15th Triennial World Congress, Barcelona, Spain

APPLICATION OF THE DYNAMIC RBF NETWORK IN A MONITORING PROBLEM OF THE PRODUCTION SYSTEMS

Zemouri Ryad, Racoceanu Daniel, Zerhouni Noureddine

Laboratoire d'Automatique de Besançon, UMR - CNRS 6596
25, Rue Alain Savary – 25000 Besançon, France
rzemouri@ens2m.fr
daniel.racoceanu@ens2m.fr
zerhouni@ens2m.fr

Abstract: A new architecture of temporal neural network, called Recurrent Radial Basis Function is proposed. This new architecture of neural network take into account the temporal aspect of the data in a dynamical way. This functionality is obtained by input layer neurons self-connections. The RRBF network is validated on a dynamic monitoring problem by analyzing strongly varying sensors signals. The obtained monitoring model is able to divert false alarms and to anticipate the system operation in order to consider corrective actions, before undesired modes occur. *Copyright © 2002 IFAC*

Keywords: Neural networks, Radial base function network, Dynamic model, Fault detection, Production system, Preventive maintenance, Sensors.

1. INTRODUCTION

In order to optimize the production costs, a great number of modern industrial systems need to replace the systematic traditional maintenance by a conditional one, based on the on-line monitoring. This kind of on-line monitoring is thus able to prevent an abnormal operation before its occurrence and to divert false alarms (Basseville and Cordier, 1996).

The production systems monitoring methods can be classified in three categories (Bernauer and Demmou, 1995): Methods based on mathematical model, methods without such a model and methods based on symbolic knowledge of the process.

In this paper, a new neural network architecture called RRBF (Recurrent Radial Basis Function) is applied in a dynamic monitoring problem. Using a self-connection of the input neurons, the RRBF neural network is thus able to treat dynamic data (temporal aspect). This type of application can be compared to a problem of pattern recognition that does not require a formal model of the system. Consequently, the monitoring model using this RRBF network is able to detect a real degradation of machine performances and to turn down false alarms.

Before introducing the new neural model, a brief recall of radial basis function neural networks is presented.

2. RADIAL BASIS FUNCTION

The Radial Basis Function networks (RBF) are a three-layer networks derived from an interpolation technique named RBF interpolation. Used for the first time in the context of neuromimetic networks by Broomhead and Low (1988), this technique proves to be a fast and efficient one, in particular for the classification (Jodouin, 1994).

The principle of the method consists to divide the N-Dimensional space in different classes or categories. Every category possesses a core called prototype, and an influence field having the shape of a hyper sphere. Several prototypes can be associated to the same category. The classification consists in evaluating the distance between an N-dimensional input-vector and the prototypes memorized by the network, and see which influence field belongs this vector to (Fig.1).

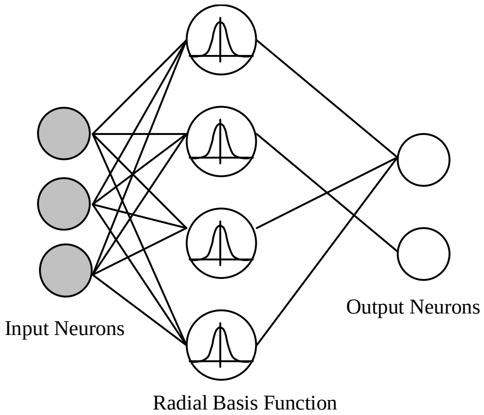

Fig. 1. Structure of a RBF network.

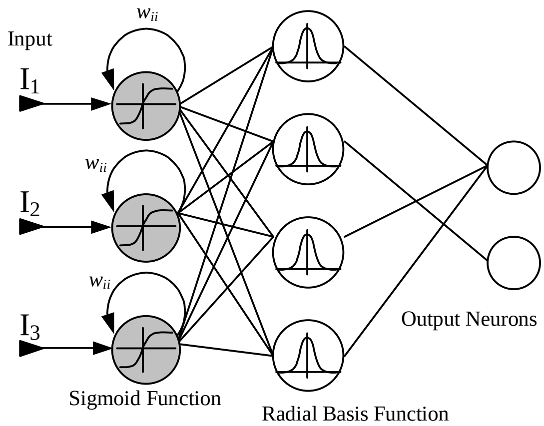

Fig. 2. RRBF Network.

