

Hybrid Neural Models for Pressure Control in Injection Molding

Tatiana Petrova, Manufacturing Engineer, Sweetheart Cup Company

David Kazmer*, Assistant Professor, University of Massachusetts Amherst

***Author to whom correspondence should be addressed:**

kazmer@ecs.umass.edu

University of Massachusetts Amherst

Eng. Lab. Bldg.

Amherst, MA 01003

Phone (413) 545-0670

Fax (413) 545-1027

Abstract

Industry standards place stringent requirements on the quality of injection molded products. Existing models for quality prediction have a limited accuracy that is not adequate to deliver desired quality targets of three defects per million. Moreover, these quality control strategies often require large amounts of training data while also becoming invalid with external process changes.

This paper compares the performance of three different models for prediction of the injection pressure: a **conventional neural** network, tuned on experimental data; a **simulation network**, trained on simulated data and fine-tuned on experimental data; and a **hybrid network**, which combines the training of neural networks with analytical knowledge of the molding process. The results indicate that the hybrid neural network perform especially well for a small number of training points. In particular, for a number of training points ranging between one and four, the sum of squared errors (SSE) for hybrid network is 13%-33% of the SSE for conventional network and approximately 45% of the SSE for simulation network. In addition, the hybrid network also shows the fastest convergence rate among the three models. However, as the number of training points increase, the conventional neural network outperforms all other models.

Introduction

Injection molding is a widely used process for high volume production of thermoplastic resin parts. It is a dynamic process that contains four stages: plastication, filling, packing and ejection. During the plastication stage, pellets of solid material are transformed into molten polymer. This is usually accomplished by simultaneously heating and mixing the material in a barrel containing a rotating screw. The viscous melt

is then forced at high pressure into a matched metal mold (filling), and held at significant pressure during polymer solidification and shrinkage (packing). Once the molded part has taken its final shape and cooled, it is ejected and the process is repeated. The molded parts then can be used as the end product, but usually they are integrated into larger assemblies to produce the commercial products. This integration often requires higher quality levels and tighter dimensional tolerances of the molded parts.

There are several approaches to quality control and process optimization in manufacturing: expert systems [1, 2], continuous process control [3], and design of experiments [4, 5]. These methods can be classified as those involving either tuning or regulation. The difference between tuning and regulation is graphically presented in Figure 1. Although quite similar, the two processes may differ in step size (usually, tuning requires larger step size), operator confirmation (regulation is hoped to be automatic) and availability of process data (no process history may be available during tuning). In Figure 1, the three circles correspond to the allowable setting of process parameters that provide the desired dimensional and functional product quality. These three circles (dimensional tolerances, absence of burn marks and flash) intersect in the shaded area. The point inside this area represents one of the admissible operating conditions for part performance. The aim of the tuning is to reach this point, while regulation is aimed on maintenance of the point at its position given process variation.

Some of the mentioned methods (e.g. expert systems) describe injection molding as a process with strictly defined relationships between process parameters and part quality. This description, however, is rarely sufficient since injection molding is a complex dynamic and stochastic process. During this process, the polymeric material

undergoes significant shear deformation, temperature and pressure changes that are difficult to relate to final part properties such as surface finish and strength. At the same time, process variation makes it difficult to control the desired quality during production. Different sources of variation in the injection molding include, for example, material properties, machine parameters, mold geometry, environmental conditions, and human presence, as presented in Figure 2. If these input factors had precisely defined and deterministic effects on the process outputs, the quality of the molded part could be easily controlled. However, there are many interactions between stochastic factors, resulting in frequent irregularities in process parameters so the ideal state of constant quality does not exist. Examples of such irregularities include mold and melt temperatures that can drift from their set values, changes in polymer properties from batch to batch, and changes in the machine and mold performance due to wear. Besides these major factors there exists small natural fluctuations making the injection molding process even more unpredictable. Therefore the modeling of the quality dynamics of the process in a deterministic fashion may not lead to the precise results. Other models (for instance, continuous process control) consider injection molding to be a process with unknown relationships, taking into account the stochasticity of the process. However, due to the large number of parameters, it is often difficult to correctly evaluate the significant parameters and their control limits.

These extreme properties of the injection molding process prevent the existing quality control methods from delivering 100% quality assurance. Thus, a more capable quality control system requires a process model that not only generally describes the relationship between the process parameters and the resulting part quality, but is also

capable to learn and express the specific process nonlinearities and complexities in a particular industry application [13]. Because of the stochastic correlations between the process parameters and characteristics of part quality, such a model should also be adaptable to the changing process conditions.

Neural Networks

In related work [6, 7, 8, 9, 10], artificial neural networks (ANN) were used as an alternative approach to other methods (DOE, regression models, SPC, etc) for on-line quality prediction and control. The results of these works show the advantage of the neural networks in terms of expression of process nonlinearities, and that ANNs can be used as a powerful tool in process regulation.

Figure 3 presents a neuron, the basic element of neural network structure, which accepts a signal or set of signals from input or other neurons, sums together the weighted signals, adds a biases value, and passes this summed value through a transfer function. This transfer function specifies how the neuron will scale the response to an incoming signal, and produces an output [11]. The most common transfer functions used in back-propagation method are sigmoid functions, such as the logarithmic sigmoid,

$$f(N) = \frac{1}{1 + e^{-N}} \quad (1)$$

and the hyperbolic tangent sigmoid [14].

$$f(N) = \frac{e^N - e^{-N}}{e^N + e^{-N}} \quad (2)$$

These functions are smoothly defined over the interval of possible input values.

A common learning process for neural networks is back propagation of error. The first step of the back propagation algorithm is computation of an error value for each

output node. During this step, the activation values of the output nodes is compared with a target value. Then the calculated error is propagated backward through the network. At this stage the error value is multiplied by the derivative of the activation function so that the effective change in the input has the desired effect on the output. Finally, the weights and biases are updated using the approximate steepest descent rule. This optimization continues with multiple training data until the error is small enough for application. After the network is trained, the system is tested with a different set of data.

Using a back-propagation algorithm, on-line predictions of mass, sink marks, and surface quality characteristics of molded parts were obtained, for example, by Hausler in [9]. The correlation coefficients for the calculated and measured quality characteristics obtained by neural networks were approximately 15% higher than similar coefficients obtained by a multiple regression model. In other work [8], the back-propagation model was used for the analysis of cavity pressure profiles and then compared with statistical process control. Results of these works show that, due to their nonlinear character, artificial neural networks have the best accuracy prediction of 96% among the other models. However, this accuracy of correlations remain inadequate given quality targets of three defects per million. Also, the model usually requires significant training, which may involve many iterations. This is often not acceptable for on-line applications in which each iteration requires a molded part. Finally, the neural network approach assumes the process to be a 'black box' with unknown relationships among different parameters, and does not provide insight into the process behavior.

