formation and drying. Powder characteristics and properties can be controlled and maintained constant throughout a continuous operation. Spray drying is the transformation of feed from a fluid state into a dried particulate form by spraying the feed into a hot drying medium. The feed can either be a solution, a suspension, an emulsion or a paste. The resulting dried product conforms to powders, or granules. The form of which depends upon the physical and chemical properties of the feed and the dryer design and operation (K. Masters). Spray drying is a procedure which in many industries meets the most desirable dried product specifications for subsequent processing or direct consumer usage. Any form of dryer provides means of moisture removal by application of heat to the feed product and control of the humidity of the drying medium.

Spray dryers can be controlled either manually or by automatic systems. Manual control is applied to small plant and large industrial units operations. S. J. Lukasiewicz states that "the demands on operating continuous operation and the maintenance of constant product quality over lengthy durations of production make automatic control a virtual necessity".

†Corresponding author : Osman Taylan
E-mail : osman_taylan@yahoo.com
Tel : +966-6952037 Fax : +966-6401686

2. System Parameters Identification

Many operational variables associated with atomization and the drying operation can alter the characteristics of the dried product. Free-flowing characteristics of dried product are vitally important. The variables causing flow-ability and rate of humidity are summarized as follows; Humidity, the result of moisture content, is an important product parameter for control system of spray dryers. E. Negre and E. Sanchez searched about the system parameters, and found out that the humidity of particles depend on so many other parameters including the outlet temperature, the inlet temperature, the feed viscosity, the pressure of the production system, the amount of chemicals used and also the solid content of the slurry in spray drying process.

The outlet temperature represents the product quality, i.e. bulk density, color, flavor, and activity, as well as the moisture content. For a fixed moisture content and dryer design, the outlet temperature must be kept within a narrow range to maintain the powder packing and flow requirements. The increase in outlet temperature decreases moisture content at constant air-flow and heat-input conditions. The increase in the inlet temperature, which is the other parameter that effects the moisture content of the spray drying process, increases the evaporative capacity at constant air rate. Increased inlet temperature often causes a reduction in bulk density, as evaporation rate is faster than products dry so it causes a more porous or fragmented structure (J. S. Reed).

The feed viscosity is another parameter that effects the product characteristics; An increase in feed viscosity through increase in feed solids or reduction in feed temperature will produce coarser sprays on atomization at fixed atomizer operating conditions. Increase in feed solids affects evaporation characteristics, where increases in particle and bulk density results. The feed pump transfer the product to the atomizer either directly or via a constant head feed tank. One of the most important features is the ease of particle size control merely through wheel speed control. The mean size of the product is directly proportional to the feed rate and feed viscosity and inversely proportional to the wheel speed and wheel diameter. Variation of pressure gives control over feed rate and spray characteristics. O. Taylan and H.Taskin developed a fuzzy model of spray dryers. The mean size of a spray product is directly proportional to feed rate and viscosity, and inversely proportional to pressure. For low viscosity feeds, fine particles can be produced, although the resulting dried powder may be agglomerated (O. Taylan, A. Golec, and H. Taskin).

Spray drying is an expensive method of evaporating volatiles to obtain optimum heat utilization conditions. The spray dryer should always be fed with the maximum solids feedstock possible. Increased feed solids lead to heavy drying chamber deposits and product degradation. The heat required to evaporate a given quantity of water is virtually the same irrespective of product output and thus if there is significant increase in product output with increase in feed solids for a given evaporative capacity, this is a clear indication of a reduction in the heat required to produce a unit weight of powder on increasing the feed solids.

In the spray drying process, as it was explained above, the relation between the moisture content and the input parameters are highly non-linear. But, spray dried powder must have a controllable particle size distribution, consists of spherical particles which is ideal for pressing operations. These characteristics meet the correct degree of free-flow properties required for pressing operations that requires a specific powder quality to overcome product sticking to the dies. Sticking leads to non-uniformity in ceramic surfaces. This non-linear relations can be modeled using non-linear empirical modeling techniques. Neural networks are capable of approximating any continuous non-linear functions, and have been applied to non-linear process modeling. A major task for a neural network is to learn a model of the environment in which it is embedded and to maintain the model sufficiently consistent with the real world so as to achieve the specific goals of the application of interest. In any event, the observations so obtained provide the pool of information from which the examples used to train the neural network are drawn.

