

Intelligent Component Monitoring for Nuclear Power Plants

Final Technical Report

PI: Lefteri H Tsoukalas, PhD

School of Nuclear Engineering

Purdue University

West Lafayette, IN 47907

tsoukala@ecn.purdue.edu

July 2010

Work performed under DOE Grant Award Number DE-FG07-ID14766

1. Project Background

Research activities for the overall project and where needed special emphasis for the last year technical accomplishments are summarized in this report. The work has involved several undergraduate and grad nuclear engineering students including a doctoral student at Purdue University who is near completion of his studies (Mr. Miltos Alamaniotis).

Per proposed work the emphasis of these activities was on validation, verification and statistical treatment of the advanced agent-based intelligent component monitoring methodology. Publications describing the new approach developed and obtained results are given at the end of the report. The rest of the report summarizes main technical findings after a brief description of the background and past accomplishments.

Reliability and economy are two major concerns for a nuclear power generation system. Next generation nuclear power reactors are being developed to be more reliable and economic. An effective and efficient surveillance system can generously contribute toward this goal. Recent progress in computer systems and computational tools has made it necessary and possible to upgrade current surveillance/monitoring strategy for better performance. For example, intelligent computing techniques can be applied to develop algorithm that help people better understand the information collected from sensors and thus reduce human error to a new low level. Incidents incurred from human error in nuclear industry are not rare and have been proven costly.

The goal of this project is to develop and test an intelligent prognostics methodology for predicting aging effects impacting long-term performance of nuclear components and systems. The approach is particularly suitable for predicting the performance of nuclear reactor systems which have low failure probabilities (e.g., less than 10^{-6} year⁻¹). Such components and systems are often perceived as peripheral to the reactor and are left somewhat unattended. That is, even when inspected, if they are not perceived to be causing some immediate problem, they may not be paid due attention. Attention to such systems normally involves long term monitoring and possibly reasoning with multiple features and evidence, requirements that are not best suited for humans.

2. Accomplishments

The main goal at this stage is to implement an intelligent agent monitoring framework and develop test cases. There were three major research activities performed during this phase, (1) investigating intelligent and learning techniques used by agents; (2) development of architectures for intelligent monitoring; and (3) studying of test cases.

2.1 Distributed Intelligence

There are actually several definitions in software engineering about “agents.” One of the most commonly used is the one from Weiss which we adopted in this project: “An **agent** is a computer system that is situated in some environment, and that is capable of autonomous action in this environment in order to meet its design objectives.”

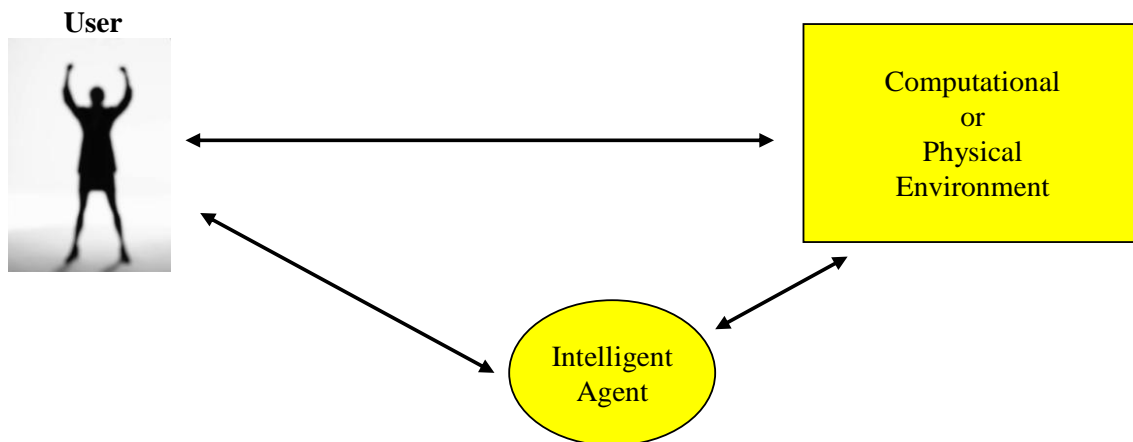


Figure 1: User – environment interface through Intelligent Agents

Properties of Agents:

Autonomy: Autonomy is defined as “agents are able to act without the intervention of humans or other systems: they have control both over their own internal state, and over their behavior.”

Social ability: The agents communicate with other agents and users via a common predefined language. This aims in their collaboration so as to achieve their goals. Specifically there is an interaction among sensors either to reach a common goal or their independent goals.