The Radial function is maximal to the core, and generally decreases in a monotonous way with the distance. The RBF function used in this study is the radial Gaussian:

$$f_i(x) = e^{\frac{-d_i(x)^2}{\sigma_i^2}} \qquad (1)$$

with $d_i(x)=\|X-C_i\|$ measuring the distance between the input vector X and the prototype C_i ,
and σ_i the size of the influence field (standard deviation).

3. RECURRENT RADIAL BASIS FUNCTION NETWORK (RRBF)

3.1 Network architecture

The proposed RRBF neural network (Recurrent Radial Basis Function) uses an internal representation of the time (Chappelier, 1996; Elman, 1990). This property obtained with a self-connection of the input layer neurons gives a dynamic aspect to the RBF network (Fig.2).

This self-connection has been used on a Multi Layer Perceptron by Bernauer (1993) for the recognition of temporal sequences of an assembly system. The major inconvenience of this neural network is the complexity of the training process (Back Propagation Algorithm). Indeed, the parameters adjustment is very delicate and requires several tests and a good knowledge of the problem. The flexibility of the training process of the RRBF network (same training algorithm - RCE (Reilly, *et al.*, 1982) - as the RBF networks) represents an important advantage of this architecture.

3.2 Effect of the self-connection

Every neuron of the input layer makes a summation at the instant *t* between its input I_i and its output of the previous instant (*t-1*) weighted by the weight of the self-connection w_{ii}. Output result of the input neuron corresponds thus to the activation function:

$$\begin{gathered} a_i(t) = w_{ii}x_i(t-1) + I_i(t) \\ x_i(t) = f(a_i(t)) \end{gathered} \qquad (2)$$

where $a_i(t)$ and $x_i(t)$ are respectively the activation and the output of the neuron *i* at time *t*, w_{ii} is the weight of the self-connection of the neuron *i*, and *f* represents the activation function of the neuron *i* having the expression of the sigmoid below :

$$f(x) = \frac{1-\exp(-kx)}{1+\exp(-kx)} \qquad (3)$$

To study the effect of the self-connection of every neuron, the neuron input is equal to zero ($I_i = 0$) and the neuron output $x_i(t) = 1$. The neuron will evolve thus, without the influence of the external input ($I_i = 0$) (Bernauer, 1996). The evolution of the output neuron is :

$$x_i(t) = \frac{1-\exp(-kw_{ii}x(t-1))}{1+\exp(-kw_{ii}x(t-1))} \qquad (4)$$

The diagram of the figure 3 shows the evolution of the output of the neuron in time. This evolution depends on the gradient of Δ (inverse of the connection weight of the neuron w_{ii}) and also on the value of the *k* parameter of the activation function.

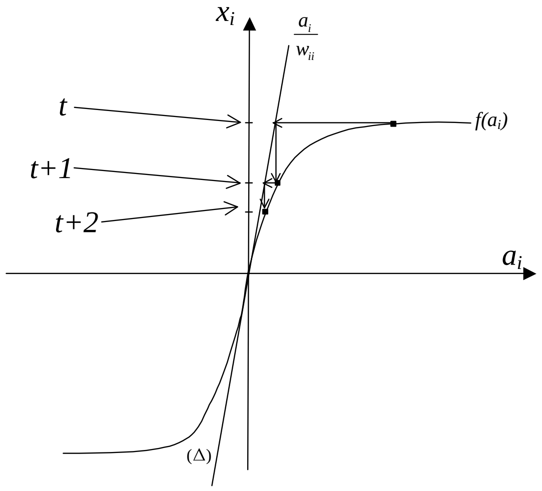

Fig. 3. The effect of the self-connection on the evolution of the neuron state.

3.3 Training Algorithm of the RRBF network

Input self-connection procures to the neuron a certain memory. This characteristic allows them to take into account the previous inputs and not only the inputs at the instant *t*. Each input $I_i(t)$ represents a calibrated signal obtained by a sensor of the production system.