In this paper, we used two different artificial neural network models to represent the knowledge about the spray drying process. In the next section, we explain these two models used to predict the moisture content of the process in detail.

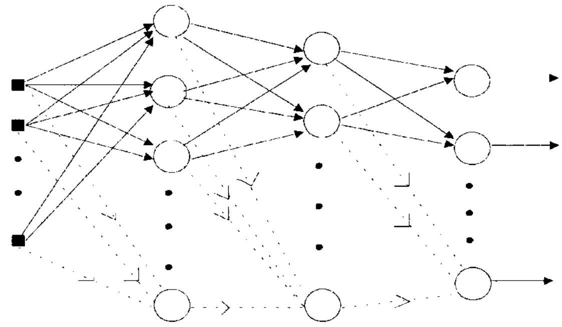

Fig. 1. Multilayer perceptron network.

3. Model Description

Artificial Neural Networks (ANNs) are systems that are deliberately constructed to make use of some organizational principles resembling those of the human brain (Chin-Teng Lin and C. S. George Lee). Generally speaking, artificial neural networks are computing systems made up of a number of simple highly interconnected signals or information processing units that are called as artificial neurons.

In this work, two different architectures are used to model the system at hand. The first approach is the Back-Propagation Multilayer Perceptron (BPMLP) algorithm. The Multilayer Perceptron (MLP) network is a feedforward network in which artificial neurons are arranged in a feedforward manner (A. Cichocki and R. Unbehauen). The network architecture of a MLP is shown in Fig. 1. It consists of n_0 inputs, n_1 artificial neurons in the first hidden layer, n_2 artificial neurons in the second hidden layer and n_3 outputs.

The MLP is supposed to perform a specific nonlinear mapping which can be expressed in terms of a given set of learning examples. Learning of the MLP consists in the adaptation of all synaptic weights in such a way that the discrepancy between the actual output signals and the desired signals, averaged over all learning examples, is as small as possible.

The standard back-propagation algorithm uses the steepest descent algorithm to minimize the mean squared error function. The error function for the p' th example is defined as follows

$$E_p = \frac{1}{2}\sum_{j=1}^{n_3}(d_{jp}-y_{jp})^2 = \frac{1}{2}\sum_{j=1}^{n_3}e_{jp}^2 \quad (1)$$

where, e_{jp}, d_{jp}, and y_{jp} are the instantaneous squared error, desired output signal and the actual output signal for the p' th learning example, respectively. The global error function can be summed over all learning examples and can be given as follows

$$E_{total} = \sum_p E_p = \frac{1}{2}\sum_p\sum_j e_{jP}^2 \quad (2)$$

In this study, we used the on-line algorithm to update the weights. In the on-line algorithm, for each learning example presented as an input, all weights are updated before the next learning example is presented. In this algorithm, all the synaptic weights w_{ji} are changed by an amount of Δw_{ji} where

$$\Delta w_{ji} = -\eta \frac{\partial E_p}{\partial w_{ji}}, \qquad \eta > 0 \tag{3}$$

In the above equation η is a learning parameter. One can derive an updating formula for the weights given as

$$\Delta w_{ji} = \eta \delta_j o_i \tag{4}$$

where δ_j is the local gradient of the hidden neuron j and o_i is the function signal at the output of neuron i. This function signal o_i at the output of neuron i is obtained by passing the weighted sum of inputs to neuron i from a nonlinear activation function. The activation function chosen in this study is a unipolar sigmoid function which can be defined as follows

$$o_i = \Psi(u_i) = \frac{1}{1+\exp(-\gamma_i u_i)} \tag{5}$$

where $\gamma_i > 0$ is a constant value.