Reactiveness: The agents “feel” their environment and react in a short time period to changes occurring in that.

Pro-activeness: Agents are not only reacting to changes in their environment but are capable of suggesting actions that are based on their own goal. In other words they act on their own according to the current state of their environment. This property requires the sensor to be able to “think”.

Mobility: Agents are not staying to the same place but they can move to different locations inside a computational environment.

Adaptivity: Agents adapt continuously in their current environment or to the users requirements. This means that they possess the ability of learning.

Veracity: Agents never give false information or data intentionally.

Benevolence: Agents try to succeed in their tasks all the time.

Rationality: Agents act in order to reach their goals and never do something without purpose. Furthermore they do not operate against their aims.

Of significant importance for an agent is the environment in which it acts. Environments are classified based on different criteria. The types of environments are:

Accessible vs Inaccessible: The principle criterion for this classification is the availability of information whether is compete, accurate and updated.

Deterministic vs Non-Deterministic: This has to do whether an action causes always the same results or not.

Episodic vs Non-Episodic: Classification has to do whether or not the environment is separated in discreet and independent parts.

Dynamic vs Static: The criterion is whether the environment changes with the intervention of the agent or not.

Discrete vs Continuous: Classification has to with whether the agent possess a deterministic number of actions and data in his memory or not.

2.1.1 Intelligent Agents for Plant Processes

Intelligent agents used for our purpose are *deliberative agents*. This means that they possess an internal symbolic representation of their environment and a set of rules in order to plan their actions in that. In general they always have a snapshot of their state and their environment so as to design their actions.

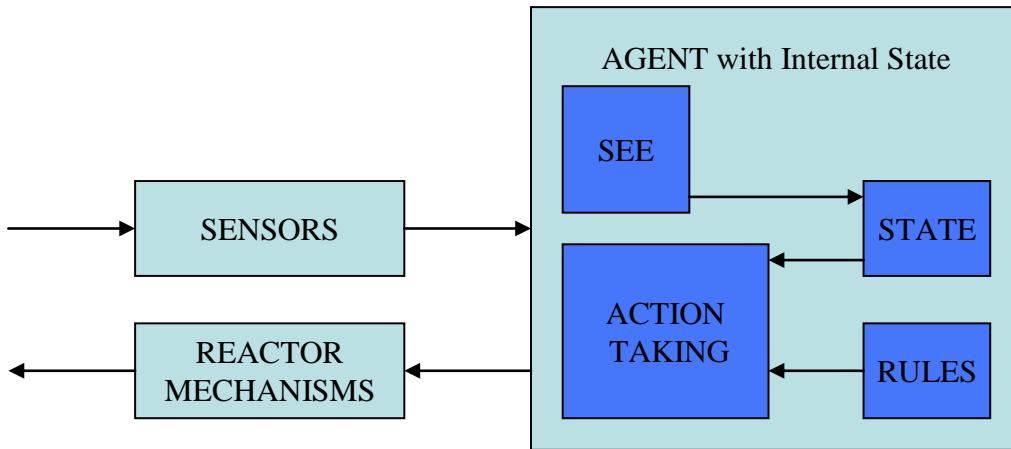


Figure 2: Scheme of deliberative agents

Usually deliberative agents have the following properties:

Beliefs: They represent the view and the opinion of the agent for its environment. It is possible that its view might be wrong.

Desires: Desires refer to the classification of the future states of the agents' environment. For example a future state might be desirable or not. At this level it is not checked whether the desirable state is reachable or not.

Goals: Goals are subset of the desires. They should be achievable and not be controversial.

Intentions: Intentions are a subset of the goals that the agent attempts to reach. It should be mentioned that is impossible to succeed all the goals and this results in a selection of a subset of them (which consists of the desires) according to some hierarchical criteria.

2.3 Communications Architectures in Multi-Agent Systems

There are two interface models that exist for the communication and collaboration of the agents: blackboard systems and message passing systems.

