Design of an Embedded System for the Proactive Maintenance of Electrical Valves

Luiz F. Gonçalves[1], Jefferson L. Bosa[2], Renato V. B. Henriques[1], Marcelo S. Lubaszewski[1]
[1]Departamento de Engenharia Elétrica - PPGEE
[2]Instituto de Informática - PPGC
Universidade Federal do Rio Grande do Sul
Av. Osvaldo Aranha, 103, Centro, Porto Alegre, RS
CEP: 90035-190, Telephone +55 51 3308-3129
[1]{luizfg, rventura, luba}@ece.ufrgs.br, [2]jlbosa@inf.ufrgs.br

ABSTRACT
This paper presents a proactive maintenance scheme for the detection and diagnosis of faults in electrical valves. In our case study, these actuators are used for controlling the flow in an oil distribution network. An embedded system implements self-organizing maps for the detection and classification of faults that lead to deviations either on torque, or on the valve opening position. For fault detection, the map is trained using a mathematical model devised for the electrical valve. For fault classification, training is performed by fault injection based on parameter deviations over this same mathematical model. In both cases, the maps store the energies of the torque and the opening position that are computed using the wavelet packet transform. Once the maps are trained, the embedded system is ready for on-line monitoring the actuator. During the on-line testing phase, the embedded system computes the best matching between an acquired input vector (current torque and position energies) and the synaptic weight vector of the trained map. This matching is quantified by computing the Euclidean distance between these vectors and guide the fault detection and classification steps. The complete scheme was prototyped using FPGAs. The results obtained for area, performance and memory requirements point out to a low cost, promising solution for embedding maintenance in electrical actuators.

Categories and Subject Descriptors
B.6.3 [**Reliability, Testing and Fault-Tolerance**]

General Terms
Design and Reliability.

Keywords
Testability Issues, Embedded Systems, Electromechanical Systems, Novel Applications of FPGAs.

SBCCI'09, August 31st - September 3rd, 2009, Natal, RN, Brazil.

1. INTRODUCTION
Recent advances in electronics and computing have made it attractive to automate and integrate proactive (also known as intelligent) maintenance tasks into embedded systems [1, 7, 11]. Differently from the traditional maintenance (corrective, preventive or predictive) that is based either on post-failure correction or on off-line periodic system checking, the proactive maintenance focuses on fault prediction and diagnosis based on component lifetimes and on system on-line monitoring [13].

By using proactive maintenance it becomes possible to keep track of the equipment or the whole industrial plant, to quantify the performance degradation of the parts and thus to determine the remaining system lifetime. As a consequence, it turns feasible to schedule the replacement of degraded parts for idle or lower production periods of time, or automatically reconfigure the system for continuous (although degraded) operation till the faulty parts can be repaired.

These systems require wide instrumentation for on-line monitoring, and distributed intelligence to accurately locate the parts that start deteriorating. Both features can be achieved by communicating intelligent vibration, temperature, torque, position and other sensors through an industrial bus, for example.

Within this context, some strategies for predictive and proactive maintenance have been proposed in the literature that are based on extensive signal processing, on statistical analysis, and often on artificial intelligence methods as well [4, 6, 15].

In this work, a proactive maintenance scheme for the detection and diagnosis of faults in electrical valves is proposed.

In our case study, these actuators are used for controlling the flow in an oil distribution network. An embedded system implements self-organizing maps [8] for the detection and classification of faults in the valve. The complete scheme, prototyped using FPGAs, enhances the overall system reliability by making it possible to quantify the actuator degradation along time, to anticipate the fault occurrence and diagnose the fault location.

On one hand, some few works have already found in self-organizing maps a tool to solve maintenance problems [6, 14]. For instance, [6] addresses the challenging issue of predicting the remaining useful life for ball bearing prognostics. On the other hand, few additional works [2, 12, 17] have proposed different hardware architectures for the implementation of self-organizing maps, but all targeting other applications than the system maintenance.

