ESANN'2009 proceedings, European Symposium on Artificial Neural Networks - Advances in Computational Intelligence and Learning. Bruges (Belgium), 22-24 April 2009, d-side publi., ISBN 2-930307-09-9.

Self-organising map for large scale processes monitoring

Edouard Leclercq, Fabrice Druaux, Dimitri Lefebvre

Groupe de Recherche en Electrotechnique et Automatique du Havre
Université du Havre, Le Havre, France

Abstract : A feed-forward neural network is proposed for monitoring operating modes of large scale processes. A Gaussian hidden layer associated with a Kohonen output layer map the principal features of measurements of state variables. Subsets of selective neurons are generated into the hidden layer by means of self adapting of centers and dispersions parameters of the Gaussian functions. The output layer operates like a data fusion operator by means of adapting the hidden-to-output matrix of weights through a winner takes all strategy. The algorithm is tested with the Tennessee Eastman Challenge Process. The results prove that the proposed neural network clearly maps the different operating modes.

1 Introduction

Due to the complexity of industrial processes, and the need to ensure production quality and reliability, engineers have to develop complex monitoring strategies. The frequently used strategies are principal component analysis or partial least-squares analysis. These methods strongly depend on data characteristics. Analytical approaches like parameter estimation and observer-based method can also be used when a detailed mathematical model of the process is available. For large and multivariate data base and when no model is available, fuzzy or neural methods can be implemented [1, 2, 3, 4].

In this paper a multilayer neural network is developed for monitoring operating modes of large scale industrial processes. The main objective of the proposed work is the conception of a selective processing tool for the transformation of a large time varying data base into a two dimensional map representative of the operating mode of an industrial process.

The network is feed-forward and its structure consists of three successive layers. The input layer carries out the data-gathering. Then a hidden layer, made of self-tuning Gaussian functions, performs a data selection and finally a Kohonen mapping is performed through the output layer. Such transformation that allows us to identify the operating mode should be seen as a preliminary step to perturbation or fault detection and to identification.

The simulations of the monitoring of the Tennessee Eastman Challenge Process operating mode, point out the performances of the proposed algorithm.

2 Network structure and learning algorithm for monitoring

The proposed feed-forward structure consists of three layers (fig. 1). The input layer gathers the data to be mapped. The data are the time varying delayed measurements of N state variables of a large scale process.

The hidden layer adapts itself in such a way that the different operating modes of the process are associated with a subset of neurones. Each subset can be considered as a signature of a particular operating mode.

The self-organising output layer, inspired from Kohonen neural network, maps the data into a two dimensional representation. The features of the resulting output representation are representative of the current operating mode.

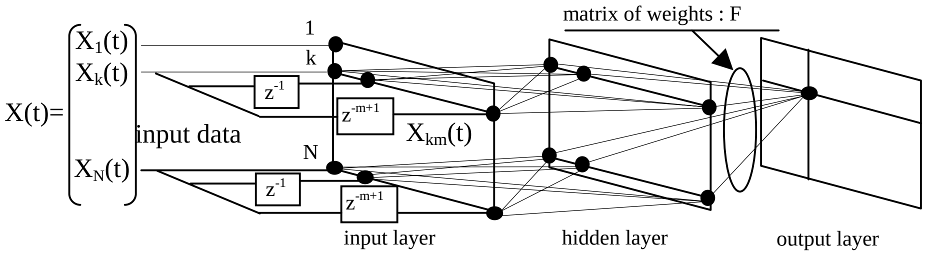


Fig 1. Neural network structure for data mapping

2.1 Input layer

The input of the network is a two dimensional layer. The number N of rows of this layer equals the number of the different measured variables. In the row k, the first neuron receives the measurement of the variable X_k at time t, $X_{k1}=X_k(t)$. The neuron j (j=2,..,m) receives delayed measurement X_{kj} of the same variable, $X_{kj}=X_k(t-j+1)$.

