Modeling a wood-chip refiner using artificial neural networks

Yu Qian, Patrick Tessier, and Guy A. Dumont

ABSTRACT: The mechanism of the wood-chip refining process is still being studied, and no thorough model has yet been developed. Neural networks can be an attractive alternative to mathematical modeling of complex processes if a sufficient amount of input–output data is available. This article examines the use of a feed-forward neural network to model a wood-chip refiner. The network's predicted outputs compared favorably with industrial refiner data. It is also shown that the network structure can be modified to optimize refiner operation and product quality. Advantages and disadvantages of applying neural-network models to simulate and optimize industrial processes are discussed.

KEYWORDS: Artificial intelligence, chips, models, neural networks, optimization, quality, refiners, simulation.

It is difficult to develop accurate mechanistic models of complex industrial processes. The modeling of industrial processes frequently involves a mix of mass and heat transfer, fluid mechanics, thermodynamics, and kinetics. The result is a system of nonlinear, coupled algebraic or differential equations. Complex systems require a large number of equations to describe the phenomena taking place in the process, and many parameters have to be estimated. Moreover, process noise and other disturbances introduce additional complications.

Artificial neural networks can assimilate operating data from an industrial process and learn about the complex relationships existing within the process, even when the input–output information is noisy and imprecise. This ability makes the neural-network concept well suited for modeling complex industrial processes. So far, neural networks have been used mainly for fault diagnosis and process control. Because industrial operating data are widely available from distributed control systems, neural-network modeling based on historical data appears to be a generic and cost-efficient approach that can be applied across many plants within a company *(1)*.

Modeling the refining process

Wood-chip refining is considered to be the most critical step in the course of manufacturing thermomechanical pulp (TMP) and chemithermomechanical pulp (CTMP). Although experience from TMP and CTMP processes has been accumulated for years, systematic acquisition and representation of this process knowledge remains a difficult task. Specific energy and refining consistency are the two major manipulating variables used to control refiners, but there are many other variables affecting pulp quality. These variables include the raw-material characteristics, refiner-disc pattern, plate wear, and previous thermal and chemical treatments of the chips.

Modeling the chip refiner from fundamental principles poses a challenge mainly because of the complexity, the many variables, and the stochastic nature of the process. There have been some recent attempts at modeling and simulating refiner performance *(2, 3)*. Nevertheless, there is still a long way to go in understanding and describing the

Qian is a post-doctoral fellow in the Dept. of Chemical Engineering. Dumont is a professor in the Dept. of Electrical Engineering. Both are affiliated with the University of British Columbia, Vancouver, BC, Canada V6T 1Z4. Tessier is president of ISAC Technology Inc., 443 East 1st St., North Vancouver, BC, Canada V7L-1B8.

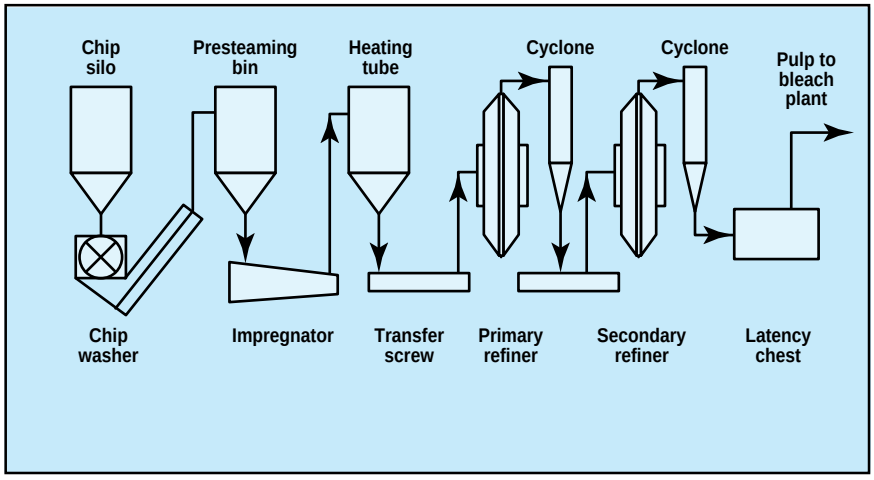
entire refining process and the complex relationships among all the variables involved. Even if it were possible to develop a complete model for one refiner or one mill, it would have to be reconstructed for each particular application, since each production line has its own characteristics.