The training algorithm used for the RRBF network is the RCE (Reilly, *et al.*, 1982), which introduces a new prototype when it is necessary and adjusts the influence field of existing prototypes in order to avoid conflicts. This training algorithm is more flexible than the one used by Bernauer and Demmou (1993). Otherwise, the problem of over-training met in back propagation algorithm doesn't have an effect in the RCE algorithm.

The RRBF network was already tested with success (Zemouri, *et al.*, 2001) for temporal sequences recognition. Each input neuron represents the occurrence of a sequence event. During the training process, events are presented to the network one by one, and the category is defined after the last event was presented to the network. Each radial neuron memorizes a prototype (vector sequence) and each neuron of the output layer represents a category (sequence). The only parameters to regulate are the weight of the self-connections (w_{ii}) and the size of the influence fields σ_i of the radial functions.

In the following paragraph, a validation of the neural model on a monitoring problem is presented. This application field put in evidence new properties of the RRBF, which seems very useful for production systems safety engineering.

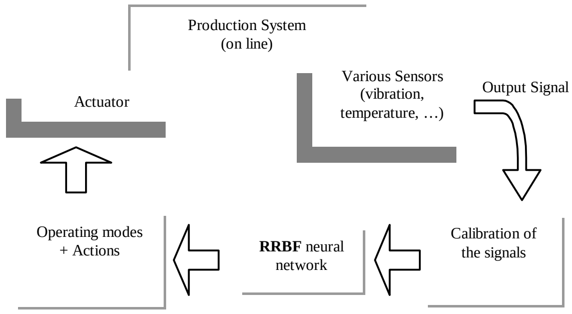

Fig. 4. General architecture of a neuronal monitoring system.

4. APPLICATION OF THE RRBF NETWORK IN A MONITORING TASK

4.1 Description of the monitoring model

The RRBF network is tested on a production system monitoring, using sensors signals (Fig.4). To simplify the model, only two operating modes and only one sensor signal are considered (Fig.5). Obviously, in practice the problem is much more complex, (several operating modes with a multitude of signals sensors) but the reasoning will be the same. The sensor signal *S(t)* represents the stimulus of the input neuron and each operating mode is represented by a neuron of the hidden layer. In the case of several signals, the neural model will have as many input neurons as sensor signals.

Figure 5 represents the network architecture with the two following operating modes:

- Operating mode 1 (a nominal operation mode),
- Operating mode 2 (a known failure mode).

Thanks to the self-connection, the RRBF network is able to take into account the temporal aspect of the input signal and thus to supervise its evolution. This characteristic procures to the network the capacity to distinguish between a false alarm and a permanent degradation in time (loss of performance). Figure 6 shows that the output *X(t)* of the looped (input) neuron is different for a same excitation value *S(t)*. The first represents degradation in time, while the second represents an abrupt change of the input signal.

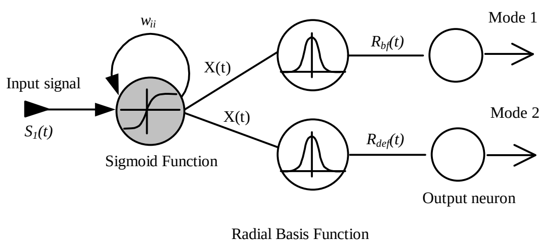

Fig. 5. Structure of the monitoring model.

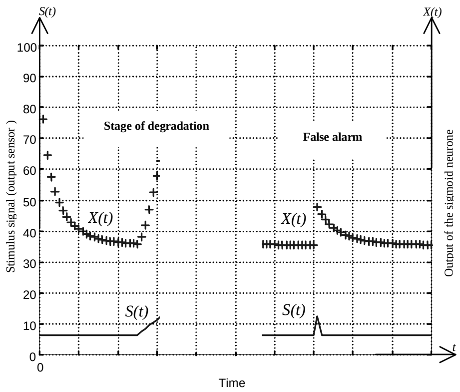

Fig. 6. Response of the monitoring model to a degradation stage and a false alarm.

