ELSEVIER

International Journal of Pharmaceutics 108 (1994) 155–164

international journal of pharmaceutics

The advantages by the use of neural networks in modelling the fluidized bed granulation process

Ere Murtoniemi [a,*], Jouko Yliruusi [a], Pirjo Kinnunen [b], Pasi Merkku [a], Kauko Leiviskä [b]

[a] *Pharmaceutical Technology Division, University of Helsinki, P.O. Box 15, FIN-00014 University of Helsinki, Finland*
[b] *Control Engineering Laboratory, Department of Process Engineering, University of Oulu, Linnanmaa, FIN-90570 Oulu, Finland*

(Received 1 October 1993; Modified version received 31 January 1994; Accepted 14 February 1994)

Abstract

The use of artificial neural networks (ANNs) in modelling a fluidized bed granulation process is reported. The granules were made in a fully instrumented laboratory-scale granulator (Glatt WSG 5, Glatt GmbH, Germany). The independent input variables were inlet air temperature, atomizing air pressure and binder solution amount. The input variables varied in three levels. The responses used were mean granule size and granule friability. Neural computing was carried out using a commercial NeuDesk software (Neural Computer Sciences, U.K.) in a 486 microcomputer with a specific accelerator card, NeuSprint (Neural Computer Sciences, U.K.). In total, 36 different ANN models were tested. The results were also compared with a statistical method (multilinear stepwise regression analysis). The results showed clearly that the best networks were able to predict the experimental responses more accurately than the multilinear stepwise regression analysis. On the other hand, it also became evident that several different structures should be trained with different training end points to generate a proper model.

Key words: Artificial neural network; Multilayer feedforward network; Neural computing; Process modelling; Fluidized bed granulation; Multilinear stepwise regression analysis

1. Introduction

The use of artificial intelligence and artificial neural networks (ANN) is a very rapidly developing field in many different areas of science and technology (Nukhodhayay and Narendra, 1993; Savkovic-Stavanovic, 1993), but the use of ANNs in pharmaceutical technology has been rather limited (Hussain et al., 1991). The application of ANNs can be divided into two main categories: classification and modelling.

Examples of classification problems are, for instance, insurance risk evaluation, marketing analysis, signal processing, machinery defect diagnosis, process supervision and general diagnosis (Yoon et al., 1993). The modelling problems can handle, e.g., the following types of systems: prediction of economic indicators, robot control and navigation, process control, aircraft landing control, prediction of performance of drugs from the

* Corresponding author. Tel: +358-0-191 2765; Fax: +358-0-191 2786.

SSDI 0378-5173(94)00057-C

molecular structure and weather forecasts (Gill and Shutt, 1992).

Fluidized bed granulation is a widely used and important process in the pharmaceutical industry, since it has several advantages (Aulton and Banks, 1981; Jones 1985; Kristensen and Schæfer, 1987). However, the process itself is very complicated and the modelling of the process requires sophisticated instrumentation (Banks 1981; Niskanen et al., 1990; Merkku et al., 1992a,b). Furthermore, the study designs and the frequently used statistical analyses, e.g., multilinear stepwise regression, require systematic approaches. This methodology in some respects may still be rather limited. The rapid development of microcomputers and software, especially in the field of neural computing, appears to provide an interesting and, evidently, effective method of modelling complicated multivariate production processes such as fluidized bed granulation. ANN analysis is quite flexible concerning the amount and form of the training data which make it possible to use more informal experimental designs than with statistical approaches. Authors also presume that neural network models might generalize better than regression models generated with the multilinear stepwise technique, since regression analyses are dependent on predetermined statistical significance levels. This means that less significant terms are not included in the models. The application of artificial neural networks is a totally different method in which all possible data are used for making the models more accurate.

In this study, the use of ANNs in modelling the fluidized bed granulation process is introduced, with a short introduction to neural computing. The results obtained on the basis of different trained ANNs are compared with those calculated using multilinear stepwise regression.

2. Theory

2.1. Neural computing

The principle of artificial neural networks (ANNs) simulates the structure and function of the human brain. Biological neural systems consist of nerve cells connected by synaptic junctions. The synaptic junctions (synapses) transmit nerve impulses from one neuron to another. The effect of impulses can be either excitatory or inhibitory in the receiving nerve cell.