The integration of process knowledge into neural networks might be one of the best ways to solve the problem of 100% quality inspection. In the application to injection

molding, such a hybrid system may speed up the convergence, improve accuracy, and eliminate the links that do not affect part quality.

Application

The data for this research was taken from experiments performed at GE Plastics (Pittsfield, MA) [14]. The part, a Hewlett – Packard printer Output Tray (Figure 4), was molded on a 550 ton Krauss Maffei machine. The single-cavity mold was filled from four gates with an unfilled grade of ABS (Cyclocac GPM5500). After the part was molded, several quality-related problems were uncovered. These include tolerances of critical part dimensions, warpage of the side walls, blush at the gates, splay near the cutout, and knit lines due to the meeting of melt fronts from the four gates.

An extensive experimental investigation was performed in support of this work at GE Plastics (Pittsfield, MA). The resulting experimental data set containing 69 subsets, each of which includes from 5 to 20 runs. The data for this investigation was selected from the experimental data set in such a way that it contains 69 representative runs with varying machine inputs. The machine inputs that were adjusted during the experiment were plastication time (sec), dwell time (sec), hold time (sec), back pressure (MPa), shot size (m), cushion (m), injection velocity (m/sec), hydraulic pressure (MPa), and melt temperature (C). The part quality attributes determined for each part were weight, surface aesthetics (presence of burn marks, sink marks, blush, flash formation), the width and length, and the presence of the short shot.

Neural Networks Configurations

In this work the melt pressure was chosen as a focus of investigation for two reasons. First, melt pressure is an observable and continuous characteristic related to the

molded product quality; the melt pressure results from many input parameters and is available to a high resolution. Second, many quality attributes, for instance, short shot and surface quality, are not directly observable as a continuous response; therefore they are subject to interpretation and not suitable for this investigation. In this work three different neural network configurations were trained to determine the best model for the prediction of melt pressure. The trained models include:

- a conventional network with initial weights set at the values corresponding to the estimated behavior of the input-output relationship,
- a conventional network with initial weights obtained from the training of the simulation data, and
- a hybrid network that involves a combination of the phenomenological model of the process and a conventional neural network.

The networks were trained on twenty randomly chosen training subsets and then their performances were compared. The network performances across multiple runs were estimated by summing the squared errors (SSE).

Data Preparation

Before training, all data inputs were scaled according to their activation function, such that every value falls within a given range. For instance, the target outputs for neural networks with sigmoidal function lie in a range from zero to one, thus all input data should be scaled down to this interval. Also the scaling of each variable into a fixed range is preferable since the input variables often cover different ranges. For instance, the shot size data varies in the range from 1.2 to 7.0 cm, whereas the melt temperature data covers the interval from 210 to 259 C. The errors due to the unscaled higher valued variable

would have a greater effect during the training. Thus, ensuring the fact that every variable covers the same range also ensures that errors in each variable contribute equally to the change of network weights. The variables v were scaled to the range from zero to one, producing another variable r , according to the equation:

$$r = \frac{v - \min(v_{1\dots n})}{\max(v_{1\dots n}) - \min(v_{1\dots n})} \quad (3)$$

The target outputs were also scaled to the range of outputs of sigmoid function. The common practice in scaling of the outputs [15, 16] is to use only the relatively linear part of the sigmoid function that lies in between 0.1 and 0.9. Therefore, all output data was scaled to this range:

$$tar = 0.1 + \left(\frac{v - \min(v_{1\dots n})}{\max(v_{1\dots n}) - \min(v_{1\dots n})} \cdot 0.8 \right) \quad (4)$$

where tar is the new output variable.

Weights Initialization

When the neural network is first set up, the weights and biases must be given some initial values from which the training process may begin. These initial values should be chosen in such a way that the learning is performed as fast and accurately as possible. Usually, the initial weight setting contains the small random numbers uniformly distributed around zero. In this work, since the analytical relationships between the input parameters and the pressure are known, the initial weights were estimated from the experimental data. For instance, it is known that the pressure generally rises as the viscosity increases. Therefore, the initial value of the weight on the link from viscosity to pressure should have a positive sign. Furthermore, the approximated value of the weights

can be found from the estimated slope of the viscosity-pressure data. It was defined from the existing experimental data that the initial values of the weights should be set within the range from 0.2 to 0.3. Using similar method, all other initial weights have been determined, as summarized in Table 1.

The number of nodes as well as the optimal network configuration were found by comparison of the performances of several networks of different complexity with different activation functions, random initial weight configurations and training subsets. The network that showed the smallest mean squared error along with the smallest standard deviation for test points was chosen as the basis model. It has to be emphasized that the aim of this investigation is to choose the network which is able to predict the pressure behavior when trained on a minimal amount of data, or, in the other words, to generalize the pressure behavior from the given data subset. To obtain better generalization and, consequently, to reduce overfitting, the number of data points in the training sample can be increased [15]. This approach, however, is not suitable for the injection molding process due to the high sampling cost. Another solution to the problem of overfitting is to reduce the network's complexity by limiting the number of hidden nodes, which is equal to limiting the number of weights. The number of weights determines the number of degrees of freedom that is sufficient for the network to fit the data. In order to perform a generalization with little data, the number of weights should be limited. In this case the capacity of the network is limited, and generalization is performed with greater accuracy [15].

Description of the Conventional Model

A conventional neural network with melt temperature, ram velocity, and shot size used as inputs was constructed and trained. The melt temperature was varied between 210 and 259 C (410 and 498 F), ram velocity between 1.2 and 7.1 cm/sec (0.5 and 2.8 in/sec), and shot size between 7.5 and 8.8 cm (2.96 and 3.47 in). The injection pressure was observed to have values distributed between 5.7 and 14.9 MPa (826.9 and 2167 Psi).

The results of training different networks with one and two hidden layers indicate that the best architecture for the conventional model is a fully connected network with two hidden layers with logarithmic-sigmoidal (logsig) and linear (purelin) activation functions. Further investigation shows that optimal number of hidden nodes in the first hidden layer for this network should be equal to three, since such an architecture gives the smallest error and standard deviation for the test points. The best conventional neural network model has a configuration presented in Figure 5.

Based on this model topology, two different types of conventional network were trained: one with flexible and the other one with fixed weights. During the training of the latter, the weights and biases have been given fixed signs corresponding to the known physical behavior (see Table 1). This method of training, however, does not show any improvements in training ability of the conventional network. Therefore, the training with fixed weights was considered to be inapplicable to this investigation.