To the best of our knowledge, this is the first work to apply a customized hardware implementation of self-organizing maps to solve the maintenance problem. In addition, an embedded system solution is proposed for the proactive maintenance of a sort of application that has only known corrective and preventive practices so far.

The paper is organized as follows. In section 2 the proactive maintenance system is proposed. A mathematical model is devised for a particular valve and, considering this model, self-organizing maps are proposed for fault detection and fault classification. Fault injection is used for training the maps. Section 3 presents experimental results, including fault detection and fault diagnosis experiments that validate the proposed maintenance method. Section 4 discusses the design of an embedded system to implement the maintenance system and evaluates the memory requirements, area and performance of an FPGA-based prototype. Concluding remarks are given in section 5.

2. PROACTIVE MAINTENANCE OF ELECTRICAL ACTUATORS

A maintenance scheme can only claim to be proactive if it is at least built over some instrumentation, appropriate signal processing and on-line monitoring features for fault detection and fault classification. Further proactiveness can be achieved by predicting the time to failure, by reconfiguring the industrial plant for continuous operation, etc.

Although the concepts exploited in this paper are quite general, the proactive maintenance system we are proposing here applies to a particular class of electrical valves. These valves have embedded sensors for torque and opening position measurement. Detectable faults in the valve are those that lead to measurable deviations either on torque, or on the opening position energies computed using the wavelet packet transform (WPT). As a reference, the normal, degraded and faulty behaviors of the valve are stored in self-organizing maps (SOM) that are on-line checked for the detection and location of faults that may occur during the valve operation. These maps are built during a training phase using a mathematical model (MM) for the valve, and are consulted on-line by computing the best matching between an acquired measure and the neurons of the trained reference map.

Figure 1 shows the proposed system. The system is composed of three main blocks: a MM that represents the actuator, the valve and the pipe behavior; the signal processing and characteristics extraction tool based on the WPT; and the artificial intelligence tool based on SOM, whose maps are computed in a PC station and are later on stored in the on-line monitoring system.

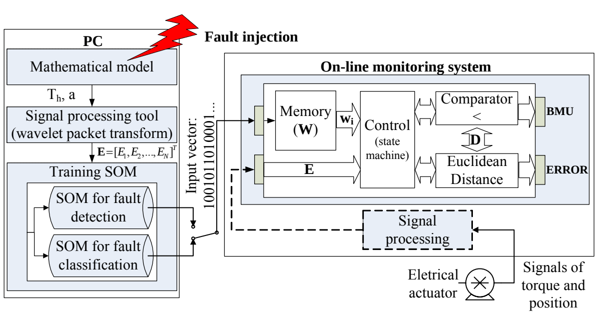

Figure 1. The proactive maintenance scheme.

2.1 Mathematical Model

Valves are devices used for flow control. They may be of different types (drawer, sphere or globe valves, for example) and each type fits better different kinds of applications. In our case study, an electrically actuated drawer valve is used for the control of the oil flow in a distribution pipe. In Figure 2 the electrical actuator, the valve itself and the fluid pipe are shown.

The valve is modeled in order to evaluate its behavior in different functioning conditions, including normal, degraded and faulty operation. As indicated in Figure 2, this model takes into account a set of forces to analyze the gate valve opening and closing movements, when the actuating force is transmitted from the electrical engine (asynchronous machine) through the valve gears.

Differential and algebraic equations are devised and included in a system of non-linear equations to model the actuator behavior and the various mechanical forces involved. Several physical constraints are considered in the modeling process in order to simplify the representation and, consequently, the computational effort required [9, 3]. A third-order model is chosen for the induction engine, because it can represent the permanent regime and transient conditions. This model is given by the differential equations (1) to (3), where V_d' and V_q' are internal voltages, and s the slip of the asynchronous engine. The differential equations (4) and (5) complete the overall fifth-order model and relate the gear system and the pipe through the gate position (a), the speed (v_a) and the valve gate acceleration (a_a).