Factor-analysis techniques have been used to analyze industrial processes when there is no mechanistic model available *(4, 5)*. Factor analysis *(6)* is a data-reduction technique for analyzing interactions between system input and output data. The technique is useful because it can determine significant independent system parameters from mill operating data. In an effort to model and optimize the pulp-refining process, Strand *(4, 5)* used "common factors" as links between intrinsic fiber properties and pulp qualities. Examining how each of the pulp properties is affected by these common factors provides a qualitative feel about the physical nature of these common factors. This approach is useful in understanding which refining variables need to be controlled or adjusted to improve pulp quality. However, the linear model used in factor analysis is an oversimplified assumption, and it would provide an inadequate description of complex processes with nonlinear interactions among system variables.

A neural-network-based simulator can overcome this shortcoming because the network does not rely on exact mathematical equations to describe a process. Using input–output data from a system, a neural network can be "trained" and calibrated to capture the nonlinear characteristics of that system.

This article presents the results of a study investigating the feasibility of using a feed-forward neural network for knowledge acquisition and processing. A neural-network model developed using data from an industrial refiner proved to be a useful vehicle for understanding the effects of process variables on pulp quality. Once such a network model has been verified, it can be used to optimize the system, either minimizing energy consumption at a given pulp quality or maximizing pulp quality.

1. Schematic of CTMP process

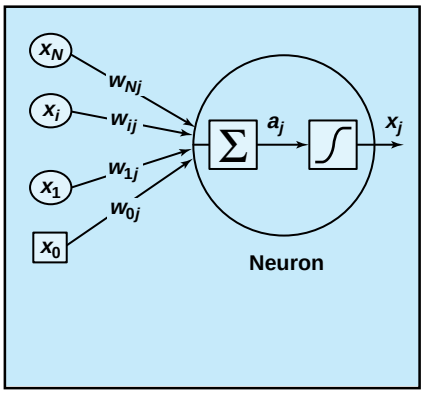


Process description

Figure 1 is a schematic representation of a two-stage CTMP process. Wood chips are delivered from the chip silos to a chip washer, which removes sand, grits, and other contaminants. A metering screw feeds the washed chips to the presteaming bin, where they are heated for approximately 20 min. The chips are then compressed into an impregnator and released into a chemical solution. After chemical treatment, the chips are heated to 110–135°C for 2–5 min in the heating tube. From there, they fall into a surge tube and then are fed to the first-stage refiner by means of a transfer screw feeder. After the pressurized primary-refining stage, the material goes to a cyclone for steam separation and pressure reduction before being fed to the secondary refiner. The pulp is finally conveyed to a tank with a retention time of approximately 20 min for latency removal.

Operating data were obtained from a commercial refiner (Sunds Defibrator RGP-60) in a Scandinavian mill *(5)*. A total of 110 sets of data were collected. Each data set consisted of three operating variables (production rate, specific energy, and refining consistency) and five pulp-quality variables (average optical fiber length, long-fiber content, pulp freeness, breaking length, and tear index). The operating conditions cover a reasonable range for this type of operation, as shown in **Table I**. The table also shows the ranges of the five pulp-quality variables.

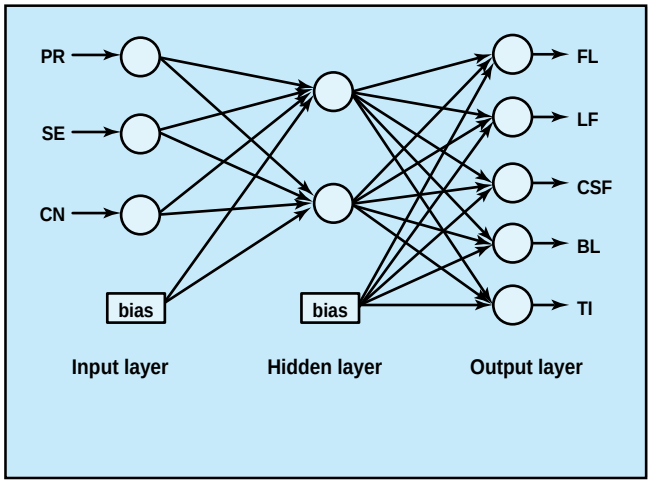
2. Processing steps inside a neuron



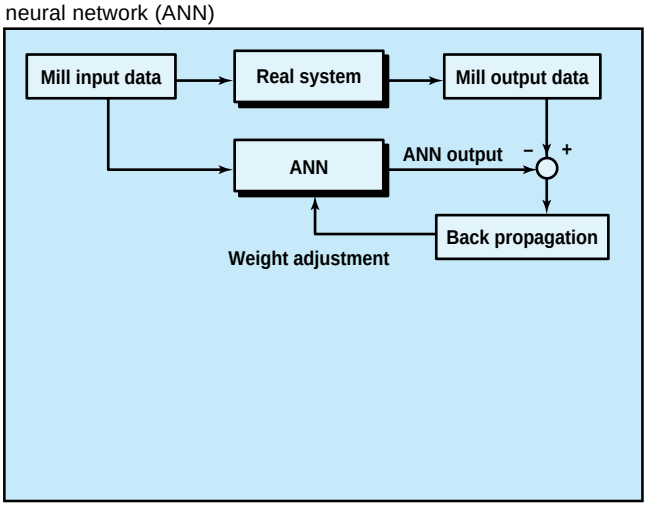
Feed-forward neural network

An artificial neural network is a computing system made up of simple, interconnected processing elements called neurons, by analogy with the human brain. Neurons are arranged in layers. The feed-forward neural