Each neuron of the hidden layer is dedicated to an operating mode. The radial function of these neurons covers an operating range regulated using the influence ray σ_i. The training process is summarized to the adjustment of some parameters (certain are given by the manufacturer). These parameters are:

- A good calibration of the input signal to avoid the saturations zones of the input neuron activation function (sigmoid),
- The adjustment of the parameters of the input neuron activation function (k and w_{ii}),
- To position the radial functions on the system operating ranges. The prototypes of the two functions (x_{bf} and x_{def}) will be experimentally defined. These prototypes represent respectively the outputs of the looped neuron having input signal S_{bf} (average signal corresponding to the normal working mode) and S_{def} (average signal corresponding to the failure mode),
- Defining the size of the influence field of the radial functions.

The figure 7 shows the correspondence between the sensor signal *S(t)* and the RBF neurons outputs $R_{bf}(t)$ and $R_{def}(t)$.

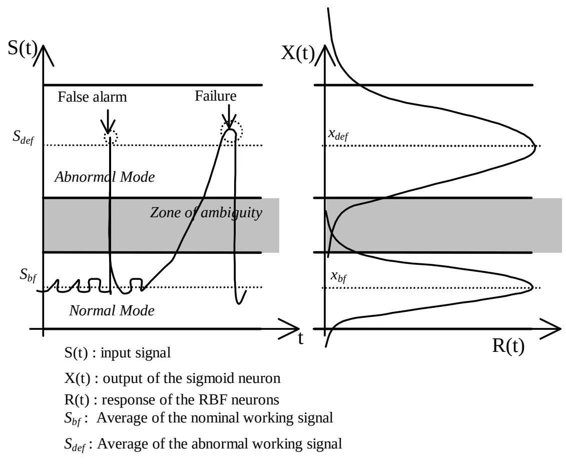

Fig. 7. Correspondence between the sensor signal and output RBF neurons.

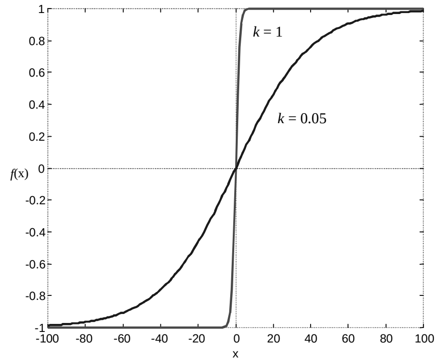

Fig. 8. Sensitivity of the neuron activation function according to the parameter *k*.

4.2. A simulation example

To apply the neural model in a monitoring problem, an output sensor signal *S(t)* of a system is simulated. The ranges of the two operating modes (normal and failure) represented by figure 7 are supposed known. The signal must be calibrate in a manner to avoid the saturation zone of the sigmoid activation function of input neuron (3). The width of the resolution zone depends on the parameter k (Fig. 8). An arbitrary width of hundred units (*S(t)* < 100) obtained for $k = 0{,}05$ is chosen. In order to give a longest storage capacity to the input neuron, the weight of the self-connection must be lower than inverse tangent at the sigmoid origin ($w_{ii} < 2/k$) (Bernauer *et al.*, 1993). The weight of this self-connection has the value $w_{ii} = 39$.

For an average input signal corresponding to the normal operation range $S_{bf} = 1$ and an input average signal corresponding to the failure mode $S_{def} = 6$, the respective outputs of the sigmoid neuron (multiplied by a coefficient 100) corresponding to the steady state of the equation (2) are: $x_{bf} = 35{,}48$, $x_{def} = 66{,}09$. The two corresponding radial functions are centered on the prototypes x_{bf} and x_{def} (Fig. 7). The influences rays σ_i of the two radial functions are given according to the width of the operating modes (Fig.7). For a width of the normal operation mode equal to 2 ($S(t) \in [\,0{,}2\,]$) and the failure mode one equal to 6 ($S(t) \in [\,3{,}9\,]$), the influences rays of the two functions have the following respective values $\sigma_{bf} = 10$ and $\sigma_{def} = 15$ (2).

To materialize the behavior of the monitoring neuronal model, four cases of a system operation are simulated :

Normal working Case. The case of a nominal operation where the signal *S(t)* is close to S_{bf}. The output *X(t)* of the input neuron is then equal to 35. This output is close to the neuron prototype corresponding to the normal operating range ($R_{bf} = 1$and $R_{def} = 0$).