Finally, the algebraic equations that describe the electrical torque of the asynchronous engine (T_e), the voltages of the asynchronous machine (V_{ds} e V_{qs}), the flow (F_f), friction (F_a) and spring force (F_m), as well as the gate shutter force (F_h) and gate shutter torque (T_h), are given by equations (6) to (12) for θ = 90º:

$$\dot{s} = \frac{1}{2H}\left(T_e - T_m\right) \tag{1}$$

$$\dot{V}_d' = \frac{-1}{T_0}\left[V_d' - I_{qs}(X_s - X_s')\right] + s\omega_s V_q' \tag{2}$$

$$\dot{V}_q' = \frac{-1}{T_0}\left[V_q' + I_{ds}(X_s - X_s')\right] - s\omega_s V_d' \tag{3}$$

$$\dot{a} = v_a \tag{4}$$

$$\ddot{a} = a_a = \frac{1}{M_h}\left(F_h - F_f - F_a - F_m\right) \tag{5}$$

$$V_{ds} = V_d' - R_s I_{ds} + X_s' I_{qs} \tag{6}$$

$$V_{qs} = V_q' - R_s I_{qs} - X_s' I_{ds} \tag{7}$$

$$T_e = V_d' I_{ds} + V'_q I_{qs} \tag{8}$$

$$F_h = \frac{T_h}{R_h \cos\theta} = \frac{-T_m K_R T_{mb}}{R_h \cos\theta} \tag{9}$$

$$F_m = K_M a \tag{10}$$

$$F_a = C_a v_a \tag{11}$$

$$F_f = \frac{V_f^{\,2} A_v}{\rho N_R^2 (100 - a)^2 C_v^{\,2}} \tag{12}$$

The solution for the system of equations (1) to (12) is obtained in this work using the Newton-Raphson method. This model can be used as is for the evaluation of the normal behavior of the valve. By deviating the nominal values of the parameters in the equations, fault injection can be performed in a simple manner and misbehaviors (degraded or faulty valve) may be observed in the torque and opening position curves. This mathematical model is implemented in MatLab and runs in a PC station.

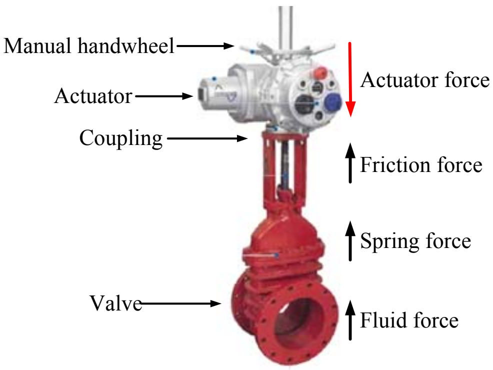

Figure 2. Electrical actuator, valve and pipe.

2.2 Signal Processing

Torque and position signals are generated by the mathematical model for normal, degraded and faulty operations. The wavelet packet transform (WPT) is the tool used in this work for processing these two classes of signals.

Since the WPT preserves timing and spectral information, it is a suitable tool for the analysis of non-stationary signals such as gears, bearings and actuators impulsive signals. Additionally, the capability of decomposing the signal in frequency bands makes the WPT more attractive here than other signal processing tools such as the Fourier and the wavelet transforms [13, 10].

Information on energy (**E**) is extracted from the valve torque and opening position WPT. The spectral density is divided into N frequency bands, and the resulting information is used by the artificial intelligence tool for the construction of the self-organizing maps described next.

Similarly to the MatLab mathematical model, the WPT tool runs in a PC station during the system training phase. During on-line testing, the WPT shall be part of the internal resources of the embedded system.