Description of Simulation and Network Model

Numerically simulated process data was obtained using Moldflow CAE moldfilling software [17]. The simulation employs physical laws that described the total dynamic behavior of the injection molding system. The dynamic model of the filling

process for a rectangular strip geometry [13] include the local mass conservation equation (eq. 5), the momentum balance in the axial direction (eq. 6) and the local energy balance (eq. 7).

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x}(\rho v) + \frac{\partial}{\partial z}(\rho \nu) = 0 \quad (5)$$

$$0 = \frac{\partial}{\partial z} \left(\eta \frac{\partial v}{\partial z} \right) - \frac{\partial P}{\partial x} \quad (6)$$

$$\rho C_p(T) \left(\frac{\partial T}{\partial t} + v \frac{\partial T}{\partial x} \right) = \frac{\partial}{\partial z} \left(\kappa(T) \frac{\partial T}{\partial z} \right) + \eta \dot{\gamma}^2 \quad (7)$$

In these flow equations, x is the axial coordinate, z is the gap-wise coordinate, and (v, ν) are the velocity components in the local (x, z) directions, t is time, T is the temperature, P is the pressure, ρ is mass density, η is viscosity, $C_p(T)$ is the specific heat, and $\kappa(T)$ is thermal conductivity of polymer melt.

The flow dynamics of the polymer during filling and post-filling stage is highly dependent on the shear viscosity of the polymer melt. Polymers exhibit a non-Newtonian, shear-thinning behavior with increasing shear rate and also are highly temperature and pressure dependent. If polymer viscosity is a function of temperature and shear rate only, then it can be calculated by the following empirical equation (8):

$$\eta = \frac{\eta_0}{\left(1 + \frac{\eta_0 \dot{\gamma}}{\tau^*} \right)^{1-n}} \quad (8)$$

where η_0 is the zero-shear-rate viscosity (eq 9), the parameter n is the power-law index, and τ^* is the stress level at which the viscosity transitions from the Newtonian limit to the

power-law asymptote corresponding to large shear rates [13]. The Newtonian limit is calculated as:

$$\eta_0 = B \exp\left(\frac{T_b}{T_p}\right) \cdot \exp(\beta P) \quad (9)$$

In this equation, B is the reference viscosity coefficient, β is the pressure dependence coefficient, and T_b characterizes the temperature sensitivity of η_0 .

The simulation of the injection molding process was performed while changing the velocity, shot size, and melt temperature. In the simulation, the part's geometry was modeled as a mesh of 1600 elements generated. Each simulation required twenty minutes to run on a 133 MHz Pentium computer with 64 MB of RAM. The injection pressure was extracted from the resulting contour plot, one of which is shown in Figure 6. The generated data contains 125 runs across which the value of shot size was changed from 6.42 to 8.89 cm (from 2.528 to 3.5 in), injection velocity from 1.01 to 6.44 cm/sec (from 0.4 to 2.536 in/sec), and melt temperature was varied from 190.2 to 245.8 C (from 400 to 500 F). The resulting injection pressure covers a range from 2.2 to 9.1 MPa hydraulic (from 315.52 to 1296.86 Psi).

It is difficult, though interesting, to compare the simulation results to the actual pressure values. This is because the exact values for shot size, melt temperature and injection velocity used in the experimental study and their simulated values are not the same. Thus, it would be incorrect to compare the responses for the different initial set points. The approximation using the surface response plotted on simulation data might not be able to provide the correct comparison results as well since the equations which describe the type of the surface (first order, second order) are not known. It must be noted, moreover, that several works [12, 27, 18] described the use of simulation in

improving molded part quality. Qualitatively, the simulation has been found to provide valuable information relevant to the nature of the molding process. Additional studies verified the accuracy of process model [10, 12, 17]. For instance, H. H. Chiang [17] studied the use of the simulation of filling and post-filling stages of the injection molding for different materials and compared the simulation and molding results. Other work [12] describes the development of the dynamic model of the mold filling process and compares the results of simulation with experiment. This comparison shows that the nonlinear dynamic model used in simulations is a reasonable representation of the molding process. Using a similar dynamic model S. Woll, D. Cooper and B. Souder [10] performed experiments that verified the model ability to produce cavity pressure patterns as a function of machine set points.

Using the simulation data, a fully connected network with the same topology as the conventional network was trained. The constructed network contains two hidden layers with logarithmic sigmoid and linear activation functions and three and one neurons in each hidden layer. Data preparation and weights initialization were performed in a similar manner, as was described above for the conventional network. The network is then tuned on a small subset of real molding data, as presented on Figure 7.

Description of the Hybrid Model

The basic equations of the flow dynamics of the polymer during filling and post-filling stage are described the phenomenological model of equations 5 through 9. The value of the shear rate of the polymer in a channel flow can be approximated as:

$$\dot{\gamma} = \frac{6Q}{w \cdot h^2} \quad (10)$$

where Q is the melt flow rate, w and h are width of flow and wall thickness respectively. The cavity pressure P_c can be calculated from the Hagen-Poiseuille equation (11) for the polymer flow of a non-Newtonian fluid between plates as follows

$$P_c = \frac{12\eta \cdot Q \cdot H_{Lc}}{w \cdot h^3} \quad (11)$$

where H_{Lc} is the distance between cavity pressure transducer and the flow front. The Poiseuille flow equation can be used under an assumption that the polymer flow is steady. The equations 6-9 are based on the assumption that the polymer flow is fully developed, laminar viscous flow.

Figure 7 presents the process model when the shear rate, viscosity, and pressure are estimated from velocity, melt temperature and shot size by using empirical equations given earlier (eqs. 8-11). The shear rate in a complex flow varies with position. In this application, the shear rate was estimated as velocity multiplied by some coefficient $K1$, with an optimal value found to be equal $382 \text{ sec}^{-1}/\text{m}/\text{sec}$ using the Matlab subroutine *fmin* (.). The viscosity was calculated according to the equations (8) and (9) with the pressure dependence coefficient β equal to zero. The remainder of the viscosity coefficients are given in Table 2, as measured from a laboratory capillary rheometer.

On the basis of this phenomenological model, the hybrid neural network was built and trained. Estimated viscosity, ram velocity, shot size, and estimated pressure were the inputs for the hybrid network, injection pressure was the output (Figure 9). The left side of Figure 9 corresponds to the analytical model of the process when each node is a

transfer function from Figure 8. The right-hand side incorporates an artificial neural network. The design was chosen such that the topology and internal coefficients of the hybrid model permit the fastest convergence when the network contains just one hidden node with a linear (purelin) activation function, as presented in Figure 9. Moreover, the intermediate values are intended to be of value to a process expert in that they are indicative of shear rate, viscosity, etc.

