Tappi Journal, Vol. 83, No. 7, p. 66, 2000.

TAPPI JOURNAL PEER REVIEWED PAPER

Analysis of pulping data using the self-organizing map

Esa Alhoniemi
Helsinki University of Technology
Laboratory of Computer and Information Science
P.O. Box 5400
Finland

ABSTRACT

Analysis of process data makes it possible to obtain useful information of processes or phenomena that are analytically difficult to deal with. To demonstrate the possibilities of this kind of approach, an analysis of the faults that occurred in a continuous digester is presented. The process data are fed to an artificial neural network, the Self-Organizing Map (SOM), which is used to form visual presentations of the data. By interpreting these visualizations, the reason for the faults can be determined.

INTRODUCTION

In modern process automation systems, it is possible to collect and store huge amounts of measurement data. Furthermore, computer improvements have made it possible to efficiently analyze these data. In **Fig. 1**, different approaches to the treatment of process data are presented.

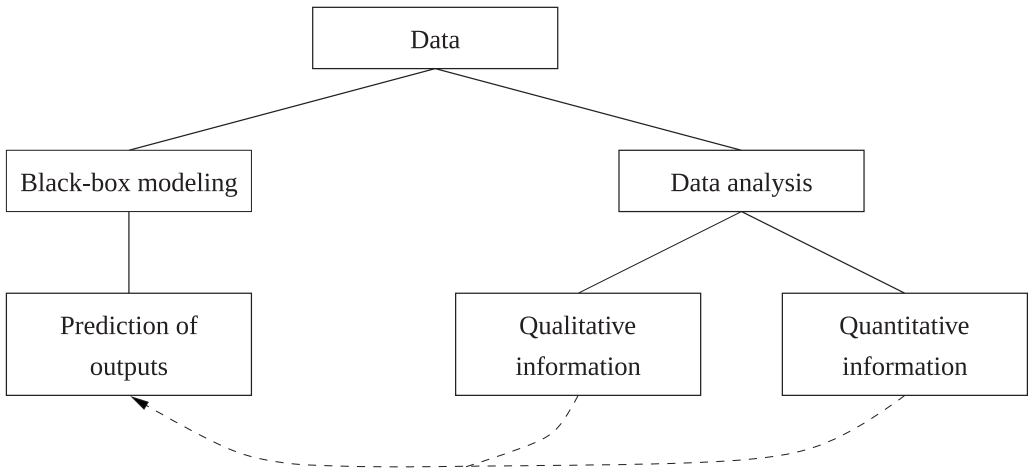

Figure 1. Different possibilities of process data treatment.

Black-box models build some type of regression model between the inputs and outputs of a system based on measurement data. After that, the models can be used to predict system outputs if the inputs are known. For example, Rudd, Dayal, and Musavi *(1–3)* show that artificial feed-forward neural networks are suggested for predicting pulping processes. The disadvantage of black-box models, however, is that they give little information about the dependencies of the variables; it is also difficult to get a general view of the system's behavior.

In data analysis, useful information regarding process data is extracted without modeling the system. Only process measurements made during normal operation of the process are considered. This approach needs to be distinguished from process experiments, where inputs are intentionally varied to determine the effect of the changes on the output. Data analysis could merely be used to ascertain reasonable setups for the experiments.

Data analysis

The application areas of data analysis are those processes or phenomena that—due to their complexity or nature—are impossible to model analytically. The advantage of the analysis is that neither process modifications nor experiments are required, and general methods independent of the application can be used.

Information of two different types may be acquired. *Qualitative* information can be obtained using a data representation that is easy to understand and interpret. This is usually done using data-visualization techniques. An example of *quantitative* information (i.e., numerical information) is the correlation coefficient that depicts the strength of a linear dependency between two variables.

Because the whole analysis is based on data, the measurements must be reliable. Signal noise is usually not a problem, because it may be reduced using filtering techniques. The real problem is sensor faults. A good example is signal non-stationarity caused by slow drifting of dirt on the surface of the measurement sensor.

Even though many methods can be applied in data analysis, the processing of the measurements before the analysis is usually done in similar ways. Different stages of data processing are presented in **Fig. 2**.