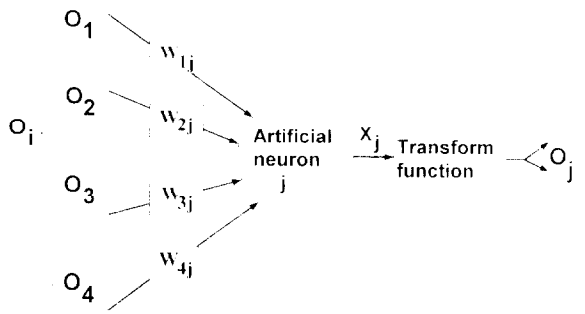

Fig. 1. An artificial neuron (O_j, input of a neuron; O_i, output of a neuron; i, index of the neuron that feeds neuron j; w_{ij}, weight of the connection).

ANNs consist of simple processing elements (Fig. 1) which simulate biological nerve cells (neurons), and interconnections between the elements. The interconnections act like the synaptic junctions. According to Freeman and Skapura (1991), it is not always appropriate to condiser that the processing elements in a neural network are in a one-to-one relationship with actual biological nerve cells. It is sometimes better to imagine a single processing element as representative of the collective activity of a group of nerve cells.

In this paper, the terms neuron and connection are used for processing elements (artificial neurons) and interconnections between them. A neuron may have several input values, but only one output value which, however, may be copied and sent to other neurons in the network. Fig. 1 depicts an example of an individual neuron (j), which has four inputs ($O_1,\ldots,O_4$). The output value is copied, and the neuron has two outputs (O_j). Four preceding neurons are connected to the neuron (j) and the symbols $W_{1j},\ldots,W_{4j}$ represent the weight values of the connections.

The output (O_j) of an individual neuron (Fig. 1) is calculated by summing up the input values (O_i) multiplied by their corresponding weights (W_{ij}) (Eq. 1) and converting the sum (X_j) to output (O_j) by a transform function (Eq. 2). The

most common transform function is a sigmoidal function (Dayhoff, 1990; Davalo, 1991) where O is the output of a neuron, i denotes the index of the neuron that feeds neuron (j), and W_{ij} is the weight of the connection.

$$x_j = \sum_i O_i \cdot W_{ij} \tag{1}$$

$$O_j = \frac{1}{1 + e^{-x_j}} \tag{2}$$

In an ANN the neurons are usually organised in layers. There is always one input and one output layer. Furthermore, the network also usually contains at least one hidden layer. The use of hidden layers also confers to ANNs the ability to describe nonlinear systems (Dayhoff, 1990). In neural computing, applications are generated by training the network with proper data, which does not require traditional programming. During the beginning of the training every connection between the neurons has a random weight value. The signal will either increase or decrease passing through the connection. The number of connections may easily be large even in relatively small networks. Fig. 2 shows an example where two hidden layers in an ANN result in 84 connections.

An ANN attempts to learn the relationships between the input and output data sets in the following way: During the training phase, input/output data pairs, called training data, are introduced into the neural network. The difference between the actual output values of the network and the training output values is then calculated. The difference is an error value which is decreased during the training by modifying the weight values of the connections. Training is continued iteratively until the error value has reached the predetermined training end point. The training end point is used for the termination of the training. In this study, training was terminated when the the error value for each input/output data pair in the training data reached the predetermined error level. This is referred to as the training end point. If a neural network is used for process modelling, the information of the process

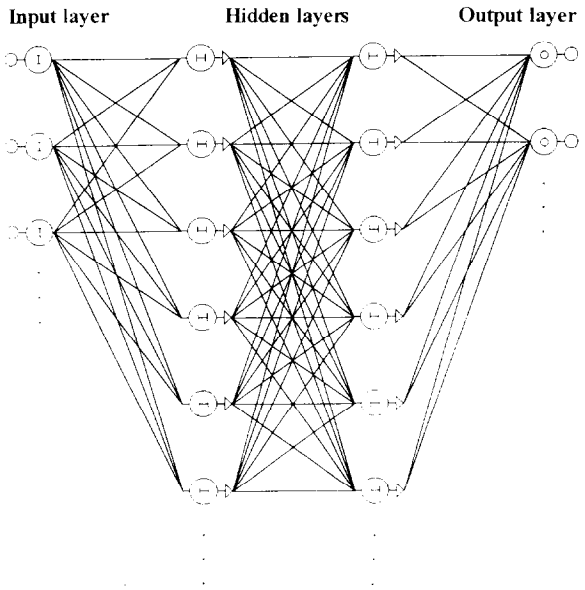

Fig. 2. Typical architecture of a multilayer fully connected neural network (I, input neuron; H, hidden neuron; O, output neuron).