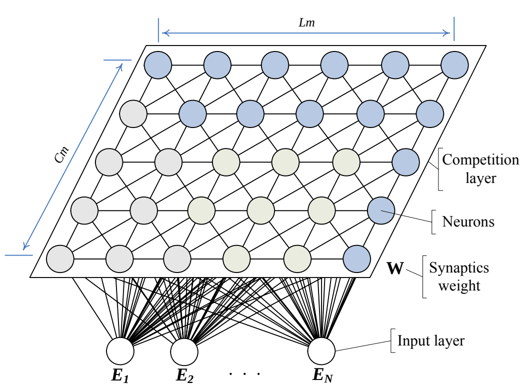

Figure 3. Basic SOM structure.

2.3 Self-Organizing Maps

The self-organizing maps, or Kohonen maps, belong to a class of neural networks that apply the unsupervised learning paradigm based on competition, cooperation and adaptation techniques [8]. Figure 3 shows the basic SOM structure.

Formally speaking, the ultimate goal of a Self-organizing Map is, after being trained, mapping any input data from a $\mathbf{R}^n$ space representation into a two-dimensional, lattice-like matrix. This lattice builds over a competition layer, with J neurons on the net, where $J = C_m . L_m$ in Figure 3.

For SOM training, the synaptic weights $\mathbf{W} = [W_1, W_2, \ldots, W_N]^T$, of all network neurons are initially assigned random values. Next, known input vectors representing the system behavior are mapped into the matrix. The input vectors, in our case, are energy values obtained from the WPT and defined as $\mathbf{E} = [E_1, E_2, \ldots, E_N]^T$. The most important step of the training phase is the competition. Using Equation (13), the competition step searches in the current map the neuron with the synaptic weights vector **W** that best matches the input vector **E**, so minimizing the Euclidean distance D_{kj}:

$$D_{kj} = \left\| \mathbf{E}^k - \mathbf{W}^j \right\| = \sqrt{\sum_{i=1}^{N} (E_i^k - W_i^j)^2} \qquad (13)$$

where: $k = 1, 2, \ldots, K$; $j = 1, 2, \ldots, J$; K is the number of input vectors. The neuron which presents the smallest value for D_{kj} (min D_{kj} is defined as the quantization error in the testing phase) is defined as the winner neuron, $\mathbf{W}_{BMU}$ [6, 8].

Two additional steps build the training phase: the cooperation and the adaptation. The cooperation step is in charge of identifying the direct neighbors of the winner neuron in the map. The adaptation step is in charge of updating the synaptic weights **W** of the direct neighbors as a function of the input vector **E**. Further details on these two steps can be found in [8].

According to Figure 1, two maps are trained in our case study, one for fault detection and another for fault classification. For fault detection purposes, the map is trained considering only typical system situations of normal, fault-free valve operation. For fault classification, the map is trained considering normal, degraded and faulty situations, whose misbehaviors are simulated by injecting parameter deviations into the valve mathematical model.

For SOM in-use mapping (on-line testing), only the competition step is performed considering the measured input vector E and the synaptic weights W of the neurons in the trained map. For fault detection, the quantization error min D_{kj} is computed and compared to a detection threshold. If the error exceeds this threshold, a fault is detected.

For fault classification, the map is colored such that the distance between neighboring neurons can be clearly visualized. The distance is given by the difference between the synaptic weights of neighboring neurons. Closer neurons will appear clustered in the map and will be assigned the same color. Different colors will denote neurons under different operation conditions: normal, degraded or faulty. Once the winner neuron is computed for a particular input vector E, the current system status can be identified in the colored map and, in case of faulty behavior; the fault shall be easily located.

3. EXPERIMENTAL RESULTS: FAULT DETECTION AND FAULT DIAGNOSIS

To train the fault detection map, a lot of simulations are performed to obtain typical values of torque and opening position under normal valve operation. For the torque case, for example, the maximum fault-free value considered equals 250 N.m.

To train the fault classification map, in addition to fault-free, fault simulation is needed. To exemplify, K_R and K_M parameters are gradually incremented in the ranges shown in Table 1, such that degraded and faulty valve behaviors can be observed.