3. Neural-network representation of the chip-refining process



4. Role of back propagation in the learning process for an artificial neural network (ANN)



I. Refiner operating conditions and pulp and paper quality

	Symbol	*Range*
Refiner operating conditions		
Production rate, o.d. metric tons/day	PR	72, 85, 91, 96
Specific energy, kW·h/o.d. metric ton	SE	1450–2250
Consistency, %	CN	35–55
Pulp and paper quality		
Average fiber length, mm	FL	0.45–0.90
Long-fiber fraction, %	LF	25–58
Freeness, mL CSF	CSF	80–380
Breaking length, km	BL	1.8–3.6
Tear index, mN·m²/g	TI	6.8–9.6

network has an input layer, one or more hidden layers, and an output layer.

The input layer simply transmits the input variables to the next layer without any processing. A single neuron is shown in **Fig. 2**. Neurons in the hidden or output layers receive weighted inputs from each of the neurons of the previous layer. These weighted inputs are added to get a net input, a_j, to the neuron.

$$a_j = \sum_{i=0}^{N} w_{ij}\, x_i \qquad (1)$$

where

N = number of neurons in the previous layer

x_0 = signal bias

w_{0j} = weight for the bias of the neuron j.

The neuron output x_j is calculated by an activation function. For classification problems, threshold neuron-activation functions are normally used. For function approximation (as in this work), continuous-value logistic neurons can be used.

$$x_j = \frac{1}{1+e^{-a_j}} \qquad (2)$$

This results in a sigmoidal output ranging from 0 to 1.

A neural network for the chip-refining process was constructed. The network is illustrated in **Fig. 3**. The input layer consists of three neurons representing the refiner operating conditions: production rate (PR), specific energy (SE), and refining consistency (CN). The hidden layer consists of two neurons. The five neurons in the output layer are the optical average fiber length (FL), the long-fiber fraction (LF), the freeness (CSF), the breaking length (BL), and the tear index (TI).

Training the network

Background

The objective of the network "training" is to determine and adjust the weight distribution of the network's input and hidden-layer neurons based on a given set of input and output data. Back propagation *(7)* is used as the learning algorithm. This method of learning involves presenting the network with an input pattern and then propagating the values forward through the network. The final values of the output vector are compared with a target vector, and an error signal is generated. The error signal is propagated back to the hidden and input layers through the network. The weight of each neuron is modified in proportion to the degree that it contributed to the error. This process is repeated for all of the input–output patterns many times. The final goal is to train the network to reproduce the input–output mappings to within an acceptably small

II. Network performance with different input–output structures

Number of nodes			Relative deviation					Average
Input	Hidden	Output	FL	LF	CSF	BL	TI	deviation
3	2	1	7.1	7.2	9.2	6.4	7.4	7.5
3	2	5	8.4	6.8	10.7	6.9	6.9	7.9

III. Network performance with different numbers of neurons and layers

Number of nodes				Number	Relative deviation					Average
Input	Hidden		Output	of links	FL	LF	CSF	BL	TI	deviation
3	1	...	5	14	8.6	6.1	11.5	8.8	9.0	8.8
3	2	...	5	23	8.4	6.8	10.7	6.9	6.9	7.9
3	3	...	5	32	7.3	6.1	9.8	6.7	6.5	7.3
3	4	...	5	41	7.5	6.1	10.2	6.6	6.4	7.4
3	2	2	5	29	7.3	5.9	9.0	7.4	6.4	7.2
3	2	3	5	37	7.3	5.9	9.1	7.2	6.3	7.2

error. The training process is illustrated in **Fig. 4**.

Once trained and tested, the neural network can be used in feature extraction, pattern recognition, and simulation of multiple-input–multiple-output systems. The knowledge is stored in the network through the weight distribution, which represents the relationship between inputs and outputs.

The 110 available data sets from the mill were divided into two groups. One group was used for training, while the remaining data sets in the other group were used to test the performance of the neural network. We conducted training sessions using 50, 70, and 90 data sets. The modeling accuracy increased with the number of data sets. The results presented here are for the network-training sessions using 90 data sets. The remaining 20 sets were used to test the network's performance. Since neural networks are phenomenological models, data supplied for training should be large enough to cover the operating ranges of interest. Such models cannot be used to simulate events beyond the operating range.