behavior is transformed to a weight matrix during training.

There are several algorithms available for training neural networks (Lisboa, 1992). One quite commonly used algorithm is the back propagation which is a supervised learning algorithm (both input and output data pairs are used in the training). It is also called the generalized delta rule (GDR) and is commonly used in many applications (Bhatt et al., 1990; Beale and Demuth, 1992; Lodewyck and Deng, 1993). The output layer weights are updated by Eq. 3

$$W_{ij}^0(n+1) = W_{ij}^0(n) + \eta\delta_j^0 O_i + \alpha\left(W_{ij}^0(n) - W_{ij}^0(n+1)\right) \tag{3}$$

and the hidden layer weights by Eq. 4

$$W_{ij}^h(n+1) = W_{ij}^0(n) + \eta\delta_j^h O_i + \alpha\left(W_{ij}^0(n) - W_{ij}^0(n+1)\right) \tag{4}$$

where n is the number of the iteration epoch, α respresents the momentum factor and η is the learning rate. If the learning rate is too high, the weights alter dramatically in each iteration epoch

and the network output does not converge on the training end point, but the output error varies randomly. On the other hand, an excessively low learning rate increases the training time (Freeman and Skapura, 1991).

In Eq. 3, δ_j^0 is the error term for the output layer neurons and is calculated according to the following formula:

$$\delta_j^0 = f'(x_j^0)(T_j^0 - O_j^0) \quad (5)$$

where T_j^0 is the training output value and O_j^0 denotes the actual output value of the network. In Eq. 4, δ_j^h is the error term for the hidden layer neurons which is calculated as follows:

$$\delta_j^h = f'(x_j^h) \sum_i \delta_j^0 W_{ij}^0 \quad (6)$$

After training the network may be consulted and response values generated using new input data.

3. Materials and methods

3.1. Study design

In previous studies (Merkku and Yliruusi, 1993; Merkku et al., 1993, 1994), a complete 3^3 factorial experimental design was used in order to evaluate the effects of three critical fluidized bed granulation variables, i.e., inlet air temperature (T), atomizing air pressure (p) and binder solution amount (m), on granule properties. The inlet air temperatures were 40, 50 and 60°C, the atomizing air pressures 1.0, 1.5 and 2.0 bar, and the binder solution amounts 150, 300 and 450 g. The factor levels used in the production of granules are listed in Table 1. A total of 38 batches, consisting of 27 experimental and 11 replicate batches, were produced.

3.2. Materials

The filler material was 80 mesh lactose α-monohydrate (DMV, Veghel, The Netherlands). 2% of anhydrous theophylline (Ph. Eur.) was added as a marker in each batch to be granulated. The 3 kg batches were granulated using a 20% water dispersion of polyvinylpyrrolidone (Kollidon® K25, BASF, Germany).

Table 1
Factor levels used in the production of granules and the measured mean granule sizes and granule friabilities (T, inlet air temperature; p, atomizing air pressure; m, binder solution amount)