K_R deviations (Equation 9) simulate the degradation of the internal valve worm gear, till it breaks. K_M deviations (Equation 10) simulate the elasticity loss of the valve spring along time.

Table 1. Range and variation rate for K_R and K_M.

Parameter	Range	Rate
K_R	11.000 - 12.000	0.010
K_M	4.215 - 5.215	0.010

As a consequence, for these faults the overtorque may reach 275 N.m and the valve may be prevented to open the gate beyond 80% of the full range (Figure 4).

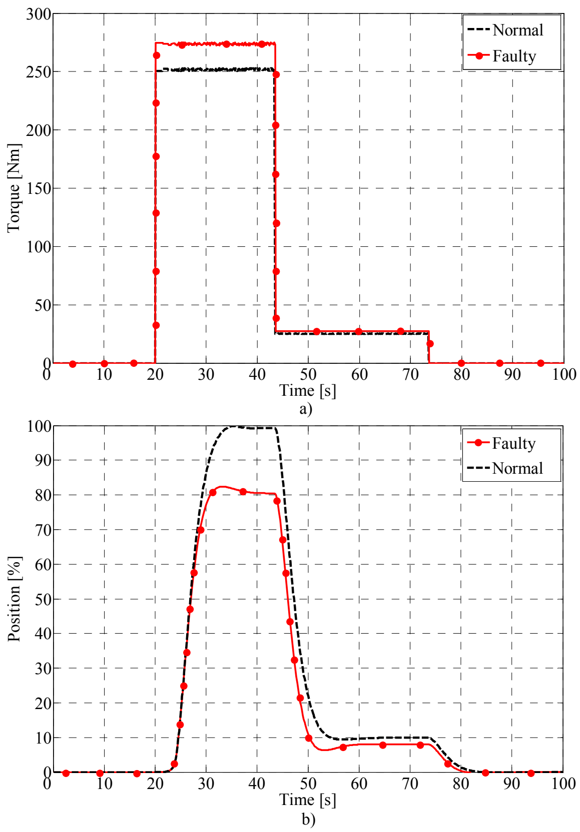

Figure 4. Fault simulation: a) torque and b) opening position.

As mentioned in Section 2.3, after the fault detection map is trained, the quantization error (min D_{kj}) is computed during the on-line testing phase.

Figure 5 shows the quantization error for an increasing degradation of K_R and K_M between the operation cycles 100 to 200. In cycle 200, the detection threshold is reached and a fault alarm can be triggered.

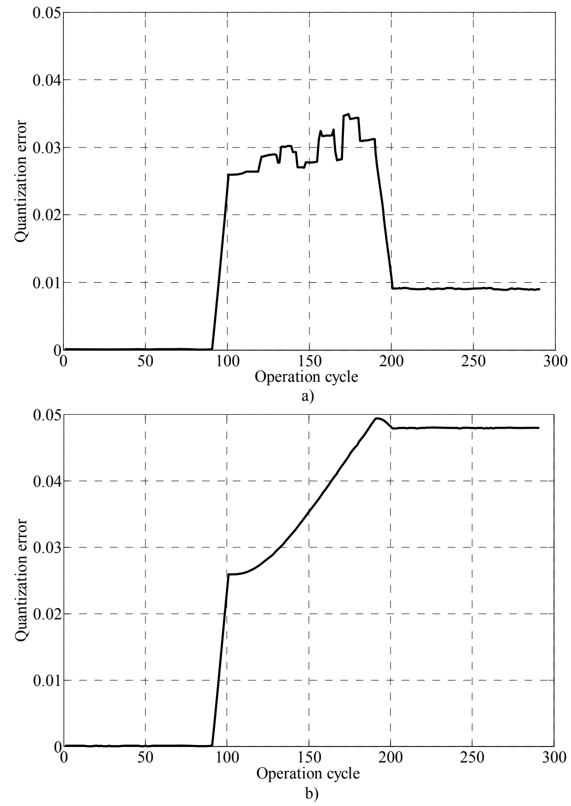

Figure 5. Quantization errors for: a) degrading K_R and b) degrading K_M.