Training and Results

Using the models described above, multiple training sets were utilized to estimate the effectiveness of the conventional, simulation, and hybrid neural networks. Since the appropriate model should reflect a real molding process where the number of different data sets is limited, the training initially was performed on only four molding points. To investigate the effect of using the simulation data for training the conventional network and the effect of integration of process knowledge into the conventional model, the neural networks were also trained on sets consisting of one and sixteen points from the molding trials. The rest of the data was used to test the accuracy of the networks. One of the subsets used in this study is presented in Figure 10, where shot size is plotted against the injection velocity (left figure) and melt temperature (right figure). Here each 'o' represents the test point and the '*', 'x', and '+' shows the training points, which were chosen randomly from the training set. In addition, the distribution of the points reflects the setting of the values of shot size, melt temperature and velocity during the experiments. For instance, the shot size was set for four different values distributed around 7.7, 8.1, 8.4, and 8.7cm.

The graphs of the *observed pressure vs. predicted pressure* for training on one, four and sixteen points are shown in Figures 11, 12, and 13. In each of the figure, each ‘o’ represents the test point and ‘*’ represents the training point. In the ideal case, when the data does not contain noise and the network is perfectly accurate, the observed value is always equal to the predicted value, as presented by the diagonal line. This ideal case, however, is never realized due to the noise in the data and modeling errors.

The training with only one point, as presented in Figure 11, shows that the conventional system is not able to correctly estimate pressure and always predicts the test point. This is expected as the conventional network is trained on only one point and thus has no access to information regarding process sensitivities. The ability to generalize pressure behavior from minimal amount of data, however, is an important limitation, since in real molding production collecting a large data set is often an expensive procedure.

The simulation network and hybrid networks are able to estimate the molding response using only one training point. In this study, both networks used the observed pressure behavior from the one point to ‘tune’ their internal models. In the case of simulation network, this internal information was introduced by the training on the simulated data. The hybrid network, however, uses this training information directly in tuning its internal phenomenological models. It is this direct implementation of the process knowledge into a neural network that makes the hybrid network outperform the simulation network for one point.

The introduction of three additional points from the molding trials significantly enhances the performance of all networks as presented in Figure 12. The accuracy of

prediction is increased by factor of two in case of simulation and hybrid networks, and even more significantly for the conventional network compared to the one-point training. The accuracy of the estimation, however, is highly dependent on the selection of four points in the training subset space. This dependence is more noticeable for the conventional network, than for the simulation or hybrid networks. For example, if training points are uniformly distributed throughout the data set, the conventional network shows the SSE of 1.11 and hybrid network shows the SSE of 0.45. However, if three points out of four are located at the center of the data set, the conventional network shows a significant reduction of accuracy (SSE=1.975), while the hybrid network, using its internal knowledge, is able to predict pressure with approximately the same accuracy. This ability of the hybrid network to balance and extrapolate from the chosen points results in the lower magnitude of error and reduced prediction variance. The ability for a network to extrapolate beyond the observed range of molding conditions is vital to development of robust quality controllers, since selection of sampling points during production is not guaranteed without incurring delays.

With the utilization of more data points into the training procedure (Figure 13), the ability of the conventional network to predict the pressure increases significantly, such that the conventional network shows the best results when the system is trained on sixteen points. In this case, the conventional model has enough process data to perform accurate training. In addition, the multiple layers of neurons provide greater model flexibility in matching the function output. Training of the simulation model on sixteen points shows an increase in accuracy as well. The hybrid network, however, shows approximately the same accuracy as the four-point training with average sum of squared

error approximately equal to 0.7, which is much less than the accuracy of the other networks trained on the same amount of data. The reason for it is the fact that the hybrid network is less flexible since it involves implementation of artificial built-in constraints that reduce the number of degrees of freedom. For instance, the temperature-pressure relationship in the conventional and simulation models may be more complex than the well-defined viscosity-pressure relation in the hybrid model (e.g. an analytic relationship of eq. 9). In addition, the number of weights that give the flexibility to the neural model is significantly lower for hybrid model compared with the others models.

Table 3 summarizes the results of pressure prediction for the three quality models with average sum of squared error and the standard deviation as a measure of accuracy. The average number of epochs to train the network and the average computing (cpu) time presented in this table to compare the speed of convergence. As can be seen from Table 3, when training is performed on a small amount of data, only the hybrid network is able to accurately estimate the molding response. Given a large amount of data, the conventional model is better able to replicate the observed process behavior with the accuracy of sixteen-point training is high enough for the use of this network in control application. However, it is acceptable only in the case when the size of the training sample is not limited or sufficient amount of data is available. In case of introducing a new product or new material, the cost of tuning with conventional network model can be significant due to the amount of rejected parts during the set up of the control system.

The simulation network produced less accurate results than the hybrid network when the training involve small amount of data, and less accurate than conventional network results when the training performed on large amount of data. However, the

simulation network does show the value of using a-priori data in network initialization when available.

To compare the models in more representative way, the sum of squared errors with their standard deviations were plotted against the number of training points for the conventional, simulation and hybrid models (Figure 14). The accuracy of the models with respect to the error measure was discussed in details above. The hybrid network trained on minimal amount of data produced not only the smallest SSE among the other models, but the smallest standard deviation as well. The standard deviation is a very important measure of performance, since it indicates a high level of model reliability. Quality models with large standard deviation will lead to incorrect quality estimation and exaggerated process disturbances, resulting in significant Type I and Type II error defect levels. A Type I error will lead to rejection of the quality model when the pressure is in fact estimated correctly. Alternatively, it may be that the pressure is replicated inaccurately, but the quality model is accepted (a Type II error) [19].

Conclusions

The complexity of the injection molding process makes it difficult to develop a model for quality prediction that delivers 100% quality assurance. Three different network configurations for the melt pressure prediction have been considered in this work. Conventional, simulation, and hybrid neural networks were trained with the back-propagation learning rule on randomly chosen subsets, which includes one, four, and sixteen points. The results of training these networks in twenty different (randomly chosen) training sets have shown that the incorporation of the process knowledge can significantly enhance the performance of the neural network. The hybrid neural system

shows the ability to predict pressure more accurately than any other model when training is performed on a minimal amount of process data (one and four points). Moreover, the internal data in a structured hybrid network can provide valuable information to the process expert.

The next step should involve the development of a regulation model that leverages the hybrid model topology for on-line quality control. Since the molded part attributes are not always directly observable, a regulation model should be able to relate the process data and machine set points to part quality. Knowing that the inaccuracies of prediction in hybrid neural system exist, the regulation model should continually tune the process model to reflect the observed quality dynamics. Such a regulation model should also continuously monitor the product quality with the objective to detect, reject, and minimize defective parts.