T (°C)	p (bar)	m (g)	Granule size (μm) ($x \pm \epsilon$) [a]	Friability (%) ($x \pm \epsilon$) [a]
40	1.0	300	475 ± 10	18.6 ± 2.4
40	1.0	450	516 ± 18	5.6 ± 0.9
40	1.5	150	354 ± 1	36.9 ± 3.7
40	1.5	300	355 ± 12	16.1 ± 2.5
40	1.5	450	439 ± 4	8.3 ± 1.2
40	2.0	150	305 ± 8	45.3 ± 1.4
40	2.0	300	328 ± 6	37.4 ± 3.2
40	2.0	450	356 ± 6	23.3 ± 1.7
50	1.0	150	362 ± 5	24.2 ± 2.4
50	1.0	300	421 ± 11	11.8 ± 1.6
50	1.0	450	405 ± 13	8.8 ± 3.1
50	1.5	150	296 ± 10	37.4 ± 2.1
50	1.5	300	344 ± 5	44.2 ± 2.1
50	1.5	450	327 ± 6	24.2 ± 1.2
50	2.0	150	321 ± 5	47.5 ± 3.3
50	2.0	300	355 ± 6	29.0 ± 2.3
50	2.0	450	323 ± 2	24.2 ± 3.0
60	1.0	150	368 ± 4	14.1 ± 3.8
60	1.0	300	433 ± 8	14.1 ± 4.7
60	1.0	450	465 ± 8	13.0 ± 1.0
60	1.5	150	315 ± 11	37.8 ± 0.9
60	1.5	300	316 ± 9	31.3 ± 2.0
60	1.5	450	431 ± 14	20.9 ± 2.9
60	2.0	150	286 ± 5	64.8 ± 3.0
60	2.0	300	356 ± 5	38.2 ± 3.3
60	2.0	450	367 ± 2	31.6 ± 3.6

[a] x is the mean and ϵ denotes the maximum error estimate defined as $\frac{1}{2}(\max - \min)$ ($n = 3$).

3.3. Preparation of granules

The granules were produced in an automated laboratory-scale fluidized bed granulator (Glatt WSG 5, Glatt GmbH, Germany). The process automation system is based on a commercial control software package (Monitor 77, Telemecanique, France). The system controls the inlet air temperature, air flow rate, atomizing air pressure and binder solution pump unit. The granulation conditions have been described previously by Merkku and Yliruusi (1993).

3.4. Mean granule size

Mean granule sizes were measured by laser diffractometry (Malvern 2600C Droplet and Particle Sizer, Malvern Instruments, U.K.) using a dry powder feeder. The focal length of the lens was 1000 mm. The mean granule sizes were calculated as averages of three parallel measurements.

3.5. Granule friability

Friability of the granules was determined by weighing 20 g of granules and 20 g of glass beads into a 100 ml container. Before analysis the granules were sieved through a 355 μm sieve and the oversize fraction was used in the analysis. The granule sample was mixed with the glass beads for 5 min in a Turbula mixer (System Schatz, W.A. Bachofen, Switzerland) after which they were sieved again through a 355 μm sieve. The friability of the granules was calculated as the loss of weight (%). Friability determinations were performed in triplicate.

4. Data analysis

4.1. Multilinear stepwise regression analysis

In earlier studies (Merkku and Yliruusi, 1993; Merkku et al., 1993), regression models were developed for the mean granule size (Y_1) and granule friability (Y_2) using multilinear stepwise regression (Eqs. 7 and 8). In those studies only significant terms ($p < 0.05$) were included. The squared multiple R for mean granule size was 0.801 and for granule friability 0.717. The models were generated using SYSTAT v. 5.0 (SYSTAT Inc., U.S.A.). The regression models had the following forms:

(a) Mean granule size:

$$Y_1 = 27.70\ T^2 - 18.78\ pm - 44.84\ p + 42.52\ m + 348.8 \quad (7)$$

(b) Granule friability:

$$Y_2 = 3.304\ T + 11.37\ p - 9.44\ m + 29.22 \quad (8)$$

where T is the inlet air temperature (°C), p denotes the atomizing air pressure (bar) and m is the binder solution amount (g).

4.2. ANN simulator software

A commercial MS-Windows based artificial neural network simulator software, NeuDesk V.2.10 (Neural Computer Sciences, U.K.), was used. Calculations were performed using a 486 personal computer with a special accelerator card, NeuSprint (Neural Computer Sciences, U.K.) which increased the processing speed approx. 100-fold.