Training for the classification of faults in K_R and K_M results in the map given in Figure 6. In the Figure the neurons appear clustered around Normal, Degraded K_R, Faulty K_R, Degraded K_M and Faulty K_M operation conditions. Each cluster is assigned a different color.

During the on-line testing phase, a winner neuron computed for a measured input vector can be easily located in this map and, consequently, the current status of the system is straightforward determined.

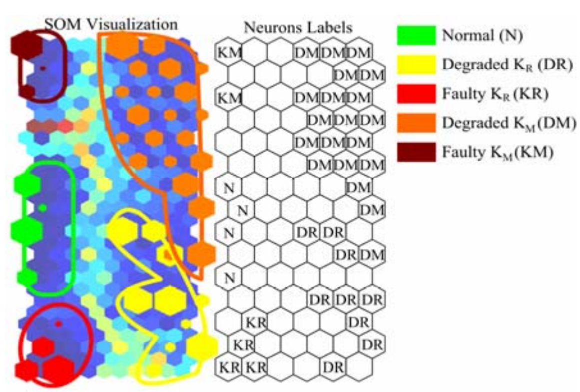

Figure 6. Fault classification map considering faults in K_R and K_M.

4. EMBEDDED SYSTEM DESIGN

According to the maintenance scheme in Figure 1, the fault simulation and map training steps can run off-line in a PC station. However, for the on-line monitoring phase, in addition to the acquisition, conditioning and A-to-D conversion of torque and position signals, the computation of the input vector WPT, of the winner neuron W_{BMU}, and of the quantization error min D_{kj}, shall be embedded into the valve.

Our proposal is to implement these functions using an FPGA platform, such as the XUP Virtex2 PRO Xilinx FPGA Development Board [16]. This board embeds various hardware resources, including A-to-D and D-to-A converters, a Virtex2 device, a 512MBytes SRAM, an OPB bus, several I/O ports and user interfaces. Based on this platform, we have chosen to reuse an available software implementation for the WPT computation [5], and decided to implement the W_{BMU} and min D_{kj} computation blocks in hardware, as an IP-core.

The WPT computation runs in a Microblaze processor synthesized for the FPGA device. The processor runs at a 100MHz clock. The Petalinux operational system and the WPT code are stored in the SRAM and occupy around 2.2 MBytes of memory.

Figure 7 shows in (a) the IP-core architecture proposed for the computation of W_{BMU} and *min* D_{kj}, and in (b) details of the hardware needed to obtain D_{kj}, the Euclidean distance. Two important points are to highlight in this architecture. First is that, for the SOM implementation, the most important and most area consuming block in the FPGA is the memory (BRAM). Second is that, for the accurate computation, the operators of the Euclidean distance circuitry shall be based on a floating point representation and the Std. IEEE 754 is used for this purpose. The circuits in Figure 7 were described in VHDL and have been synthesized and implemented in the Virtex2 PRO FPGA.

Experiments performed with this system have shown that map sizes of 90 neurons, with synaptic weight vectors of 20 elements of 32 bits each, are quite enough to reach the accuracy required for the valve application. Other applications may require much greater maps to distinguish between faults with close behavior, for example. In this case, the memory requirements may become critical for the implementation of the embedded system. The FPGA device in-use can implement up to 512 neurons, with weight vectors of 32 elements of 32 bits each.

In this work, we have implemented the embedded maintenance system using a powerful board. However, it is likely that in real life the already existing board for the actuator control is re-designed to accommodate an FPGA device implementing the maintenance functionality. In this case, the question that arises is whether the area still available in the FPGA will be enough to implement those functions that are currently implemented by other board devices.

Table 2 gives an answer to this question. As it can be noticed, keeping only the IP-core into the FPGA, there is definitely room enough to integrate the WPT in hardware.