Nomenclature

a	Output of neural networks or hidden layer
b	Biases of neural networks
B	Reference viscosity coefficient
β	Pressure dependence coefficient
C_p	Specific heat
f	Activation function of neural networks
h	Thickness of the wall
H	Total length of polymer
k	Thermal conductivity of polymer
n	Power-law index
N	Input for transfer function
P	Injection pressure
P_c	Cavity pressure

ρ	Mass density
R	Input to the neural networks
r	New value of variable v after scaling
t	Time
T	Melt temperature
tar	New value of the output after scaling
T_b	Temperature sensitivity
Q	Average melt flow rate of the polymer
γ	Shear rate
τ^*	Modified-Cross model parameter
η	Viscosity of polymer melt
η_0	Zero-shear-rate viscosity
(u, v)	Velocity components in the local (x, z) directions
v	Input , output variable
V	Injection velocity
W	Weights of neural networks
w	Width of plate shape mold
x	Shot size

References

1. Kameoka, S., Haramoto, N., Sakai, T., Development of an expert system for injection molding operations, *Advances in Polymer Technology*, 12 (4), p. 403 (1993).
2. Rogers, J. K., 'Intelligent' molding: expert systems are coming on line now, *Modern Plastics*, April 1991, p. 56.
3. Potente, H., Wortberg, J., Hanning, D., Hausler, J., Process monitoring in plastics processing, *SPE ANTEC Tech. Papers*, p. 597 (1993).
4. Blyskal, P. J., Meheran, P. J., Applying DOE analysis techniques to the injection molding process, *SPE ANTEC Tech. Papers*, p. 729 (1994).
5. Vaatainen, O., Jarvela, P., Valta, K., JarveLa, P., The effect of processing parameters on the quality of injection moulded parts by using the Taguchi

Parameter Design method, *Plastics, Rubber and Composites processing and applications*, 21 (4), p. 211 (1994).

6. Hausler J., Wortberg J., Neural network- based system boosts quality, *Modern Plastics*, December 1996, p. 65.
7. Richard, C., Helps, G., Griffen, B.T., Predicting mold-cavity temperatures with an ANN, *Plastics Engineering*, October 1994, p. 25.
8. Woll, S., Cooper, D., Souder, B., Online pattern-based part quality monitoring of the injection molding process, *Polymer Engineering and Science*, 36 (11), p.1477 (1996).
9. Hausler, J., Wortberg, J., Quality Assurance in injection molding with neural networks, *SPE ANTEC Tech. Papers*, p. 123 (1993).
10. Woll, S., Cooper, D., Souder, B., Pattern-based closed-loop quality control for the injection molding process, *Polymer Engineering and Science*, 37 (5), p. 801 (1997).
11. Hagan, M. T., Demuth, H. B., Beale, M., Neural Network design. PWS Publishing Company, 1995.
12. Chiu, C.-P., Shih, M.-C., Wei, J.-H., Dynamics modeling of the mold filling process in an injection molding machine, *Polymer Engineering and Science*, 31 (19), p. 1417 (1991).
13. GENESIS Data Collection System for KM550, GE Plastics, Jan 1997.
14. Smith, M., Neural Networks for statistical modeling, Van Nostrano Reinhold, New York, 1993.
15. Swingler, K., Applying Neural Networks: A Practical Guide, Academic Press, San Diego, 1996.
16. Kennedy, P., Flow analysis reference manual. 1993.
17. Chiang, H. H., Simulation and verification of filling and post-filling stages of the injection molding process, PhD Thesis, Cornell University, Aug 1989.
18. Bushko, W. C., Prediction of dimensional changes and residual stresses in injection molded plastic parts, PhD Thesis, University of Massachusetts, 1996.
19. Devore, J. L., Probability and Statistics for Engineering and the Sciences, Duxbury Press, Belmont, etc, 1995

List of Tables/Figures

Table 1: Initial values of weights for Conventional and Hybrid Networks.

Table 2: Coefficients to calculate viscosity.

Table 3: Results of pressure prediction for conventional, simulation and hybrid networks.

Figure 1: Tuning vs. Regulation

Figure 2: Injection Molding as a System.

Figure 3: Microstructure of Neural Network.

Figure 4: HP Printer Output Tray.

Figure 5: Conventional Neural Network.

Figure 6: Melt Pressure Contour Plot at the end of the Filling Stage.

Figure 7: The Structure of the Simulation Network.

Figure 8: Hybrid Model for Prediction of Injection Pressure.

Figure 9: Hybrid Neural System.

Figure 10: The distribution of Testing and Training Points.

Figure 11: The observed vs. Predicted Pressure for Conventional, Simulation and Hybrid Models, 1 point.

Figure 12: The Observed vs. Predicted Pressure for Conventional, Simulation and Hybrid Networks, 4 points.

Figure 13: The Observed vs. Predicted Pressure for Conventional, Simulation and Hybrid Networks, 16 points.

Figure 14: Sum of Squared Errors and Standard Deviations for Conventional, Simulation and Hybrid Networks.

Table 1: Initial values of weights for Conventional and Hybrid Networks.

Input	Weight Sign	Weight Value
velocity	>0	0.2 ... 0.3
shot size	>0	0.3 ... 0.4
melt temperature	<0	-0.4 ... -0.5
estimated viscosity	>0	0.2 ... 0.3
estimated pressure	>0	0.4 ... 0.5

Table 2: Coefficients to calculate viscosity.

B	3.44e-6 PaSec
τ^*	5.18e4 Pa
n	0.319
Tb	1.04e4 K

Table 3: Results of pressure prediction for conventional, simulation and hybrid networks.

	One Point			Four Points			Sixteen Points		
	convent	simulat	hybrid	convent	simulat	hybrid	convent	simulat	hybrid
SSE	7.3234	2.0517	0.9319	2.197	1.597	0.7257	0.6409	0.6714	0.7325
STD	4.4226	0.7816	0.5719	0.6455	0.4457	0.328	0.6104	0.4133	0.5718
Epoch	40.8	33	28.6	158.4	88.5	72.5	696.4	669.8	807.6
cpu time	2.35	3.48	1.88	4.77	4.03	2.53	16.46	10.09	12.04