Table 1 Case of a normal working situation

	S(t)	X(t)	R_{bf} (t)	R_{def}(t)	Working mode	Result
t	1	35	1	0	Normal operation	OK

Case of false alarm. Often false alarms are due to disturbances of various natures (acquisition disturbance). This disturbance signal generally does not persistence (Fig. 7). the neural network is insensitive to these abrupt disturbances. The table 2 shows the answers of the network for this kind of disturbance. At the moment of the perturbation ($t=t_1$), the two output neurons give approximately the same answer, corresponding to a possible failure. At the next step ($t=t_1+1$) the input signal return to its normal value. The neuron response corresponding to the correct working range tends to grow while the failure range response tends to decrease (Fig 9). This behavior is equivalent to false alarm detection.

Table 2 Cases of a false alarm

	S(t)	X(t)	R_{bf} (t)	R_{def}(t)	Working mode	Result
$t<t_1$	1	35	1	0	100% normal working	OK
$t = t_1$	7	47	0.19	0.23	19% normal working mode 23 % failure working mode	Possibility of existence of a failure
$t = t_1+1$	1	45	0.33	0.15	33% normal working mode 15% failure working mode	Possibility of false alarm
$t = t_1+2$	1	43	0.46	0.11	46% normal working mode 11% failure working mode	OK
$t = t_1+3$	1	42	0.58	0.08	58% normal working mode 8% failure working mode	OK

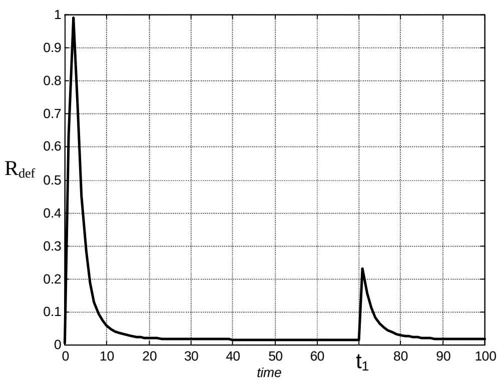

Fig. 9. Response of the RBF neuron corresponding to the failure mode. No alarm is generated in the case of acquisition disturbance.

Case of a progressive degradation. The case of progressive degradation induces a decreasing output corresponding to the normal working mode neuron and a growth of the failure mode neuron output, until the detection of the failure (Fig.10). The neural network is able to detect the failure before the signal reaches its maximum (critical) value (*S(t)*=7). The monitoring model is thus able to anticipate the system operation in order to consider corrective actions, before undesired modes occur.

Table 3 Progressive degradation case

	S(t)	X(t)	R_{bf} (t)	R_{def}(t)	Working mode	Result
$t<t_2$	1	35	1	0	100% normal working mode	OK
$t = t_2$	2	37	0.93	0.028	93% normal working mode 3 % failure working mode	loss of performances
$t = t_2+1$	3	41	0.65	0.07	65% normal working mode 7% failure working mode	Degradation of performances
$t = t_2+2$	4	46	0.26	0.18	26% normal working mode 18% failure working mode	Possibility of failure
$t = t_2+3$	5	52	0.05	0.43	5% normal working mode 43% failure working mode	Failure
$t = t_2+4$	6	57	0.005	0.74	0.5% normal working mode 74% failure working mode	Failure
$t = t_2+5$	7	62	0.0004	0.95	0.04% normal working mode 95% failure working mode	Failure

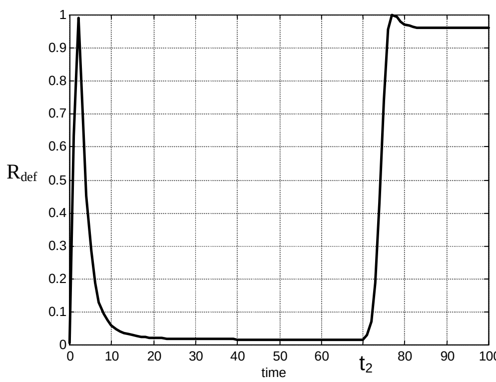

Fig. 10. Neuron response corresponding to the failure mode. Earlier detection of the failure.