Input–output structure of the networks

There is no specific methodology for determining the best network configuration for a particular modeling application *(8)*. The refining process could be modeled by several separate networks, each of them representing a mapping between the input-variable vector (PR, SE, and CN) and one of the output pulp-quality variables (FL, L, CSF, BL, or TI). Networks built this way should be more specifically adapted to the chosen output variable, especially when output variables are independent of each other. However, output variables are not necessarily independent of each other. In the present case, the fact that the system has more output variables than input variables suggests the existence of integration among output variables.

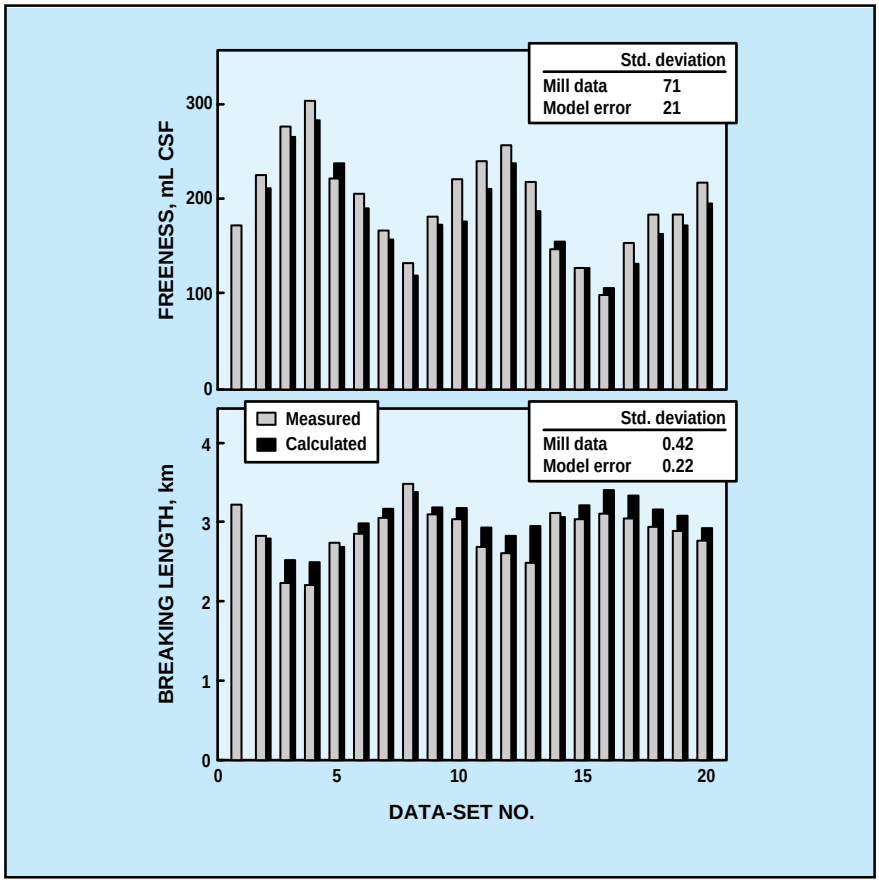
Another approach would be to train a neural network to correlate the set of input variables and the whole set of output variables simultaneously. A network structured in this way must be capable of recognizing the common pattern in the transformation from the input-variable vector to the output vector.

Table II compares the performance of these two network structures.

The structure of a network can be represented as

$$\{n_{in}, n_{hidden\text{-}1}, n_{hidden\text{-}2}, \ldots, n_{out}\} \qquad (3)$$

5. Comparison of network outputs with 20 data sets from the refiner



where

n_{in} = nodes in the input layer

$n_{hidden\text{-}1}$ = nodes in the first hidden layer

$n_{hidden\text{-}2}$ = nodes in the second hidden layer

n_{out} = nodes in the output layers.

As seen in Table II, the relative deviation for the five networks with {3, 2, 1} structures was in the range of 6.4–9.2%, with an average deviation of 7.5%. The average deviation of the network correlating all five output variables—the {3, 2, 5} structure—is nearly equivalent at 7.9%. Since the latter structure needs far less parameters (23 weights) than the former (55 weights), we chose the {3, 2, 5}structure in our modeling, which is more reliable in parameter estimation.

Effect of hidden layers and node numbers

Having determined the input–output structure of the neural network, we went on to test the effects of the number of hidden layers and nodes. The results are shown in **Table III**.