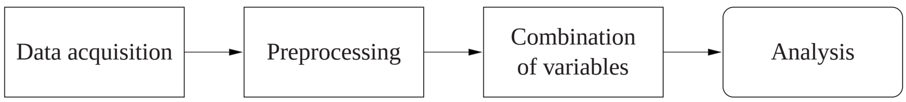

Figure 2. Data processing steps that need to be carried out before the analysis. Typical preprocessing operations are removal of erroneous measurements, noise filtering, and compensation of delay between measurements. Combination of variables refers to a refinement of data, where new variables are computed based on the original ones.

In this paper, an artificial neural network, the Self-Organizing Map (SOM), is applied in data analysis. The use of the SOM enables the application of an exceptionally large number of data-visualization techniques.

The Self-Organizing Map

The Self-Organizing Map (*4*) is a neural network structure consisting of neurons arranged in a two-dimensional regular grid. Each neuron *i* is represented by a *d*-dimensional model vector m_i.

The SOM is trained using a set of *d*-dimensional training (input) vectors. In the training, the model vectors settle in the input space so that they roughly approximate the probability density of the input data. The presentation of the data using the SOM also has another attractive property. The mapping *orders* the model vectors so that neighboring model vectors on the map grid resemble each other.

Training algorithm

The training consists of two phases: initialization and training (see **Fig. 3**). In the initialization phase, the model vectors of the map are given initial values, which may be random. However, this is not a good choice due to fact that the training phase can be done in fewer steps if the model vectors are initially roughly ordered.

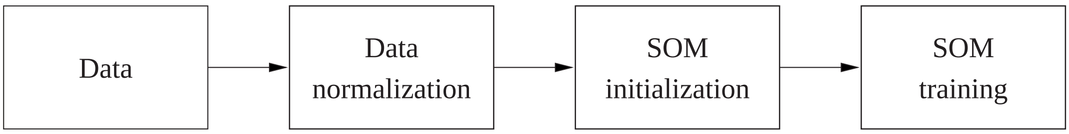

Figure 3. Training of the SOM. Before training, the variables of the input data are typically normalized to unit variance to guarantee each variable equal contribution in the map ordering.

The iterative training algorithm is as follows:

Step 1: Choose M, the number of training steps, and set n=0.

Step 2: Choose $N(0)$ and $\alpha(0)$, the initial values for the neighborhood radius $N(n)$ and the adaptation coefficient $\alpha(n)$. $N(n)$ and $\alpha(n)$ are decreasing functions of time so that at the end of the training, the neighborhood radius $[N(M)]$ is 1 and the adaptation coefficient $[\alpha(M)]$ is 0.

Step 3: Randomly choose a vector x among the training vectors.

Step 4: Find the best-matching unit (BMU) c, whose model vector m_c is closest to the vector x:

$$\| x - m_c \| = \min_i \{\| x - m_i \|\} \qquad (1)$$

Usually, the squared Euclidean distance is used:

$$\| x - m_i \| = \sum_{j=1}^{d} (x_j - m_{ij})^2 \qquad (2)$$

Step 5: Update model vectors of the BMU and its neighbors:

$$m_i (n + 1) = \begin{cases} m_i (n) + \alpha(\mathrm{n}) [x - m_i (n)], & i \in N_c (n) \\ m_\mathrm{i} (\mathrm{n}), & i \notin N_c (n) \end{cases} \qquad (3)$$

Step 6: Set $n = n + 1$. If $n = \mathrm{M}$, stop; otherwise, return to Step 3.

Kohonen (*4*) suggests that the training algorithm should be applied twice. In the rough-ordering phase, $N(0)$ is about half of the map size and $\alpha(0)$ equals 0.5. In the fine-tuning phase, $N(0)$ is small (e.g., 3) and $\alpha(0)$ is small (approximately 0.05).

SOM in data analysis

The SOM is an effective tool for data analysis. It has been successfully applied in various engineering applications: pattern recognition, image processing, process monitoring, control, and fault diagnosis (*5–7*).

An SOM trained using measurement data of a process is, in a sense, a representation of the process states with respect to process variables used. Each map unit depicts an operating point of the process.

The greatest advantage of the SOM in analyzing measurement data is efficient visualization of the data (*8–10*). The presentation can be obtained by visualizing the model vectors of the map or the training data using the model vectors; traditionally, only the data are visualized. See **Fig. 4** for data presentation schemes.