4.3. Training data

The input (granulation variables) and output (granule properties) factor levels presented in Table 1 were used as the training data. In the replicate experimental points, average values for the granule properties were calculated and used in the training. All variables were converted to values between '0' and '1' with 10% headroom after which the sigmoidal function (Eq. 2) operates in the more linear region. The network output cannot reach the values 0 or 1 with the sigmoidal transform function, since Eq. 2 approaches the values 1 and 0 asymptotically. Each value of the training data is converted with the scaling values calculated according to Eqs. 9 and 10 as follows:

$$\mathrm{Ls} = \mathrm{Min} - 0.1(\mathrm{Max} - \mathrm{Min}) \quad (9)$$

$$\mathrm{Us} = \mathrm{Max} + 0.1(\mathrm{Max} - \mathrm{Min}) \quad (10)$$

where Ls and Us are the lower and upper scaling values, respectively. Max and Min are the maximum and minimum data values in the training data, respectively. The converted value for an individual data item is calculated using Eq. 11:

$$\mathrm{ConvertedValue} = (\mathrm{Value} - \mathrm{Ls})/(\mathrm{Us} - \mathrm{Ls}) \quad (11)$$

where Value is an individual data item of the training data.

Table 2
Factor levels and the measured mean granule sizes and granule friabilities used for the generalization ability test (T, inlet air temperature; p, atomizing air pressure; m, binder solution amount)

T (°C)	p (bar)	m (g)	Granule size (μm)	Friability (%)
45	1.8	225	409	42.3
55	1.8	225	396	43.3
45	1.3	375	530	12.8
50	1.4	200	344	32.7
50	1.7	410	403	25.4

4.4. Test data

In studying the generalization ability of neural networks, a number of further granulation experiments were performed (Table 2). In the experimental points, the factor levels of input variables were chosen so that they were within the range of the original training data (interpolation). The generalization ability was studied by consulting the network with test data and observing the output values. The output values are predicted granule properties. This operation is called interrogation or querying.

Average error percentage is used for examination of the best generalization ability of neural networks (the smallest average error percentage). The average error percentage is calculated according to Eq. 12:

$$\mathrm{ERR}\% = \sum_{i=1}^{n} \left[\mathrm{abs}\left(1 - \left(\frac{\mathrm{ANN}_i}{\left(\sum_{j=1}^{m} M_{ij} \right)/m} \right) \right) \right] \times 100\%/n \qquad (12)$$

where n is the number of experimental points, m denotes the number of replicate experiments, M_{ij} is the measured granule property value in the experimental point i (replication j), and ANN_i represents the granule property value in the experimental point i predicted by the neural network.

5. Results and discussion

5.1. Network topologies

The properties of the training data naturally determine the number of input and output neurons. In this study, the number of factors (inlet air temperature, atomizing air pressure and binder solution amount) forced the number of input neurons to be three, and the number of granule properties included (mean granule size and granule friability) forced the number output neurons to be two. Only quite simple and small networks were studied because one should never use more hidden layer neurons than training samples. In fact, the number of hidden layer neurons should be much lower than the number of training samples. Otherwise the network simply 'memorizes' the training samples, resulting in poor generalization (Hush and Horne, 1993).

The tested ANNs had either one or two hidden layers. All networks were fully connected multilayer networks, which means that each neuron is connected to every neuron in the layer below and in the layer above. The neurons are not connected to other neurons in the same layer. This structure is commonly used with the back propagation training algorithm. Similar small networks have been used in previous reports (Hussain et al., 1991; Savkovic-Stavanovic, 1993), however, greater networks have also been used with large data sets (Le Cun et al., 1990). The amount of the training data here is relatively small and it is reasonable to focus on simple networks. The number of neurons in the hidden layer ranged from 3 to 15. The lower limit is mainly dependent on the number of input neurons, and the upper limit was fixed experimentally during the study when the effects of the different networks were observed. Neurons were added to the hidden layer one at a time. The networks were trained and tested after each addition. After addition of the 15th hidden neuron, it became evident that more hidden layer neurons did not improve the generalization ability of the network. This technique is quite usual in practice, since there is no generally acceptable theory for the limitation of neurons in the number of hidden layers even if

some limited rules of thumb are presented (Padgett and Ruppel, 1992; Hush and Horne, 1993).