Networks with a single hidden layer (the first four rows in the table) perform about as well as networks with additional hidden layers (rows 5 and 6). Although networks with more neurons and weights perform slightly better than networks with less neurons, no particular network gives minimum deviation for all properties at the same time. It seems that the accuracy of the modeling for simulation problems like this one is relatively insensitive to the number of nodes and hidden layers over a wide range of values. It is understandable that if more hidden layers are used, there will be more nodes and weight parameters to be estimated, and the network will be more sensitive and vulnerable to errors. To avoid over-parameterization, we suggest that a simple network structure, say {3, 2, 5} or {3, 3, 5}, should be sufficient for modeling the present static nonlinear system.

Performance of the neural network

The performance of the trained network was tested with the remaining 20 data sets from the mill. The neural network was able to predict the five pulp-quality variables under different operating conditions. **Figure 5** compares the network predictions with the refiner data for freeness and breaking length. The standard deviation for the model error—the difference between the mill data and the predicted values—is well below the standard deviation for the mill data.

It is well accepted that at least two variables should be manipulated to control pulp quality. When the production rate is dictated by economic reasons and equipment constraints, the operators can still use the closing pressure (or motor load or plate gap) and the dilution water to change the specific energy and the consistency, respectively. **Figure 6** shows good agreement between the predicted and mill values for tear index and long-fiber content at variable specific energy and consistency.

Modeling and optimizing the refining process

Using the weighted inputs obtained from the training, the feed-forward neural network can be used as a simulator to predict pulp quality from refiner operating conditions. **Figure 7** shows how pulp quality (freeness, breaking length, tear index, and fiber length) is predicted to change as we manipulate the two operating variables (specific energy

and refining consistency). Alternatively, the network can be used to analyze the interactions between input and output variables and thus determine the operating conditions required to achieve a specified level of pulp quality. It can also be used to give a visual image of the input–output relationship to the operators.

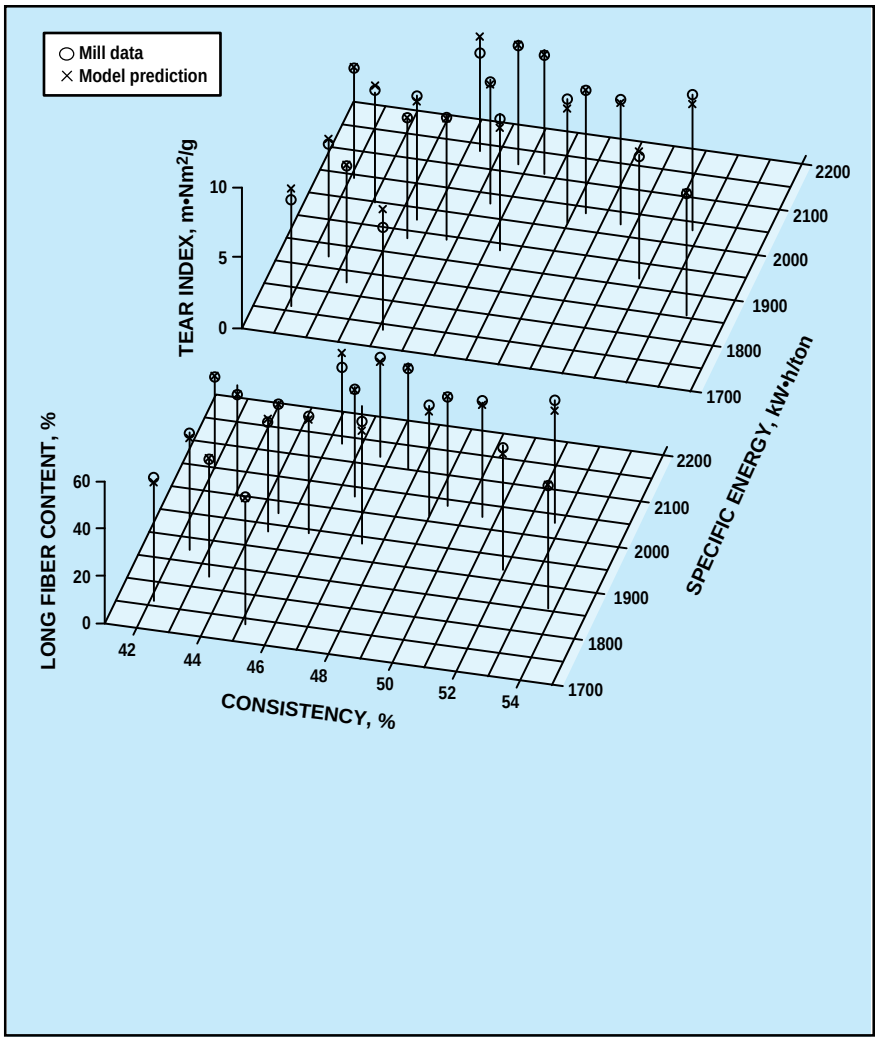
For example, there is a controversy in describing the energy-consumption–tear-index relationship. Under different circumstances, tear index increases and then decreases over the operating range for specific energy. From our neural network modeling, it is shown that, at low consistency, the tear-index–specific-energy curve is a convex function. At high consistency, tear index changes little over a wide range of specific energy. The relationship at low and high levels of specific energy also shows different modes. These characteristics would not have been found if a linear factor-analysis method had been used.