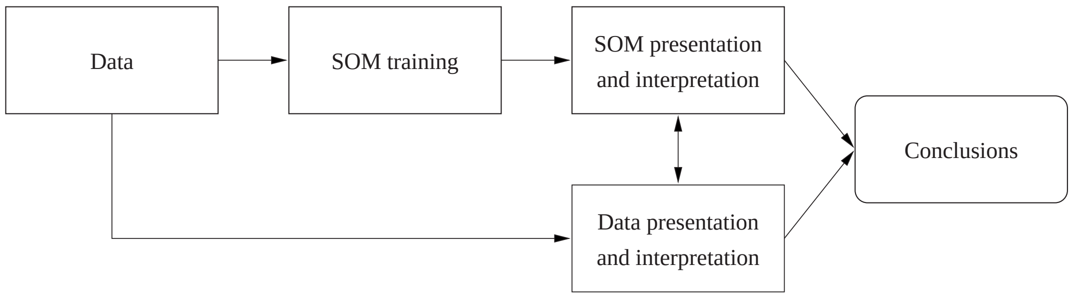

Figure 4. Data presentation schemes.

In this work, the following visualization methods are applied:

- *Component plane representation* shows the distribution of one model vector component (i.e., one variable) on the map. By displaying all planes (distributions) at the same time, one may roughly observe relationships between the variables and distinguish different operating states of the process. (See Fig. 7.)

- In *continuous coloring*, each map unit is assigned a color, which is similar in neighboring units. The following techniques, based on coloring, are used in this paper:

  (a) Two variables are selected. Corresponding components of the model vectors are drawn in a scatter plot, and each point is dyed using the corresponding map unit color. The color coding enables one to study the dependence of two variables in different process states. (See Fig. 8.)

  (b) The coloring can also be used in visualizing the *time series*: The BMU of each sample is determined, and the corresponding points in the series are dyed using the color of the BMU (See Fig. 9.)

Case study: Analysis of a continuous digester

The continuous digester

In a case study, the behavior of a continuous vapor phase digester at UPM-Kymmene's Pietarsaari pulp mill was studied. An illustration of the digester and separate impregnation vessel is shown in **Fig. 5**.

Presteamed wood chips, together with black and white liquor, are fed into the impregnation vessel. After impregnation, the chips are fed into the digester. At the top of the digester, the chips are heated to cooking temperature using steam, and the pulping reaction—removal of lignin—begins. The cooking ends at the extraction screens by displacement of the hot cooking liquor by cooler wash liquor. The wash liquor is injected to the digester through the bottom nozzles and bottom scraper. The liquor moves countercurrently to the chip flow and performs diffusion (high-heat) washing of the chips.

There is no central pipe in the digester; the cooking chemical (alkali) is fed into the impregnation vessel, at the digester top, and in the wash circulation. Another feature of the digester is an extra pair of screens, which are used as extraction screens before modifying the digester for higher capacity (from 750 to 1100 o.d. metric tons/day). Now, the screens are used for extracting small amounts of liquor for transfer circulation, like the top screen.

Analysis of digester measurements

Problems were observed in the digester operation: Pulp consistency in the digester outlet dropped. In those situations, the kappa number was smaller than the target value; digester chip levels increased and black liquor flow out of the digester through the extraction screens decreased.