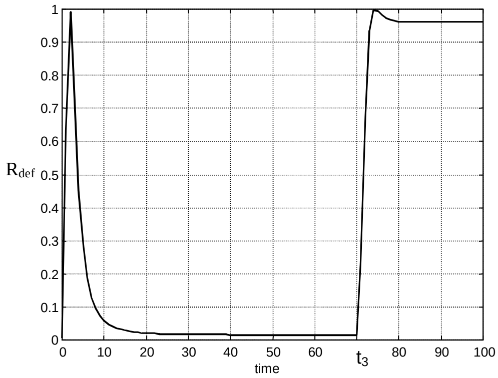

Fig. 11. Response of the neuron corresponding to the failure mode. Breakdown detection.

Case of a sudden and persistent breakdown. The RRBF neural network quickly detects this type of failure. The neuron output corresponding to the normal working mode decreases quickly and the output corresponding to the failure mode grows in the same way (Fig. 11).

Table 4 Cases of an abrupt and persistent breakdown

	S(t)	X(t)	R_{bf} (t)	R_{def}(t)	Working mode	Result
$t<t_3$	1	35	1	0	100% normal working mode	OK
$t = t_3$	7	47	0.19	0.23	19% normal working mode 23 % failure working mode	Possibility of failure
$t = t_3+1$	7	56	0.009	0.67	0.9% normal working mode 67% failure working mode	Failure
$t = t_3+2$	7	62	0.0006	0.93	0.06% normal working mode 93% failure working mode	Failure
$t = t_3+3$	7	65	0.0001	0.99	0.01% normal working mode 99% failure working mode	Failure

5. CONCLUSION

A neural network production system monitoring does not require the existence of a formal model of the system. Otherwise, the integration of dynamic evolution of parameters in the RRBF plays a very important role for the earlier detection of failures. This dynamic behavior is integrated using a self-connection of the input neurons. The application example shows that the RRBF network is able to dissociate between a true degradation and a false alarm. The training algorithm – RCE – is more flexible than the one used in the MLP models. The neural monitoring model presented in this paper can be used to learn on-line new operating modes and this without the problem of over-training. A practical hardware implementation open future prospects for on-line applications (real time).

6. REFERENCES

Basseville, M. and M.O. Cordier (1996). Surveillance et diagnostic des systèmes dynamiques: approche complémentaire du traitement du sugnal et de l'intelligence artificielle, research rapport. N°2861, INRIA/France.

Bernauer, E. and H. Demmou (1993). Temporal sequence learning with neural networks for process fault detection. *IEEE International Conference on Systems, Man, and Cybernetics*, IEEE-SMC 93, **Volume 2**, pp. 375-380. Le Touquet, France.

Bernauer, E. and H. Demmou (1995). Les méthodes de détection, localisation et diagnostic. Research rapport N°95473, LAAS /Toulouse, France.

Bernauer, E. (1996). *Les réseaux de neurones et l'aide au diagnostic: un modèle de neurones bouclés pour l'apprentissage de séquences temporelles*, PhD thesis, LAAS/Toulouse, France.

Broomhead, D.S. and D.Lowe (1988). Multivariable Functional Interpolation and Adaptive Networks, *Complex Systems*, **Volume 2**, pp. 321-355.

Chappelier, J.C. (1996). *RST: une architecture connexionniste pour la prise en compte de relations spatiales et temporelles*, PhD Thesis, ENST/PARIS, France.

Elman, J.L. (1990). Finding structure in time, *Cognitive Science*, **Volume 14**, pages 179-211.

Jodouin, J.F. (1994). *Les réseaux neuromimétiques*, informatique collection, Hermes, France.

Reilly, D.L., L.N.Cooper and C.Elbaum (1982). A Neural Model for Category Learning, in *Biol. Cybernetic*, **Volume 45**, p.35-41.

Zemouri M.R., D.Racoceanu and N.Zerhouni (2001). The RRBF - Dynamic Representation of time in Radial Basis Function Network, in *the 8th IEEE International Conference on Emerging Technologies and Factory Automation*, ETFA' 2001.