In Equation 5, u_i is the weighted sum of the inputs to the neuron i, and for the first hidden layer it can be defined as

$$u_i = \sum_{j=1}^{n_0} w_{ij} x_j + \theta_i \tag{6}$$

where θ_i is a bias value and x_j is the j' th component of the input pattern (learning example). The local error of the internal hidden layer is determined on the basis of the local errors at the upper layer. Starting with the highest output layer we compute δ_j^{out} which is a vector of the local gradient at the output layer of the j' th neuron using the equation given as

$$\delta_j^{out} = (d_{jp} - y_{jp}) \frac{\partial \Psi_j^{out}}{\partial u_j^{out}} \tag{7}$$

where Ψ_j^{out} is the unipolar sigmoid function at the output layer. One way to improve the back-propagation multilayer learning algorithm is to smooth the weight changes by over-relaxation. In other words, by adding the momentum term, that is defined as;

$$\Delta w_{ji}(k) = \eta \delta_j o_i + \alpha \Delta w_{ji}(k-1) \tag{8}$$

where $0 \le \alpha < 1$

The second term in Equation 8 is called the momentum term that makes the current k' th search direction an experimentally weighted average of past $k-1$' th directions. This term damps the effect of the learning parameter, η, that may cause parasitic oscillations which prevent the algorithm from converging to the desired solution. Hence, it enables the improvement of the convergence rate and the steady state performance of the back-propagation multilayer learning algorithm.

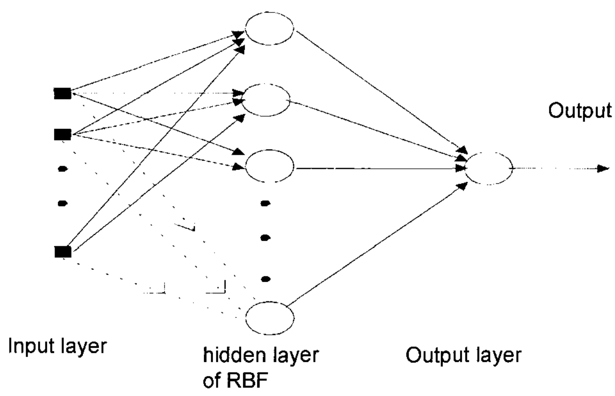

Fig. 2. Radial basis function network.

The second approach that we used in this study is the so called Radial Basis Function (RBF) network. The RBF network used in this work is given in Fig. 2. It consists of an input layer, one hidden layer and an output layer.

The transformation from input space to output space is nonlinear and the transformation from hidden unit space to output space is linear (Simon Haykin). The set of basis functions $\{\Phi_i(x) \mid i = 1,2,.........,M\}$ is defined as follows

$$\Phi_i(x) = G(\| x - t_i \|) = \exp(-\| x - t_i \|), \quad i = 1,2,3,...........,M \tag{9}$$

where $\{t_i \mid i = 1,2,...........,M\}$ is the set of M centers to be determined and x is the one of the training (input) data in a set $\{x_i \mid i = 1,2,...........,M\}$ of size N. Typically, the number of basis functions is less than the number of data points (i.e. $M \le N$). Our aim is to find the suitable w values in order to minimize the Euclidean norm.

$\| d - Gw \|^2$, where $d = [d_1, d_2, ..., d_N]^T$

$$G = \begin{bmatrix} G(\| x_1 - t_1 \|) & G(\| x_1 - t_2 \|) & \cdots & \cdots & G(\| x_1 - t_M \|) \\ G(\| x_2 - t_1 \|) & G(\| x_2 - t_2 \|) & \cdots & \cdots & G(\| x_2 - t_M \|) \\ \vdots & \vdots & & & \vdots \\ \vdots & \vdots & & & \vdots \\ G(\| x_N - t_1 \|) & G(\| x_N - t_2 \|) & \cdots & \cdots & G(\| x_N - t_M \|) \end{bmatrix} \tag{10}$$

and $w = [w_1, w_2, ..., w_M]^T$.