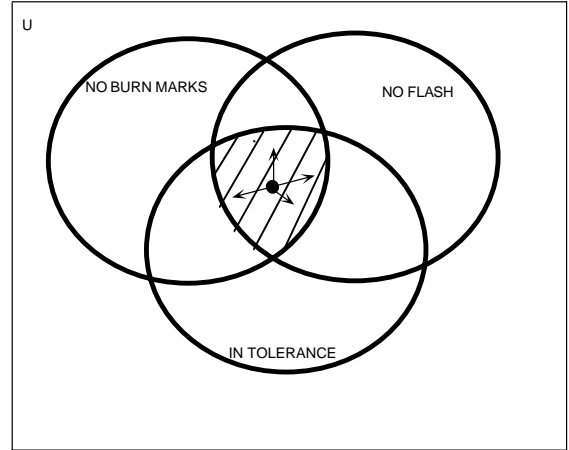
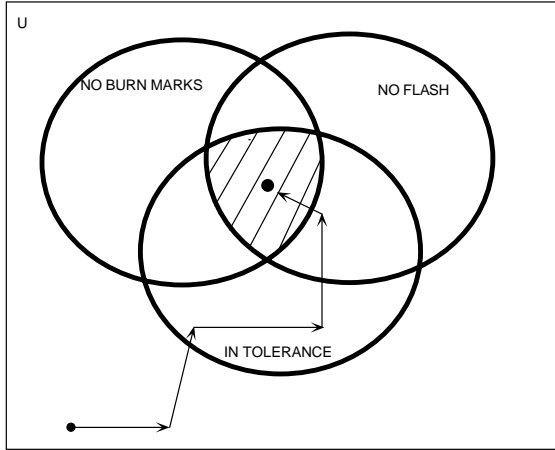


Figure 1: Tuning vs. Regulation

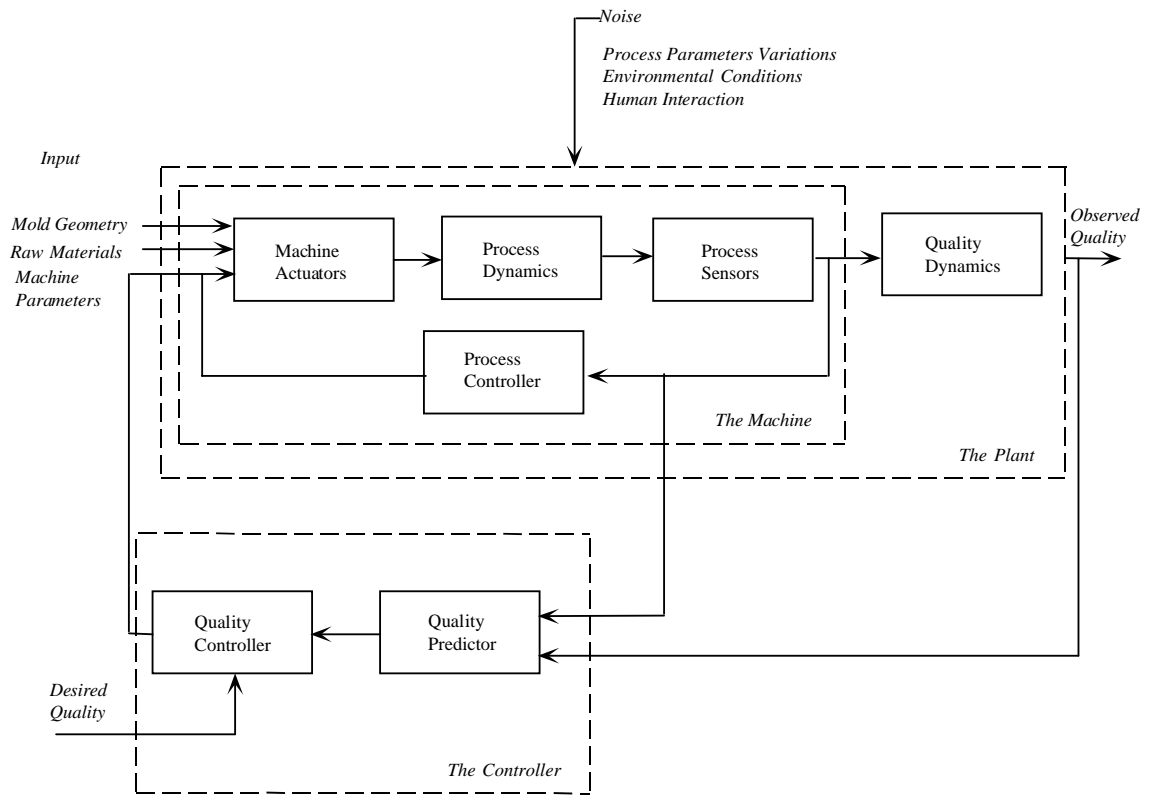


Figure 2: Injection Molding as a System.

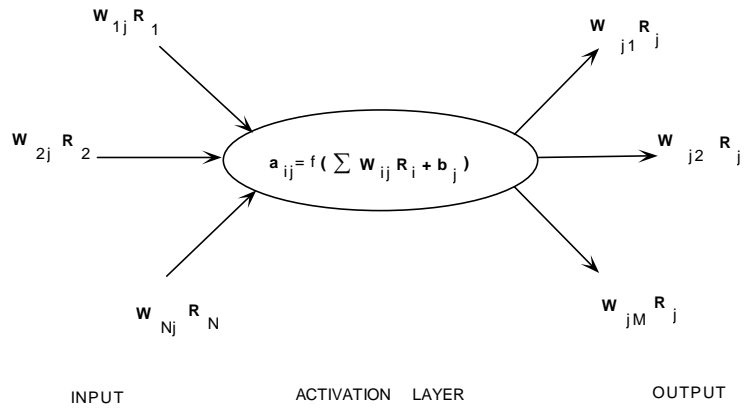


Figure 3: Microstructure of Neural Network.

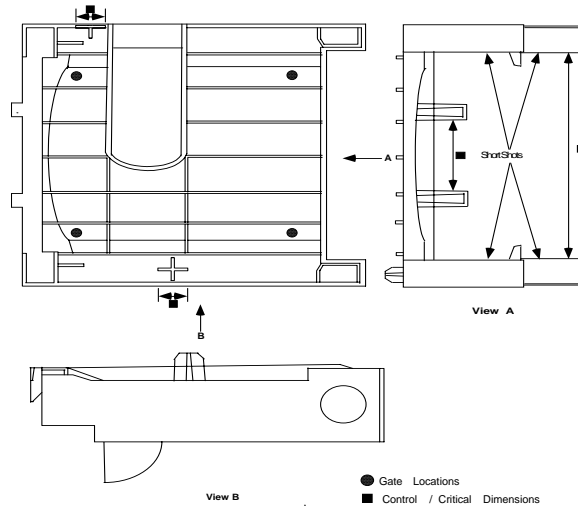


Figure 4: HP Printer Output Tray.

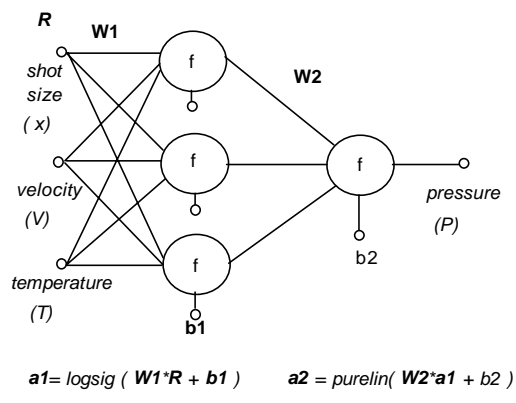


Figure 5: Conventional Neural Network.