Another important requirement is the embedded system performance, measured here by the computation time. The gate valve takes 100s to travel the full range, from the closed to the 100% open position and vice-versa. The whole travel is performed in 20 incremental steps, lasting 5s each.

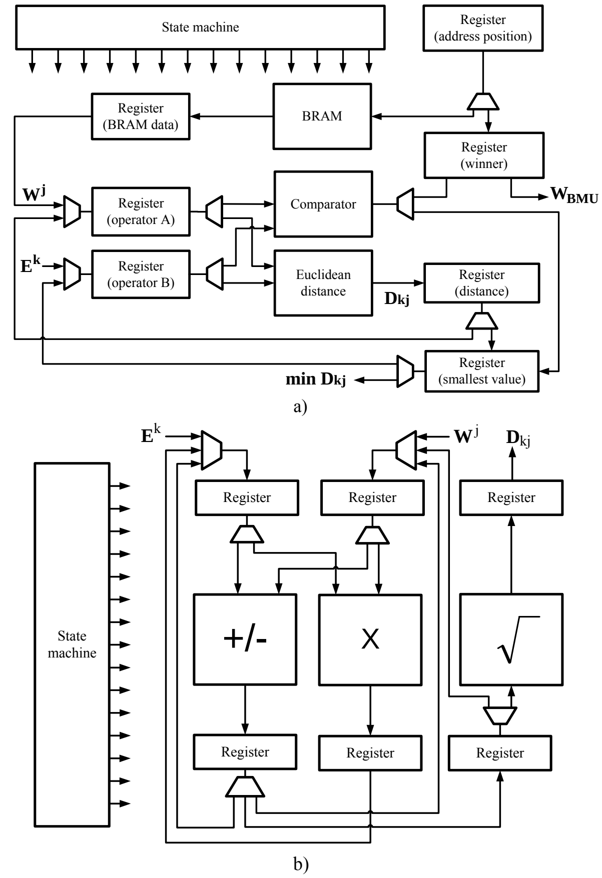

Figure 7. Proposed IP-core: a) SOM datapath and b) the Euclidean distance computation.

Therefore, in this application, t=5s is the time limit for all computations. Table 3 shows that the embedded system as is already meets the time requirements for the application. A much better performance is expected in the case the WPT is implemented in hardware in the same FPGA device where the SOM-IP is programmed.

Table 2. FPGA area evaluation.

Parameter	FPGA device	Embedded System		SOM - IP	
	Available	Used	Utilization	Used	Utilization
Slices	13696	5724	41,80%	2158	15,75%
Flip-flops	27392	5762	21,04%	2597	9,48%
LUTs	27392	6366	23,25%	2664	9,72%
BRAMs	136	42	30,90%	29	21,32%
IOBs	556	134	24,00%	-	-

Finally, it should be mentioned that the power consumption of the embedded system, either implemented as a board or as a single FPGA device, is not a critical issue in the valve application. This is because the valve engine consumes much more power to move the valve gate than the embedded electronics altogether does.

Table 3. W_{BMU} and min D_{kj} computation time for a single (T_h, a) measurement.

Algorithm	Time
WPT in software	0.3812s
SOM in hardware	0.0245s
Total	0.4057s

5. CONCLUSIONS

In this work, a proactive maintenance scheme is proposed for the detection and diagnosis of faults in electrical valves. These valves are used for flow control in an oil distribution network. To the best of our knowledge, this is the first attempt to apply a proactive maintenance methodology to this sort of actuators that have only known corrective and preventive practices so far.

Another novelty brought in by this work is that a customized hardware implementation of self-organizing maps is proposed to solve the valve maintenance problem. An embedded system implements these maps for the detection and classification of faults that lead to deviations either on torque, or on the valve opening position.

For fault detection and classification, the self-organizing maps are trained using a mathematical model and a fault injection procedure devised for the actuator, valve and pipe. During the on-line monitoring phase, the embedded system computes the best matching between an acquired measure and the neurons of the trained maps. This matching guides the fault detection and classification steps that show up very effective.