The task of optimizing operations or controlling product quality involves maximizing or minimizing certain process variables by manipulating some operating conditions, maintaining other operating conditions, and holding product quality within a desired range. To make this clear, we distinguish these as three sets of variables:

1. Controlled variables P_C: those to be controlled or to be optimized (either a product-quality specification or an operating variable)
2. Manipulated variables P_M: those to be manipulated or adjusted (usually a number of operating variables) in order to optimize P_C
3. Other variables P_O: those expected to remain constant or within steady ranges (product quality window and/or ordinary operation ranges).

This type of problem is usually solved via a nonlinear mathematical programming approach, which is complicated and difficult when there is no complete mathematical model of the relationships among the process variables.

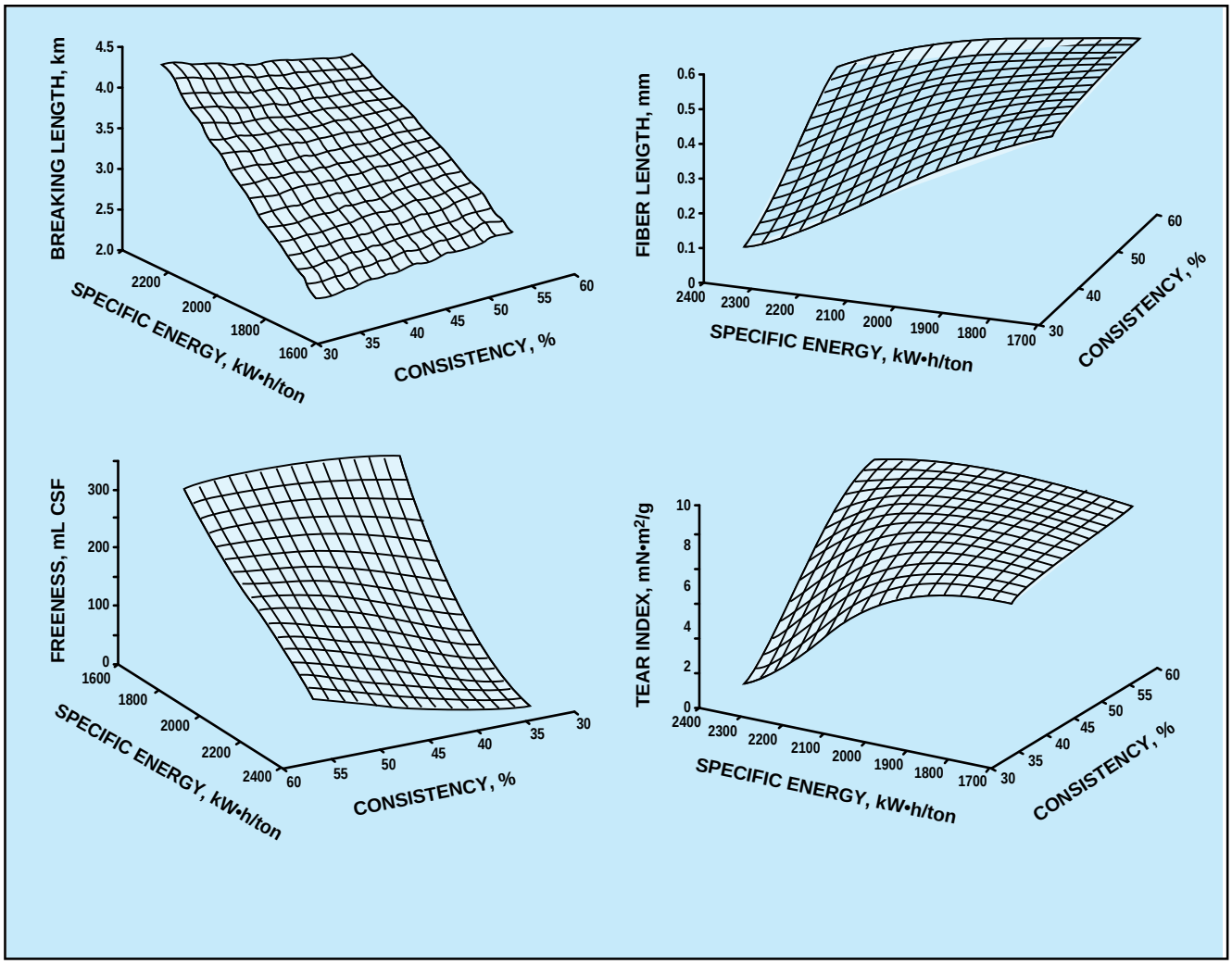
6. Comparison of network outputs with 20 data sets from the refiner