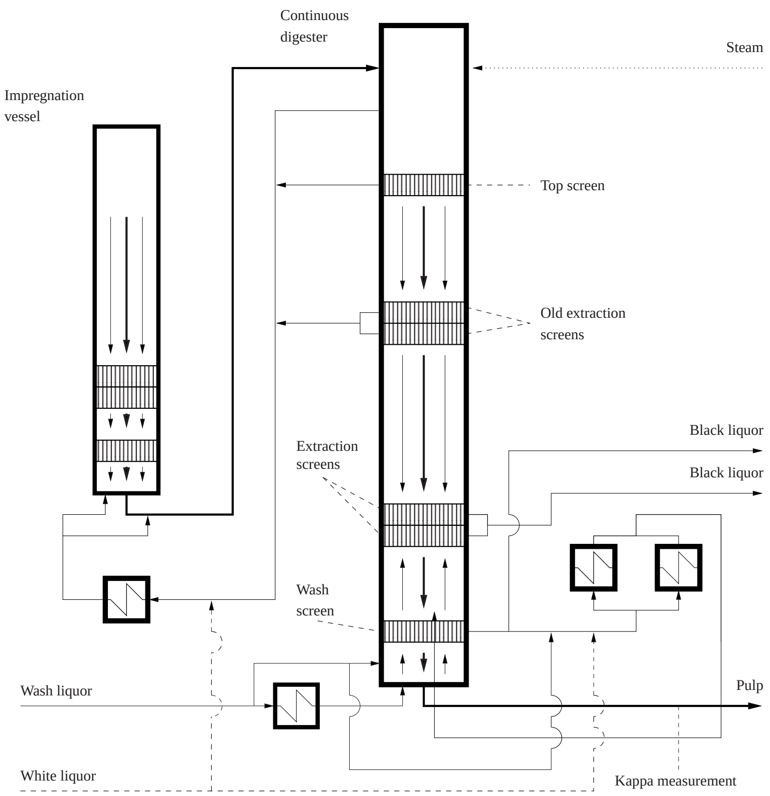

Figure 5. The continuous digester and the impregnation vessel. The cooking and wash liquor flows are marked by thin lines and the chip flow by the thick line. The four square-shaped symbols are heat exchangers.

An example of measurement signals in such a situation is presented in **Fig. 6**. Each signal point describes one 10-min average. Thus, the curves describe the behavior of the digester during about 1 week. At the end of the period, the digester is so faulty that the production speed must be decreased by 100 o.d. metric tons/day.

The signals were collected from the mill's automation system. The measurement period was selected so that there were no errors in the measurements. In preprocessing, the signals were delayed with respect to each other using known digester delays. Because the signals were already averaged, no further noise reduction by filtering was required.

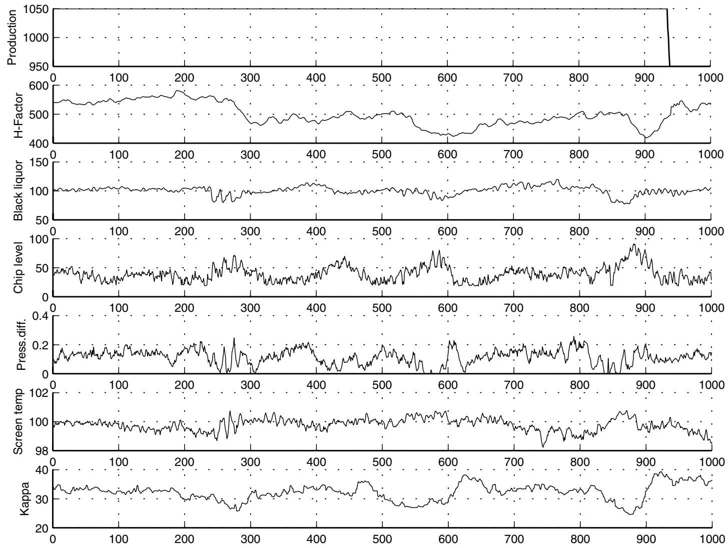

Figure 6. Measurement signals from top to bottom are fiber line production speed (o.d. metric tons/day), H-factor of the digester, black liquor flow through the extraction screens (L/s), chip level (%), pressure difference between the digester bottom and blow line (bar), "screen temp." (%), and kappa no. Variable "screen temp." is the percentage of the lower extraction screen temperature compared to the temperature of upper extraction screen. A value of more than 100% indicates that the pulping is not stopped at the screens as it should be.

In **Fig. 7**, the component planes of an SOM trained using the signals of Fig. 6 are presented. In the lower part of the map, especially in the right corner, the kappa number is low, the screen temp. is high, the chip level is high, and the black liquor flow is low. This means that the problematic states are mapped into that part of the map.

Then consider **Fig. 8**, where the map of Fig. 7 is presented so that each map unit is assigned a color. The faulty states are given green and yellow hues. In the four different scatter plots of Fig. 8, the variables H-factor, screen temp., chip level, and black liquor flow are on the *y* axes. The *x* axis of each plot shows the kappa number.

The scatter plots indicate that there is little correlation between kappa number and H-factor *in the faulty states*. But—to some extent—the other plotted variables show some dependency with kappa number. However, in the nonfaulty states marked by blue, magenta, and dark colors, the H-factor is clearly dependent on the kappa number. This is because, in a faulty situation, the downward movement of the chip plug in the digester slows. This is due to fact that the plug is so tightly packed at the extraction screens that the wash liquor cannot pass it. There are two consequences: The wash liquor slows the downward movement of the plug and the pulping reaction doesn't stop.