2.2 Hidden layer

This layer is composed of Gaussian functions centered on C_{kj} with a dispersion σ_{kj}.

$$H_{kj} = \exp(-(S_k - C_{kj})^2 / 2\sigma_{kj}^2) / \sigma_{kj}$$

The entry S_k of this function corresponds to the mean value of the m inputs of the row k of the input layer. It results a reduction of the sensitivity to measurement noises.

$$S_k = (\sum_{q=1}^{m} X_{kq}) / m$$

The initial value of the centers associated with each neuron are divided from the min to the max value of the considered input variable. Initial values of the dispersion parameters are chosen equal to the difference between two successive centers.

$$C_{kj} = \left(j\left(X_{k_{max}} - X_{k_{min}}\right)/(m+1)\right) + X_{k_{min}} \; ; \forall j = 1,..,m \quad \sigma_{kj} = \left(X_{k_{max}} - X_{k_{min}}\right)/(m+1);$$

The self-adapting algorithm consists in modifying the center and the dispersion of the winner neuron after a lateral excitation/inhibition process of each row at each time t:

$$\text{for } k = 1,..,N, \text{ and } j = \arg\max\{H_{kq},\ q = 1,..,m\}$$

$$C_{kj_{new}} = (C_{kj_{old}} + \sum_{q=1}^{m} X_{kq})/(m+1)$$

$$\sigma_{kj_{new}} = \max\left[\sigma_{k_{lim}}\ ;\ \sqrt{\left[\sigma^2_{kj_{old}} + \sum_{q=1}^{m}\left(X_{kq} - C_{kj}\right)^2\right]/(m+1)}\right];$$

$$\text{with } \sigma_{k_{lim}} = (X_{k_{max}} - X_{k_{min}})/(\alpha(m+1))$$

The initial values lead to a fair distribution of the centers of the Gaussian functions into the interval $[X_{k_{min}}, X_{k_{max}}]$. The adaptive algorithm makes the neurons more selective as the learning phase progresses. To avoid excessive influence of noise measurement on the adaptation of hidden units, a lower bound $\sigma_{k_{lim}}$ is set, α is an arbitrary coefficient. In order to set the output of the hidden neurons to finite values, each row is normalised at each step time. Steady states of the measured variables characteristic of the operating modes are progressively associated with a set of neurons in the hidden layer. More often they are winners, more selective neurons become. Once the learning stage is completed, each row of the hidden layer is divided into several subsets of neurons associated with a particular operating mode.

2.3 Mapping output layer

The output layer is a $N_s x N_s$ matrix randomly initialised. The output mapping M={$M_{kq;}$ k=1,..,N_s; q=1,..,N_s} is performed by linear neurons associated with a process of lateral excitation-inhibition and a winner takes all algorithm.

Let r defines the radius of the excitation neighbourhood, neurons inside this neighbourhood are excitatory neurons and those outside are inhibitors.

$$M_{kq} = \sum_{l=1}^{l=N}\sum_{h=1}^{h=m} F_{lh}^{kq} H_{lh} + \sum_{\substack{d\le r\\ d\ne 0}} \delta_1 (\sum_{l=1}^{l=N}\sum_{h=1}^{h=m} F_{lh}^{k'q'} H_{lh}) + \sum_{d>r} \delta_2 (\sum_{l=1}^{l=N}\sum_{h=1}^{h=m} F_{lh}^{k'q'} H_{lh});$$

$$\text{where k' and q' are such that } d = \sqrt{(k-k')^2 + (q-q')^2}$$

The weights F_{lh}^{kq} are randomly initialised into [-1,+1], δ_1 and δ_2 are respectively an arbitrary positive coefficient for excitatory neurons and negative one for inhibitors. At each step time, the output values are included into the interval [-1,+1].

$$M = M/\max\{abs(M_{kq});\ k=1,..,N_s;\ q=1,..,N_s\}$$

The winner takes all algorithm is:
for (i,j) such as $M_{ij} = \max\{M_{kq};\ k = 1,..,m;\ q = 1,..,m\}$, and for u=1,..,N,
and v such as $H_{uv} = \max\{H_{uz},\ z = 1,..,m\}$,

$$F^{ij}_{uv_{new}} = F^{ij}_{uv_{old}} + \mu\left(F^{ij}_{uv_{old}} - H_{uv}\right), \text{ where } \mu \text{ is the learning rate}$$

During the learning phase, the weights of the links from subsets of the hidden layer associated with the different operating mode of the process adapt themselves. The result is a mapping of the outputs such that particular features are associated with each operating mode.

3 Tennessee Eastman Challenge Process (TECP) simulation

The TECP is a simulating tool proposed to the community of engineers and scientists by the Eastman Corporation [5, 6]. It consists of a chemical reactor, a separator, a stripper, a condenser and a compressor. It comprises 50 state variables, 41 measured variables and 12 manipulated variables. It was developed and studied by numerous engineers and researchers working in modelling and control area [7-10]. The 7 operating modes are precisely described in [9, 10]. These modes are defined for different mass ratio R=G/H, where G and H stand for the rates of output products, and for different objectives such as purge rate minimisation, production cost minimisation, etc…

In that work, stabilisation and control of the TECP is achieved using a neural controller previously developed by our laboratory [11-13]. For the monitoring approach, 4 variables were chosen (temperature and pressure of the reactor, temperature and pressure of the separator).