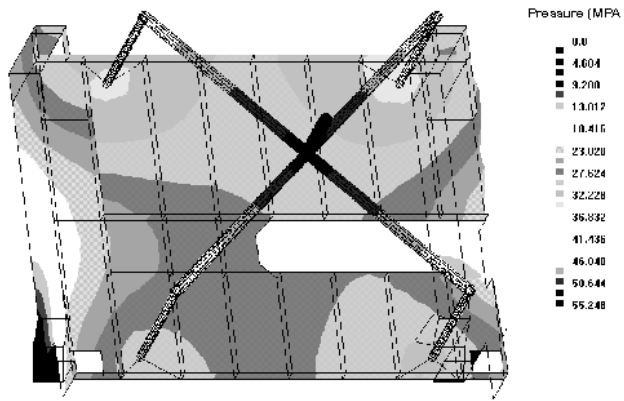


Figure 6: Melt Pressure Contour Plot at the end of the Filling Stage.

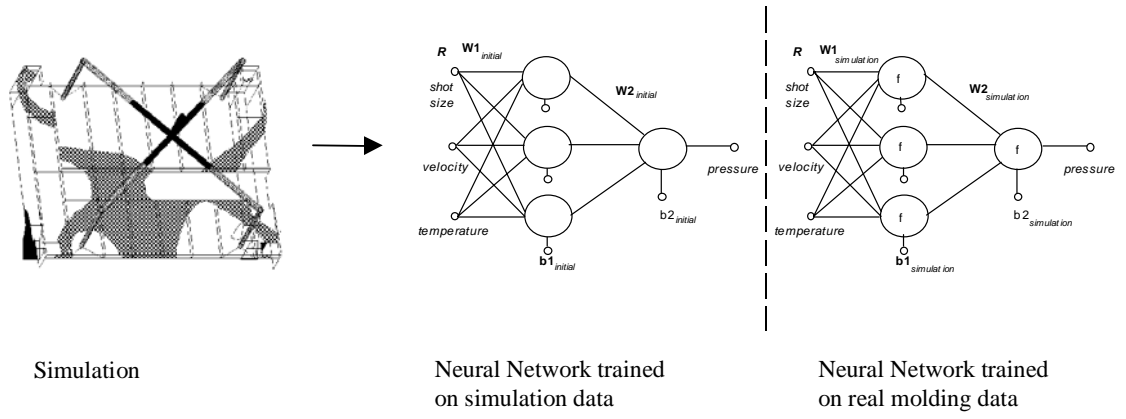


Figure 7: The Structure of the Simulation Network.

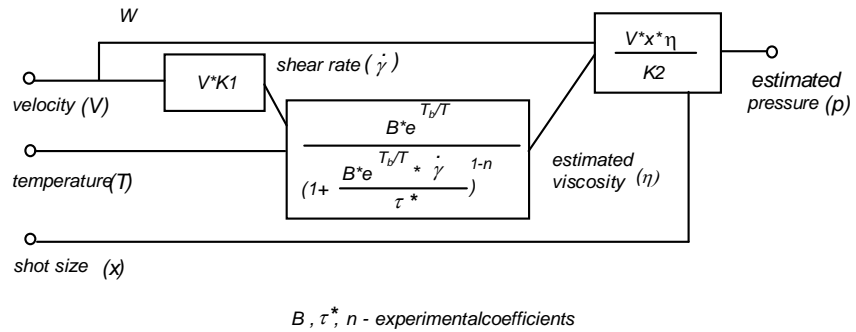


Figure 8: Hybrid Model for Prediction of Injection Pressure.

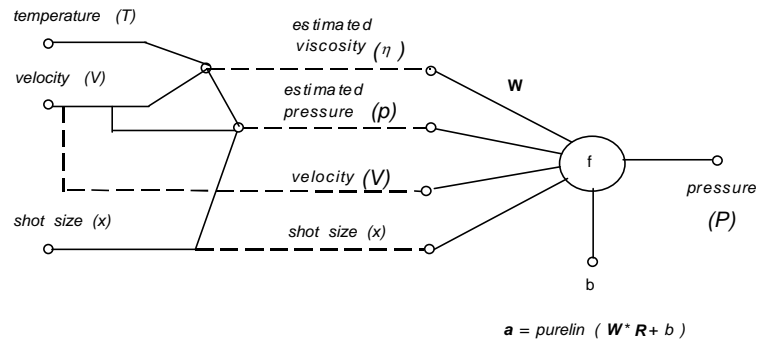
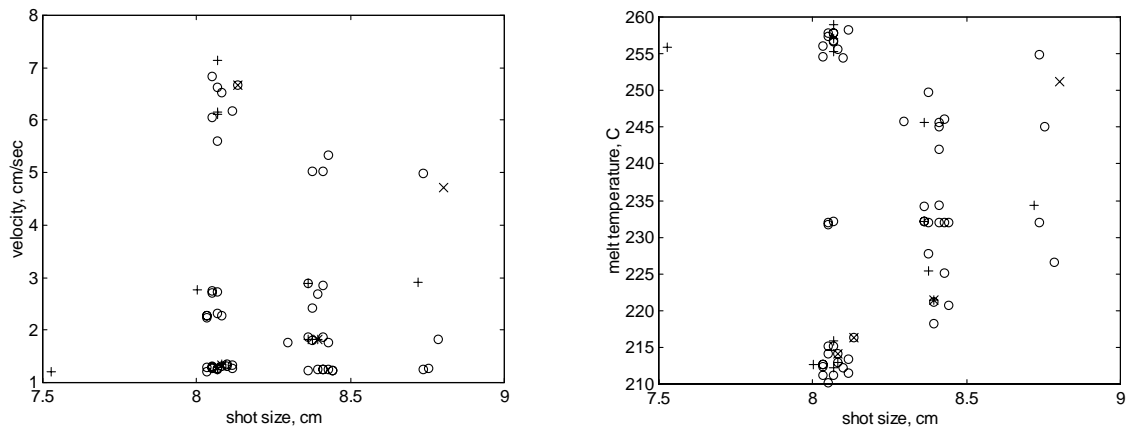


Figure 9: Hybrid Neural System.



- o test point
- * point used for one-point training
- x points used for four-point training
- + points used for sixteen-point training

Figure 10: The distribution of Testing and Training Points.