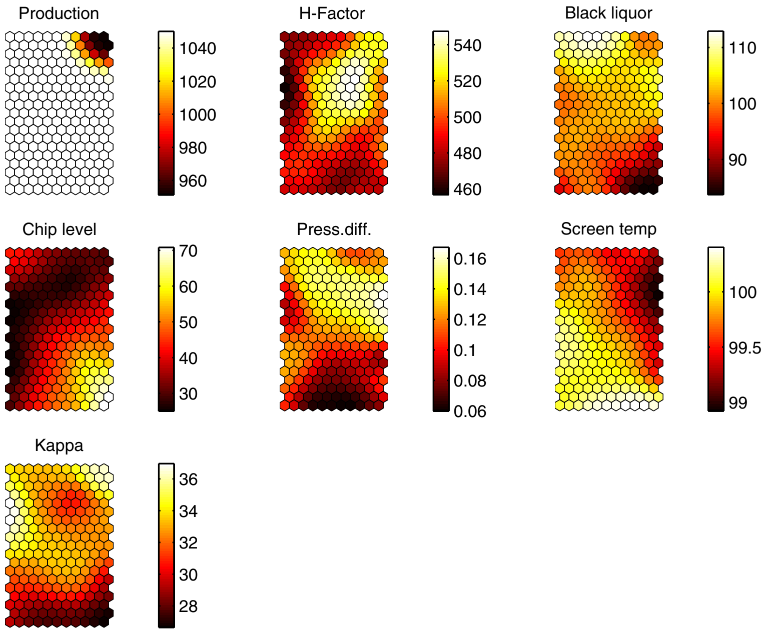

Figure 7. Component planes of the SOM trained using signals of Fig. 6.

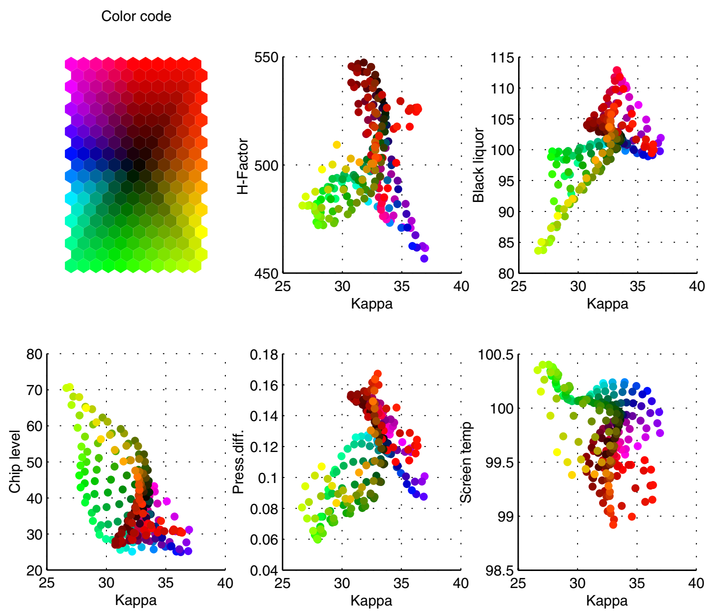

Figure 8. Color coding of the SOM with four scatter plots.

Because the cooking continues, the kappa number becomes too small. In addition, the H-factor–based digester control fails, because assumed constant cooking times for chips become longer due to the slowing of the chip plug movement.

Finally, in **Fig. 9**, the signals of Fig. 6 are colored using the coding presented in Fig. 8. The problematic control areas with green and yellow hues are now easy to see: They occur approximately at times 250–300, 500–600, and 850–900 seconds.