A 80 hours learning phase is performed which consists of 8 successive transitions between 3 different modes: mode 0 (base case R=50/50) , mode 1 (R=50/50; minimum operating cost) and mode 2 (R=10/90; minimum operating cost) (table 1). The learning parameters are arbitrarily set to: r=2 ; δ_1=0.1 for excitatory neurons and δ_2 = -0.005 for inhibitors, μ = 0.01, α = 10 . The sampling period during simulation is 0.02 h.

Time (h)	0-3	3-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80
mode	0	2	0	1	2	1	0	2	0

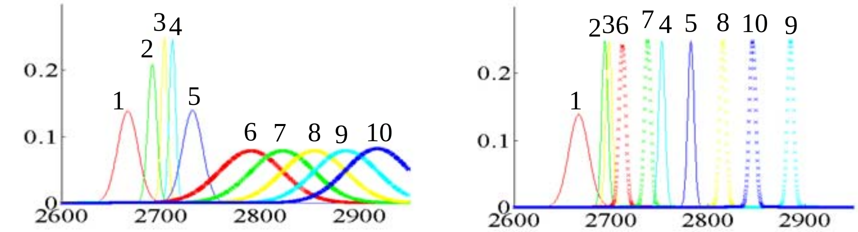
Table 1 : Operating mode sequence 1 for learning stage

4 Results

While the network is learning, center of Gaussian functions of each hidden neuron moves to get closer to one of the steady state values of the process. Subsets of neurons provide a specific signature for each steady state (fig. 2). Decreasing the dispersion parameters makes each neuron more sensitive to the input values in a very narrow neighbourhood.

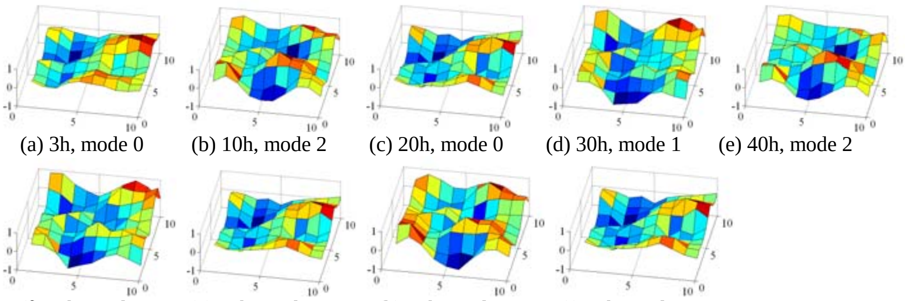
After learning has been completed, a running stage for the learning data base (sequence 1) demonstrates the ability of the system to distinguish the different operating modes. When the steady state becomes the same, the output map presents features very close to each other (fig. 3).

Then a second operating mode monitoring is simulated with sequence 2 (table 2). Using maps of fig 3, three representative matrices (M0, M1, M2), are selected as signatures of the different operating modes. Performance is evaluated by comparisons between the output map M and M0, M1 and M2. At each step time three square errors E0, E1 and E2 are computed between M and respectively M0, M1 and M2. The current output map M is labelled 0, 1 or 2 by selecting the lowest value among E0, E1 and E2 (fig.4). The mode 0 is perfectly identify while during mode 1, 52% misclassifications into mode 2 are observed and for mode 2, 6% misclassifications into mode 1 are observed. This misclassifications are principally due to the fact that the chosen measurement variables during mode 1 and 2 present very nearest mean values.



(a) (b)

Fig 2. Example of neuron gathering in Gaussian layer. Neurons associated with the reactor pressure measurements, the pressure scale is [2600 kPa, 2950 kPa]. (a) after 3h learning. (b) after 80h learning. The number attached to each curve corresponds to the number of the corresponding neuron on the hidden layer.



(a) 3h, mode 0 (b) 10h, mode 2 (c) 20h, mode 0 (d) 30h, mode 1 (e) 40h, mode 2

(f) 50h, mode 1 (g) 60h, mode 0 (h) 70h, mode 2 (i) 80h, mode 0

Fig 3. Output layer mapping for the learning data base.