The vector d is an N-dimensional desired response vector, the matrix G is a NxM matrix of Green's functions and the vector w is M-by-1 weight vector for the linear transformation from hidden unit space to output space. The minimum norm solution to the over-determined least squares data fitting problem can be given as follows.

$$w = (G^T G)^{-1} G^T d \tag{11}$$

The set of centers $\{t_i \mid t = 1,2,........,M\}$ can be selected randomly from the set of data points, and can be selected using the clustering techniques to find the suitable centers or can be selected using gradient descent algorithm. In this study, we used random selection and also k-means clustering algorithm to find the set of centers for the radial basis functions.

3.1. Experimental Results and Discussions

For this study, we took 450 numerical data for input and output parameters of a spray dryer which produce powder for a porcelain production factory in Turkey. All these data were taken during 8 months period of observation in the factory. The input parameters related data are collected during the production for every hour and then in order to determine the moisture content for this duration, the samples are taken from the finished product and measured in the laboratory by staff.

In this study, our aim is to predict the moisture content which is the most effective product parameter for the spray drying process. We used two different artificial neural network approaches, namely the back-propagation multilayer perceptron algorithm and the radial-basis function network in order to find a non-linear relation between the moisture content and the six input parameters given as; the inlet temperature ($x1$), outlet temperature ($x2$), the feed viscosity ($x3$), the pressure of the production system ($x4$), the amount of chemicals used ($x5$) and also the solid content of the slurry ($x6$) in spray drying process. In this study, we used 450 data for training and testing processes. Each data consist of 7 values, including the 6 input parameters given above and an output value that shows the moisture content (x out) of the system. These 7 parameters are normalized using the formulas given below.

$$x1_{nor} = (x1 - 300)/60,\ x3_{nor} = (x3 - 10)/10,\ x5_{nor} = (x5 - 80)/15$$

$$x2_{nor} = (x2 - 90)/10,\ x4_{nor} = (x4 - 10)/10,\ x6_{nor} = (x6 - 1550)/40,\ xout_{nor} = xout/5 \quad (12)$$

This normalization is performed in order to get rid of the great difference between the values of these 6 parameters and the output is normalized in order to make it in the range [0,1]. However, in the calculation of training errors and testing errors using equation 1 and equation 2, we multiplied the predicted values by 5 and compared them with the desired values. Among these 450 data obtained at different time intervals, 350 of them are randomly selected for training and the rest are used for testing.

The first approach is to model the system at hand using BPMLP algorithm. As we mentioned in the model description part, there are some system parameters that are not known and should be predicted. These system parameters affect the effectiveness and convergence of the back propagation learning algorithm. Among them is the learning constant η, the momentum parameter α, and the number of hidden neurons in the first layer n_1 and in the second hidden layer n_2. It is known that, there is no single learning constant value suitable for different training cases and hence η is usually chosen experimentally for different problem. A larger value of η could speed up the convergence but might in result in overshooting while a smaller value of η has a complementary effect. The other system parameter is the momentum parameter α which is mainly helping to speed up the convergence and to achieve an efficient and more reliable learning profile. The momentum parameter α is in the range of [0, 1] and usually a value of 0.9 is used for this parameter.

In real-world problems, the fundamental question is raised about the size of the hidden layers. The exact analysis of this issue is rather difficult because of the complexity of the network. Hence the parameters n_1 and n_2 are usually determined experimentally. In order to select the parameters given above, we performed an experiment. For this experiment, the momentum parameter i was fixed to 0.9 and the weights are initialized as small random values. In Fig. 3, we have shown the total training error for different learning constant values ($\eta = 0.1,0.15,0.2,0.5$), for different number of neurons in the first hidden layer ($n_1 = 5,7,9$) and for different number of neurons in the second hidden layer ($n_2 = 3,5,7$).