5.2. Training of the networks

Networks were trained using different training end points. The maximum error of the network output was chosen and compared with the training end point after each iteration epoch. If the maximum error value converged under the predetermined training end point, the iteration was terminated. In this study all networks tested converged to the desired error level. The training end point had an influence such that at the first the generalization ability became better, however, if the training is continued for too long, extra training makes the generalization ability worse. This effect is called overtraining. The first training end point (0.1) was chosen on the basis of a short convergence time (a few seconds). Later, the training end point was taken more accurately until it became evident that the network lost its generalization ability.

The generalization ability of the neural network models was analyzed by interrogating the networks with the test data (Table 2). The best networks were chosen on the basis of the average error percentage. The training algorithm was a modified back propagation algorithm designated as the Weigend weight eliminator (WWE) (Weigend et al., 1992). During normal (back propagation) training, WWE tends to decrease the less important and unwanted weight values close to zero.

Tables 3 and 4 represent the average error percentages for mean granule size and granule friability as a function of network topology and training end point. The best generalization ability for the mean granule size was achieved with a network containing 11 neurons in one hidden layer. For granule friability, the best network had 12 hidden neurons in one layer. In both cases, the training end point was 0.1. The network seems to lose its generalization ability easily when it is overtrained.

In every case, the training end point 0.07 provided less accurate results than that of 0.1. If the training end point is 0.05 or 0.03, the average

Table 3
Average error percentages of mean granule size using different numbers of neurons in the hidden layer and different training end points

Number of neurons in the hidden layers		Training end point			
		0.10	0.07	0.05	0.03
Layer 2	Layer 1				
0	3	17.28	–	–	–
0	4	17.37	–	–	–
0	5	15.19	–	–	–
0	6	14.80	18.90	–	–
0	7	15.12	15.90	16.72	–
0	8	14.92	18.96	19.24	20.46
0	9	15.15	18.83	19.05	19.72
0	10	14.93	18.12	19.60	16.77
0	11	14.71	18.69	18.35	18.13
0	12	14.87	18.14	–	–
0	13	15.10	18.86	–	–
0	14	15.07	18.21	–	–
0	15	14.89	19.07	–	–
4	4	14.95	–	–	–
5	5	15.19	–	–	–
6	6	14.92	–	–	–
7	7	14.77	–	–	–

Table 4
Average error percentages of granule friability using different numbers of neurons in the hidden layer and different training end points

Number of neurons in the hidden layers		Training end point			
		0.10	0.07	0.05	0.03
Layer 2	Layer 1				
0	3	21.11	–	–	–
0	4	13.89	–	–	–
0	5	8.80	–	–	–
0	6	8.33	14.31	–	–
0	7	8.14	10.26	21.83	–
0	8	8.09	13.74	20.93	25.46
0	9	8.15	8.85	22.09	26.96
0	10	7.76	10.83	22.44	29.97
0	11	7.81	12.71	20.55	24.92
0	12	7.75	8.48	–	–
0	13	7.80	10.90	–	–
0	14	7.78	11.47	–	–
0	15	7.89	11.71	–	–
4	4	8.81	–	–	–
5	5	8.76	–		
6	6	8.27	–	–	–
7	7	8.07	–	–	–

error percentage increases rapidly and the network loses its generalization ability almost totally. In practice this means that the networks are overtrained.

However, the number of hidden neurons does not appear to influence markedly the average error percentage, except when the networks are very small (three or four neurons in one hidden layer). Using three or four hidden neurons the average error percentages are higher (Tables 3 and 4) when compared to networks that contain five or more hidden neurons. In such cases, the average error percentages are quite close to each other. It is evident that a greater number of hidden neurons does not automatically improve the generalization ability of the network.

In this specific study, the training phase did not take much computational time to achieve proper weights because the amount of training data was relatively small. The training of an individual network with this data set takes only a few seconds. The situation might change dramatically if the amount of learning data increases.

5.3. Comparison of neural network and regression model

In Fig. 3, experimental mean granule sizes are compared with the sizes predicted by the neural network (ANN) and with those calculated by the regression model (REG) (Eq. 7). The ANN sizes are closer to the experimental values than the REG sizes. It is interesting to observe that both methods seem to underestimate the experimental values.

In Fig. 4, the friability of granules is presented according to the same principles as above. Also, the friabilities are predicted more accurately by the ANN. In this case, the underestimation of the modelling methods is no longer so apparent. Furthermore, the estimations are now more accurate. This is expected because the measuring accuracy of friability is much higher than that of mean granule size.