Time (h)	0-5	5-20	20-50	50-70	70-100	100-130	130-160
mode	0	1	0	2	1	0	2

Table 2 : Operating mode sequence 2 for running stage

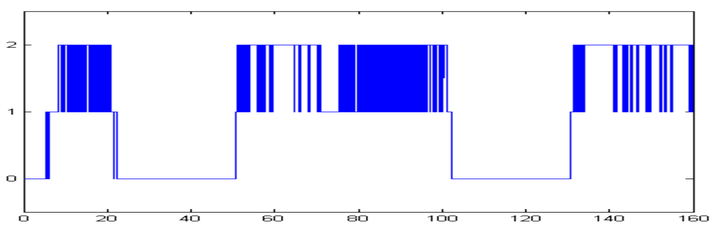


Fig 4. Real time identification of the operating mode at each step time for sequence 2

5 Conclusion

A multilayer neural network was developed for industrial process operating mode monitoring. The main objective of this work is the conception of a selective processing tool for the transformation of a large time varying data base into a two dimensional mapping of the operating mode.

The three layers feed-forward network consists of an input layer that carries out the data-gathering, a hidden layer made of self-tuning Gaussian functions that performs a data selection and finally a Kohonen mapping through the output layer.

Simulation of the monitoring of the different operating modes of the Tennessee Eastman Challenge Process points out performances of the proposed algorithm.

This transformation performs a labelling of the operating mode and for that reason can be considered as a first step for a larger processing tool for perturbation and/or fault detection and identification. Our future work will consist of an extension of the methods to larger data base to avoid misclassifications and then, disturbance detection will also be studied.

References

[1] D. Wlodzislow, G.H.F. Diercksen, "Feature space mapping as a universal adaptive system" , Computer Physics Communications, 87, pp. 341-371, 1995.

[2] A.P. Leitão, S. Tilie, M. Mangeas, J.-P. Tarel, V. Vigneron, S. Lelandais, "Road singularities detection and classification", proceedings of ESANN'03, pp. 301-306, Gruges, Belgium, 2003.

[3] V. Uraikul, C.W. Chan, P. Tontiwachwuthikul, "Artificial intelligence for monitoring and supervisory control of process systems", Engineering Applications of Artificial Intelligence, 20, pp. 115-131, 2007.

[4] T. Alhonnoro, M. Sinola, "Feature selection on process fault detection and visualization", proccedings of ESANN'08, pp. 265-270, Bruges, Belgium, 2008.

[5] N. L. Ricker, Tennessee Eastman Challenge Archive: Chemical Engineering University of Washington, Seattle, WA,USA,1993. http://depts.washington.edu/control/LARRY/TE/download.html,

[6] D.E. Rivera, Tennessee Eastman Challenge Problem in MATLAB, Control Systems Engineering Laboratory of Arizona State University, USA, 1998 http://www.eas.asu.edu/~csel/Software-TennEast.htm,

[7] J.J. Down, E.F. Vogel, "A plant-wide industrial process control problem", Computer and Chemical Engineering", 17 (3), pp. 245-255, 1993.

[8] N.L. Ricker, "Optimal steady-state operation of the Tennessee Eastman Challenge Process", Computer Chemical Engineering, 19 (9), pp. 949-959, 1995.

[9] H-Q. Wang, Y-C. Gao, K. Song, "NN-PLS, On-line component soft-analyzer for Tennessee Eastman Challenge Process" Proceedings of the 5th World Congres on Intelligent Control and Automation, June 15-19, Haugzhen, China, 2004.

[10] A. Singal, D.F. Seborg, "Evaluation of a pattern matching method for the Tennessee Eastman Challenge Process", Journal of Process Control, 16, pp. 601-613, 2006.

[11] S. Zerkaoui, F. Druaux, E. Leclercq, D. Lefebvre, "Multivariable adaptive control for non-linear systems: Application to Tennessee Eastman Challenge Process", ECC'07, Kos, Greece, 2007.

[12] S. Zerkaoui, F. Druaux, E. Leclercq, D. Lefebvre, "Improving the robustness and stability properties of neural adaptive control for non-linear systems", International Journal of Factory, Automation, Robotics and Soft Computing, Issue 2, 81-87, 2008.

[13] D. Lefebvre, S. Zerkaoui, F. Druaux, E. Leclercq, "Adaptive control with stability and robustness analysis for nonlinear plant wide systems by means of neural networks", in : Progress in Nonlinear Analysis Research, Ed, Erick T. Hoffmann, Nova Publishers, Chap. 8, pp. 187-212, 2008.