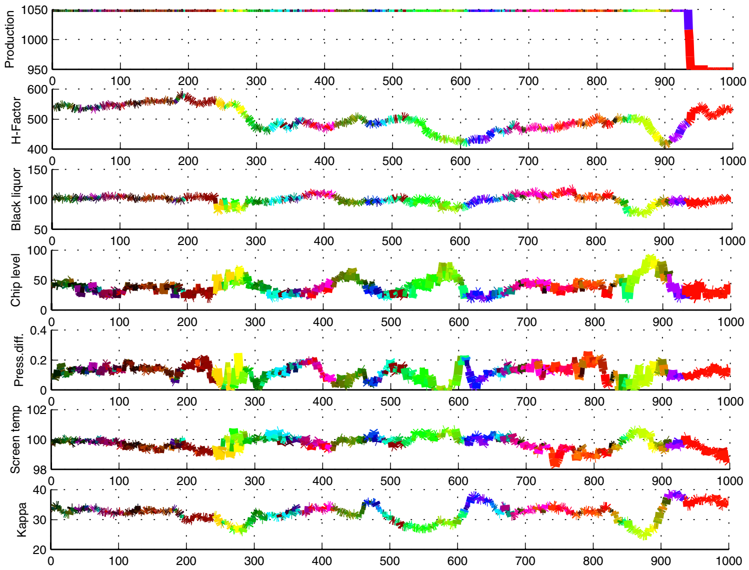

Figure 9. Color coding of the signals of Fig. 6.

CONCLUSIONS

A data analysis approach is useful in problems where the system or phenomenon of interest is difficult to analyze due to its complexity or nature. In normal operation, the behavior of the continuous digester can be modeled based on first principles (*11*, *12*). In the faulty states, however, an analysis of the pulping data is the only way to investigate the behavior of the system.

The self-organizing map can be effectively used to find and visualize correlations between process variables in different operational states of the process. In faulty states of the continuous digester, variables that normally have little effect on the pulp quality (e.g., chip level) seemed to affect the pulping. This was due to the fact that they correlated with chip plug movement in the digester. In the problematic situations, the H-factor–based kappa number control failed due to the increased residence time of the chips in the digester; this time is assumed to be approximately constant at constant production speed. In other words, the H-factor was larger than the control system expected it to be.

It should be emphasized that the variables of the case study were determined after an iterative data-mining process. Several tests were made and interpreted using the knowledge of process experts. In the beginning, there were dozens of measurements that were later reduced to the variables presented.

ACKNOWLEDGMENTS

The author wishes to thank the UPM-Kymmene Wisaforest pulp mill for the pulping data and the UPM-Kymmene Pulp Center for help in interpreting the results. Financial support by the Technology Development Centre of Finland and UPM-Kymmene is gratefully acknowledged.

LITERATURE CITED

1. Rudd, J. B., *Proceedings of the CPPA 80th Annual Meeting, Technical Section,* Vol. B, CPPA, Montreal, 1994, p. 169.
2. Dayal, B. S., MacGregor, J. F., Taylor, P. A., *et al*., *Pulp Paper Can*. 95(1): 26(1994).
3. Musavi, M. T., Coughglin, D. R., and Qiao, M., *Proceedings of the 1995 IEEE International Symposium on Circuits and Systems*, Vol. 3, IEEE, Piscataway, p. 1716.
4. Kohonen, T., *Self-Organizing Maps*, Springer, Berlin, Heidelberg, 1995.
5. Kohonen, T., Oja, E., Simula, O., *et al.*, *Proceedings of the IEEE* 84(10): 1358(1996).
6. Simula, O. and Kangas, J. in *Neural Networks for Chemical Engineers* (A. Bulsari, Ed.), Elsevier, Amsterdam, 1995, Chap. 14, p. 371.
7. Tryba, V. and Goser, K. in *Artificial Neural Networks* (T. Kohonen, K. Mäkisara, and O. Simula, Eds.), North-Holland, Amsterdam, Netherlands, 1991, p. 847.
8. Himberg, J. in *Intelligent Data Engineering and Learning* (L. Xu, L. W. Chan, I. King, *et al*., Eds.), Springer, 1998, p. 427.
9. Kaski, S., Venna, J., and Kohonen, T. in *Visual Explorations in Finance with Self-Organizing Maps* (G. Deboeck and T. Kohonen, Eds.), Springer-Verlag, 1998, Chap. 14.
10. Vesanto, J., Himberg, J., Siponen, M., *et al.*, *Proceedings of the 5th International Conference on Soft Computing and Information/Intelligent Systems* (T. Yamakava and G. Matsumoto, Eds.), 1998, Vol. 1, p. 64.
11. Härkönen, E., *Tappi J*. 70(12): 122(1987).
12. Gustafson, R. R., Sleicher, C. A., McKean, W. T., *et al.*, *Industrial & Engineering Chemistry Process Design and Development* 22(1): 87(1983).

Received for review: February 25, 1999
Accepted: December 21, 1999

This paper was accepted for publication and abstracting in the July 2000 issue of TAPPI Journal.