From the Fig. 3, we deduced that the best prediction can be accomplished by selecting the learning constant value equal to 0.1 and the number of neurons in the first and second hidden layer as 7 and 3 respectively. Using these selected parameters, we have plotted the testing error for the 100 test data in Fig. 4. From this figure one can easily observe that almost all the predicted values for the test data are very close to the actual values. The total error for the 100 test data is equal to 0.8.

The second approach is to model the system using radial basis function network. In this approach, the number of basis functions M and also how the initial set of centers $\{t_i \mid i = 1,2,\ldots\ldots\ldots,M\}$ will be selected are very important for the performance of the system. To see the effect of M and also the effect of selecting of t_i values, we performed an

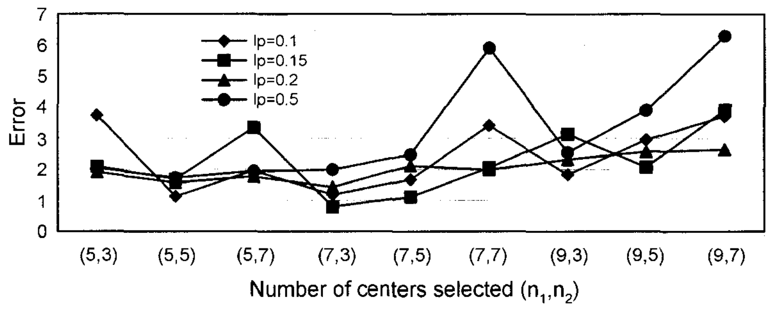

Fig. 3. Total error for different number of centers and for different lp parameters.

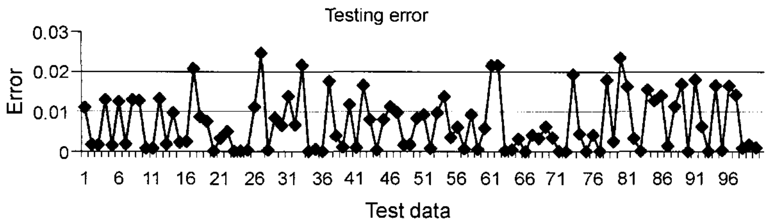

Fig. 4. Error for the test data using BPMLP algorithm.

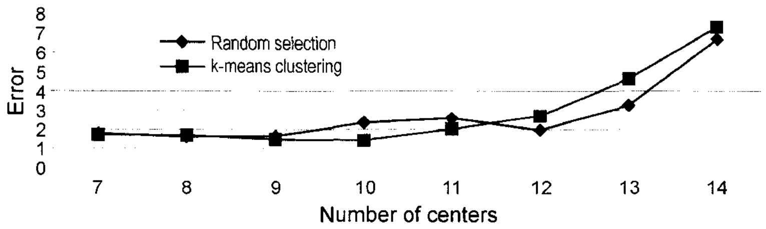

Fig. 5. Total error for different number of centers for RBF.

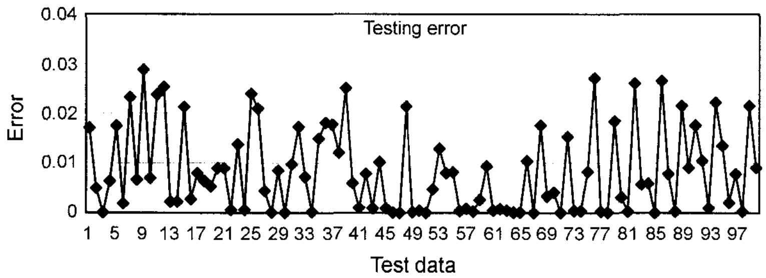

Fig. 6. Error for the test data using RBF network.

experiment using different number of basis functions (M = 7,8,9,....,14) and using two different selection methods, namely random selection and k-means clustering method. The results are shown in Fig. 5. From the results, we conclude that the selection of t_i's through the use of k-means clustering method outperforms the random selection. Also we observed that 9 – 10 basis functions are enough to model the system at hand.