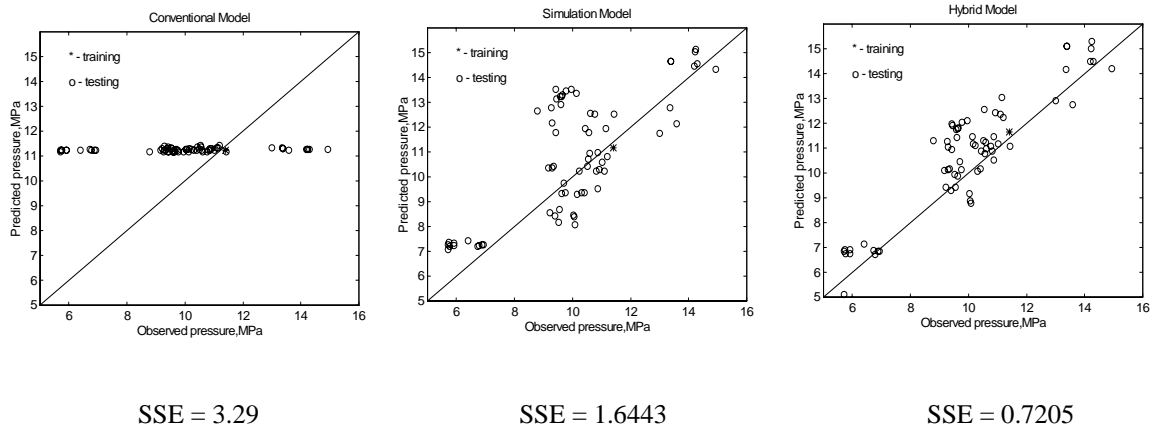
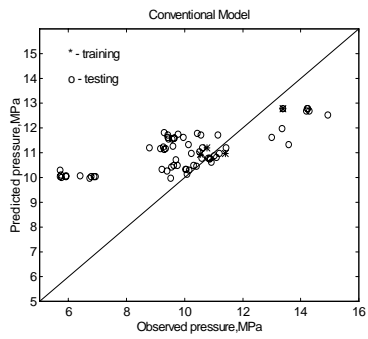
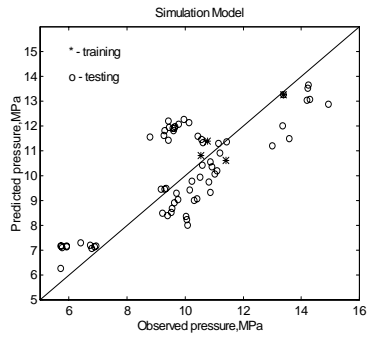


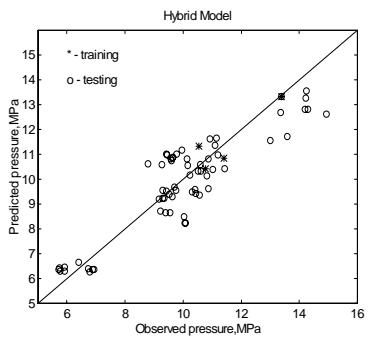
Figure 11: The observed vs. Predicted Pressure for Conventional, Simulation and Hybrid Models, 1 point.



SSE = 1.9750

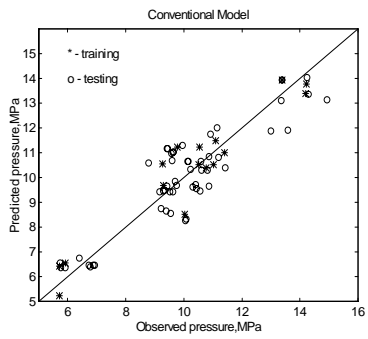


SSE = 0.9700

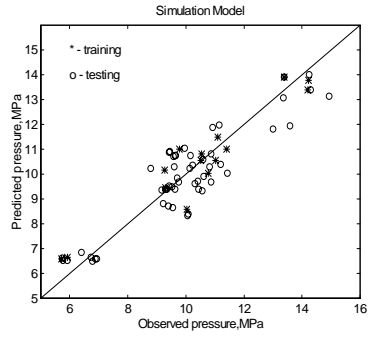


SSE = 0.4501

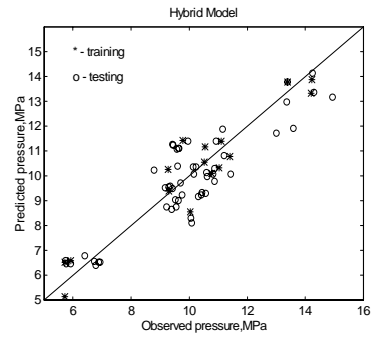
Figure 12: The Observed vs. Predicted Pressure for Conventional, Simulation and Hybrid Networks, 4 points.



SSE = 0.3



SSE = 0.3



SSE = 0.4

Figure 13: The Observed vs. Predicted Pressure for Conventional, Simulation and Hybrid Networks, 16 points.

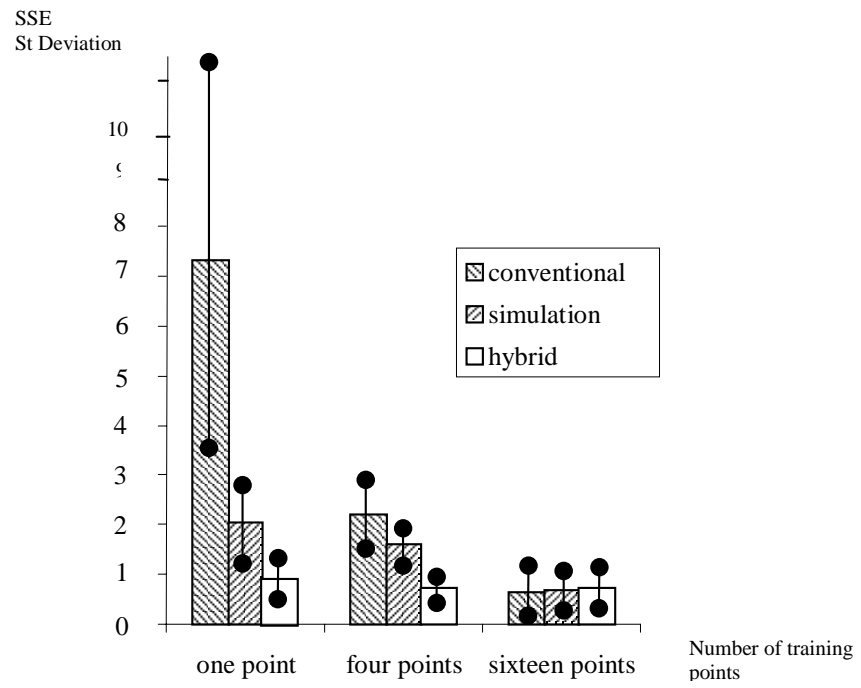


Figure 14: Sum of Squared Errors and Standard Deviations for Conventional, Simulation and Hybrid Networks.