Predicted granule properties (Fig. 3 and 4) in the case of the ANN were closer to the experimental values as compared to the regression model. This effect was found systematically in each experimental point although the differences were not marked. The average error percentage (Eq. 9) according to the regression model was 16.6% for mean granule size and 11.4% for granule friability. The corresponding values according to the neural model were 14.7 and 7.8%, demonstrating the ability of the neural model to predict granule properties more accurately than the regression model. On the other hand, it is not possible to know beforehand which network topology and training end point will have the best generalization ability. Much work will be needed in order to determine the proper topology and training end point and/or a large amount of training data.

The experiments were designed primarily in order to test and to create regression models (Merkku and Yliruusi, 1993). Here, the same data

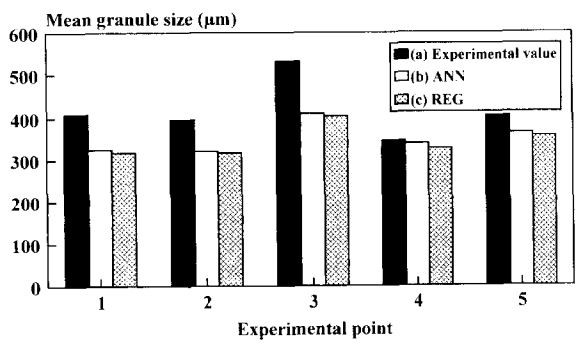

Fig. 3. Mean granule sizes: (a) experimental, (b) artificial neural network estimated (ANN) and (c) regression model estimated (REG).

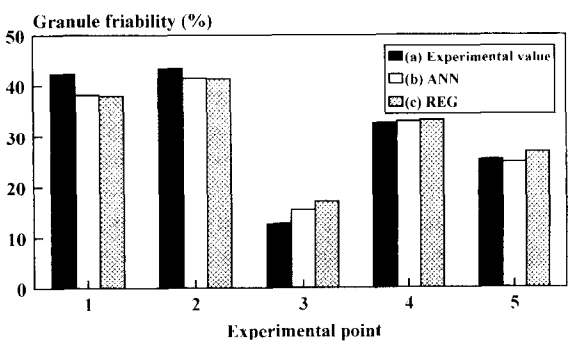

Fig. 4. Granule friability (%): (a) experimental, (b) artificial neural network estimated (ANN) and (c) regression model estimated (REG).

were used for training neural networks. In general, ANN analysis is not limited to a specific number of experiments which is often the case in statistical approaches. Some other type of experimental design may be more effective for neural computing, nevertheless, it was possible with these training data to create neural models which were more accurate than the regression models. Based on the above, it is possible to claim that artificial neural networks are promising methods for modelling pharmaceutical fluidized bed granulation processes.

The quality of the training data plays a very important role in modelling applications by neural computing (Stein, 1993a,b). Collection and reprocessing of the training data should be studied more closely in the future. At present, a large amount of data can readily be collected from instrumented industrial processes. By using such data as the training data of a neural network, it is in theory possible to model more accurately complicated non-linear pharmaceutical processes.

Acknowledgements

This study was supported by the Technology Development Centre in Finland (TEKES) and the Finnish pharmaceutical industry (Leiras Oy and Orion Corp., Orion-Farmos Pharmaceuticals). The authors wish to thank Jaana-Liisa Aro, B.Sc. (Pharm.), for technical help and assistance.

References

Aulton, M.E. and Banks, M., Fluidised bed granulation – factors influencing the quality of the product. *Int. J. Pharm. Tech. Prod. Mfr.*, 2(4) (1981) 24–29.

Banks, M., Studies on the fluidized bed granulation process. Thesis, Leicester Polytechnic, Leicester (1981).

Beale, M. and Demuth, H., Building a neutral network application. *Sci. Comput. Automation*, 5 (1992) 18–21.

Bhat, N.V., Minderman, P.A., Jr, McAvoy, T. and Sun Wang, N., Modelling chemical process systems via neural compution. *IEEE Control Systems Mag.*, 4 (1990) 24–29.