Hence by selecting a value of 10 for the M parameter and by using the k-means clustering algorithm for the selection of centers, we obtained the weights of the RBF network using the 350 training data and using these weights, we plotted the error for the same 100 test data as in the previous approach in Fig. 6. We have again observed that almost all the predicted values are very close to the actual values and we calculated that the total testing error is equal to 0.7.

From the testing errors plotted in Figs. 4 and in 6, we selected 10 samples randomly and tabulated them in Table 1 in order to compare the performances of these two approaches.

From Figs. 4 and 6, we observed that the total error for testing data obtained using RBF network is less than the one obtained using BPMLP algorithm. Table 1 confirms this fact on the basis of single test data.

Table 1. Comparison of Test Data with the Actual Process Data

Actual data	BPMLP	RBF
2.2	2.229	2.141
2.3	2.285	2.243
2.4	2.467	2.339
2.5	2.426	2.43
2.5	2.48	2.486
2.4	2.372	2.29
2.4	2.431	2.38
2.4	2.554	2.4
2.3	2.488	2.22
2.5	2.41	2.417
2.5	2.32	2.5
2.7	2.725	2.689
2.3	2.435	2.3
2.2	2.245	2.187
2.5	2.293	2.442

4. Conclusions

Spray dryers have been used into all major industries ranging from advanced ceramics, to the food and pharmaceutical manufacture, and porcelain manufacture which requires high tonnage outputs in the heavy chemical fields. As it was mentioned, the critical control parameters of the plant was observed and 450 data were taken for each control parameters, for training and testing of the ANNs approaches. From the set, 350 of them were used for training of the model and the rest were used for testing. It is known that in this processes, the relation between the moisture content and the input parameters are highly non-linear. From ANN models, we concluded the following:

The moisture content of the product obtained from the spray drying process can be effectively estimated using both of the ANN approaches.

In the BPMLP algorithm, the selection of system parameters namely; learning the constants, the number of hidden neurons in the first layer n_1 and in the second hidden layer n_2, are very important and the total testing error varies for different parameter selection. RBF network seems to be more robust to parameter changes than the BPMLP algorithm. The total error for 100 testing data using RBF network is approximately equal to 0.7. The average error for a single testing data is equal to 0.007 and the average actual output value is approximately equal to 2.45. Hence, using equation 1, one can conclude that the average percentage error for each testing data is approximately 4.8% which is an acceptable error. From the results, we deduced that the moisture content of the resulting dried product can be estimated with an acceptable error using artificial neural network approaches. It is achieved that, ANN approaches can be used for modeling this powder production processes We used C++ programming language for writing the programs of two ANN modeling approaches.

REFERENCES

1. K. Masters, "Spray Drying Handbook," *ISBN* No: 0582062667, Published in Singapore, 1991.
2. S. J. Lukasiewicz, "Granulation and Spray Drying," *Ceram. Eng. Handbook*, 100-08 (1995).
3. E. Negre and E. Sanchez, "Advances in Spray-Dryed Powder Processing for Tile Manufacture," *Sci. of Whitewares, the Am. Ceram. Soc.*, Ohio, 1996.
4. J. S. Reed, "Spray Drying and Implications for Compability of Product Granules," *Ceram. Eng. Sci. Proc.*, **18** [2] (1997).
5. O. Taylan and H. Taskin, "Fuzzy Modeling of a Production System," *J. of Naval Sci. and Eng.*, **1** [1] 1-13 (2003).
6. O. Taylan, A. Göleç, and H. Taskin, "Intelligent Control of a Porcelain Factory Subsystem," *Proceedings of 2nd International Symposium on Intelligent Manufacturing Systems*, pp. 651-62, Sakarya Uni., August 6, 1998.
7. C.-T. Lin and C. S. George Lee, "Neural Fuzzy Systems," *Prentice Hall*, 1996.
8. A. Cichocki and R. Unbehauen, "Neural Networks for Optimization and Signal Processing," John Wiley & Sons, 1993.
9. S. Haykin, "Neural Networks," Macmillan College Publishing Company, 1994.