Neural networks can contribute significantly in analyzing and optimizing the performance of processing units and plants. One of the unique and useful features of neural networks is that their inputs and outputs do not have to conform to those in the physical situation they are modeling. Thus the network's input can be a mix of easily obtained physical inputs and outputs of a unit or plant, while the network's output can be part of the operating conditions. This approach is shown in **Fig. 8**. It is, of course, necessary that all conditions and properties significant to the process be included either as input or output data for the network to adequately portray the physical process.

To illustrate this approach, consider the following example. There are a variety of criteria used to evaluate pulp quality and papermaking characteristics for different end products. Among them, breaking length (BL) and tear index (TI) are usually used to represent the paper strength. A modified network structure can be useful in determining how refiner operating conditions should be adjusted to control these two quality parameters. **Figure 9** illustrates a new network construction for the purpose of optimizing pulp quality. In this network, the controlled variables (TI and BL) are

7. Model results showing four pulp-quality variables as functions of specific refining energy and pulp consistency



placed in the set of *input* variables. Two manipulated variables (specific energy, SE, and refining consistency, CN) are put in the set of *outputs*. Production rate is not going to be changed so it remains in the input layer. All other pulp-quality variables are dependent variables.

The same set of mill data was used to train this restructured network. After training, the network can be used to predict which SE and CN setpoints should be chosen when new quality requirements are given. **Figure 10** is a contour plot showing how the operating conditions (SE and CN) should be changed as the pulp-quality requirements change.

In this study, we assigned only three main operating variables as inputs of the neural network. Actually, there is quite a large set of variables that can affect refiner operation, such as plate age and wood-chip properties. It is necessary to include all significant variables of a real refiner operation in a neural-network modeling system. Thus when a disturbance is observed, the neural network can make the appropriate response and adjust the other operating parameters to compensate for the disturbance.

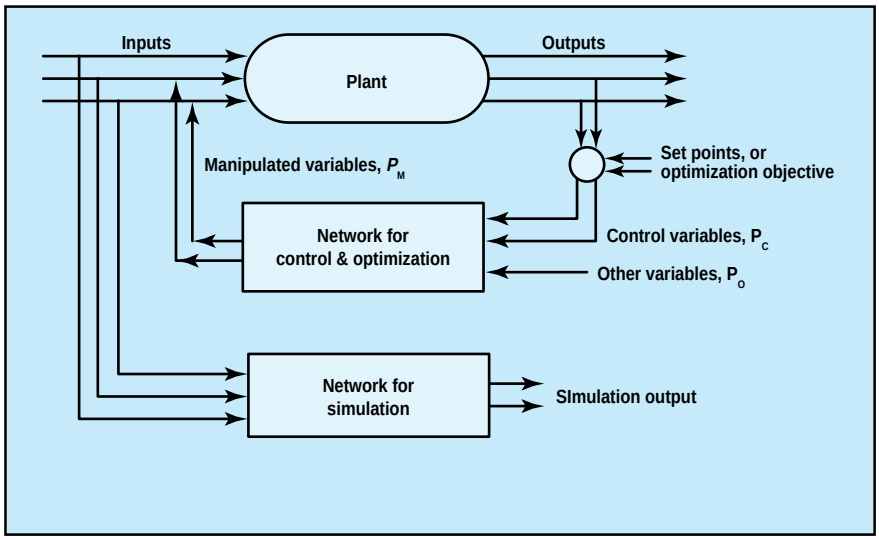
The primary advantage of the neural-network approach, compared with other modeling approaches, is that it does not require the development of a complete relationship model, a difficult task for many processes. Neural networks allow mapping the input–output relationships without a full understanding of the functional relationships.

The disadvantage of this approach is the need to provide a set of data that spans the region of interest. Since neural-network modeling is a phenomenological approach rather than a mechanistic approach, the model cannot safely be used to make predictions outside the range of operating conditions covered by the set of data used to train the neural network.