Davalo, E. and Naïm, P., *Neural Networks*, Macmillan, London, 1991, pp. 19–24.

Dayhoff, J.E., *Neural Network Architectures An Introduction*, Van Nostrand Reinhold, New York, 1990, pp. 58–66.

Freeman, J.A. and Skapura, D.M., *Neural Network Algorithms, Applications and Programming Techniques*, Addison-Wesley, Houston, 1991, pp. 12–105.

Gill, T. and Shutt, J., Optimizing product formulations using neural networks. *Sci. Comput. Automation*, 9 (1992) 18–26.

Hush, D. and Horne, B.G., Progress in supervised neural networks. *IEEE Signal Processing Mag.*, 10 (1993) 8–39.

Hussain, A.S., Xuanqiang, Y. and Johnson, R.D., Application of neural computing in pharmaceutical product development. *Pharm. Res.*, 8 (1991) 1248–1252.

Jones, D.M., Factors to consider in fluid-bed processing. *Pharm. Technol.*, 4 (1985) 50–62.

Kristensen, H.G. and Schæfer, T., Granulation. A review on pharmaceutical wet-granulation. *Drug Dev. Ind. Pharm.*, 13 (1987) 803–872.

Le Cun, Y., Boser, B., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W. and Jackel, L.D., Handwritten digit recognition with a backpropagation network. *Advantages Neural Information Processing Systems*, 2 (1990) 396–404.

Lisbon, B.G.J., *Neural Network Current Applications*, Chapman & Hall, London, 1992, pp. 9–34.

Lodewyck, R.W. and Deng, P.S., Experimentation with a back-propagation neural network. *Information Management*, 24 (1993) 1–8.

Merkku, P. and Yliruusi, J., Use of 3^3 factorial design and multilinear stepwise regression analysis in studying the fluidized bed granulation process: I. *Eur. J. Pharm. Biopharm.*, 39 (1993) 75–81.

Merkku, P., Antikainen, O. and Yliruusi, J., Use of 3^3 factorial design and multilinear stepwise regression analysis in studying the fluidized bed granulation process: II. *Eur. J. Pharm. Biopharm.*, 39 (1993) 112–116.

Merkku, P., Lindqvist, A.-S., Leiviskä, K. and Yliruusi, J., Influence of granulation and compression process variables on flow rate of granules and on tablet properties, with special reference to weight variation. *Int. J. Pharm.*, 102 (1994) 117–125.

Merkku, P., Yliruusi, J. and Hellén, L., Testing of an automated laboratory scale fluidized bed granulator using different bed loads. *Acta Pharm. Fenn.*, 101 (1992a) 173–180.

Merkku, P., Yliruusi, J., Kaukonen, A., Hellén, L. and Kristoffersson, E., The use of an automated fluidised bed granulator in pharmaceutical process research. *Proc. 11th Pharm. Technol. Conf., Manchester*, 2 (1992b) 227–259.

Mukhopadhyay, S. and Narendra, K.S., Disturbance rejection in nonlinear systems using neural networks. *IEEE*, 4 (1993) 63–72.

Niskanen, T., Yliruusi, J., Niskanen, M. and Kontro, O., Granulation of potassium chloride in instrumented fluidized bed granulator: I. Effect of flow rate. *Acta Pharm. Fenn.*, 99 (1990) 13–22.

Padgett, M.L. and Roppel, T.A., Neural networks and simulation: Modeling for applications. *Simulation*, 58:5 (1992) 295–305.

Savkovic-Stevanivic, J., A neural network model for analysis and optimization of processes. *Comput. Chem. Eng.*, 17 (1993) 411–417.

Stein, R., Preprocessing data for neutral networks. *AI Expert*, 8 (1993a) 32–37.

Stein, R., Selecting data for neural networks. *AI Expert*, 2 (1993b) 42–47.

Weigend, A.D., Rumelhart, D.E., and Huberman, B.A., Generalization by Weigend-elimination with application to forecasting. *Advantages Neural Information Processing Systems*, 3 (1992) 875–882.

Yoon, Y., Swales, G., Jr and Margavio, T.M., A comparison of discriminant analysis versus artificial neural networks. *J. Opl. Res. Soc.*, 44 (1993) 51–60.