The embedded system was prototyped using an XUP Virtex2 PRO Xilinx FPGA Development Board. The results obtained for memory requirements, area and performance point out to a low cost, promising solution for embedding maintenance in electrical actuators.

6. ACKNOWLEDGEMENTS

This research is supported by the CNPq Brazilian Research Agency under contract number 142027/2008-1, CAPES Brazilian Research Agency and Petrobras S.A.

7. REFERENCES

[1] Djurdjanovic, D., Lee, J., and Ni, J. 2003. Watchdog agent – an infotronics based prognostic approach for product performance degradation assessment and prediction. Advance. Engineering Informatics. 17, 5, 109-125.

[2] Fang, X., Thole, P., Goppert, J. and Rosenstiel, W. 1996. A hardware supported system for a special online application of self-organizing map. IEEE International Conference on Neural Networks, Washington, 2 (Jun. 1996), 956–961.

[3] Fitzgerald, A. E. 1990. Electric machinery, McGraw-Hill, New York.

[4] Grall, A., Dieulle, L., Bérenguer, C., and Roussignol, M. 2002. Continuous-time predictive-maintenance scheduling for a deteriorating system. IEEE Transactions on Reliability, IEEE, 51, 2 (Jun., 2002), 141-150.

[5] Guleryuz, O. G. 2009. Wavelet packets source codes. Polytechnic Institute of New York University. Available at http://eeweb.poly. edu/~onur/source.html (May, 2009).

[6] Huang, R., Xi, L., Li, X., Liu, C. R., Qiu, H., and Lee, J. 2005. Residual life predictions for ball bearings based on self-organizing map and back propagation neural network methods. Mechanical Systems and Signal Processing, Elsevier, 21, (Oct., 2005), 193–207.

[7] Jinhua, D. and Erland, O. 2002. Availability analysis through relations between failure rate and preventive maintenance under condition monitoring. Department of Information Design and Product Development, Mälardalen University, Sweden, 21.

[8] Kohonen, T. 2001. Self-organizing maps, Springer.

[9] Kundur, P. 1994. Power system stability and control, McGraw-Hill, New York.

[10] Lathi, B. P. 1998. Modern digital and analog communication systems, 3rd ed.

[11] Muller, A., Marquez, A. C. and Iung, B. 2008. On the concept of e-maintenance: review and current research. Reliability Engineering & System Safety, Seville, 93, 8 (Aug., 2008), 1165-1187.

[12] Pino, B., Pelayo, F. J., Ortega, J. and Prieto, A. 1999. Design and evaluation of a reconfigurable digital architecture for self-organizing maps. 7th International Conference on Microelectronics for Neural, Fuzzy and Bio-Inspired Systems, Granada, (Apr. 1999) 395–402.

[13] Qiu, H., Lee, J., Lin, J., and Yu, G. 2006. Wavelet filter - based weak signature detection method and its application on rolling element bearing prognostics. Journal of Sound and Vibration, 289, 4 (Feb., 2006), 1066–1090.

[14] Saxena, A. and Saad, A. 2007. Evolving an artificial neural network classifier for condition monitoring of rotating mechanical systems. Applied Soft Computing Journal, Atlanta, 7, 1 (Jan., 2007), 441-454.

[15] Singh, G. K. and Kazzaz, S. A. 2003. Induction machine drive condition monitoring and diagnostic research – a survey. Electric Power Systems Research, Elsevier, 64, 145–158.

[16] Xilinx Inc. 2005. Embedded system tools reference manual, Embedded Development Kit EDK 8.1i, Xilinx Inc., 5 (Oct., 2005).

[17] Yamakawa, T., Horio, K. and Hiratsuka, T. 2002. Advanced self-organizing maps using binary weight vector and its digital hardware design. 9th International Conference on Neural Information Processing, Singapore, 3 (Nov. 2002), 1330–1335.