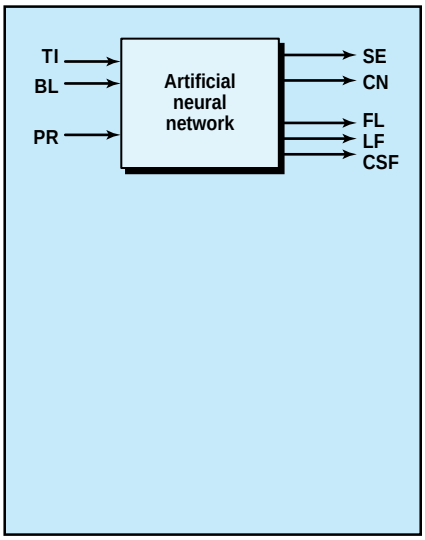
Conclusion

This work demonstrates that a complex wood-chip refining system can be modeled and implemented in a

8. Network configurations for process simulation and for process control and optimization



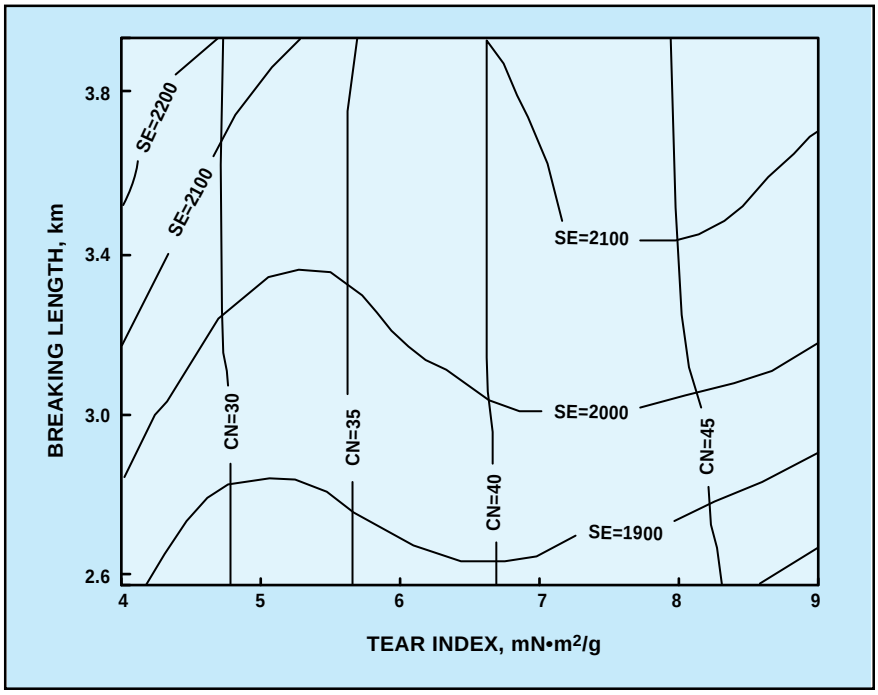
9. Network configuration for controlling and optimizing tear index (TI) and breaking length (BL) by manipulating specific energy (SE) and consistency (CN)



feed-forward neural network. Considering the difficulty and complexity of developing a mechanical model of the wood-chip refining process, the neural network can be an effective alternative. The neural network can learn about the relationships among process variables using mill data from the distributed control system and operator logbooks. This approach does not require the enormous efforts that are required to develop mechanistic models or rule-based expert systems.

A neural-network model was "trained" using data from a commercial CTMP refiner. The model's predictions of pulp and paper properties were in excellent agreement with industrial refiner data. This demonstrates that a neural network can be successfully used in steady-state modeling and optimization of complex industrial processes, including the wood-chip refiner. TP

10. Contour plot of two operating conditions—specific energy (SE) and consistency (CN)—as functions of the pulp-quality requirements breaking length (BL) and tear index (TI)



Literature cited

1. Åström, K. J. and McAvoy, T. J., *J. Process Control* 2(3): 115(1992).
2. Jones, G. L., *Pulp Paper Can.* 89(6): T214(1988).
3. Jones, G. L., *Pulp Paper Can.* 91(2): T81(1990).
4. Strand, B. C., *TAPPI 1987 Pulping Conference Proceedings*, TAPPI PRESS, Atlanta, p. 61.
5. Strand, B. C., "Simulation, control and on-line optimization of a mechanical pulping system," Ph.D. thesis, University of Idaho, Moscow, ID, 1989, pp. 123–51.
6. Cureton, E. E. and D'Agostino, R. B., *Factor Analysis: An Applied Approach*, Lawrence Erlbaum Assoc., New York, 1983.
7. Rumelhart, D., McClelland, J., Stein, R., *et al.*, *Parallel Distributed Processing*, MIT Press, Boston, 1986.
8. Bhat, N. V. and McAvoy, T. J., *Computers & Chemical Engineering* 16(4): 271(1992).

Received for review Sept. 2, 1993.

Accepted Aug. 2, 1994.