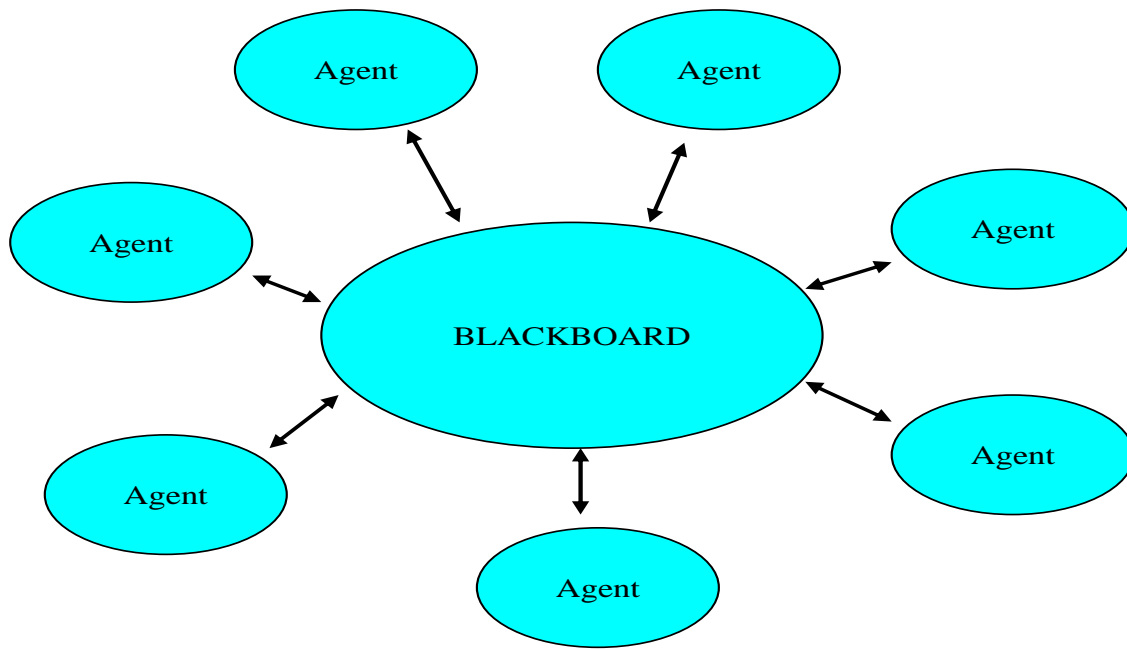


Figure 3: Blackboard Architecture

In the blackboard systems there is a common work space for all the agents of the system. Through that individual agents share their results or their tasks. Needless to say, whatever is stored in that common area is available and accessible from all system agents.

In the message passing systems the agents communicate through the exchange of messages. In order to have an efficient communication a common high level language is used.

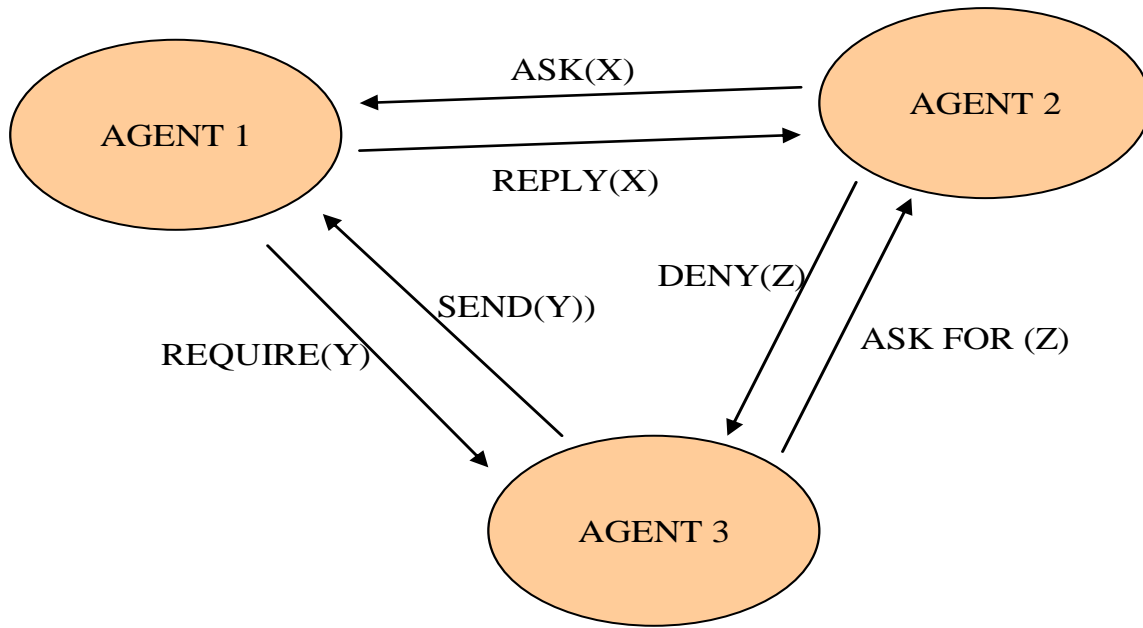


Figure 4: Scheme of message passing among agents

2.4 Purdue University Reactor (PUR-1)

The study of the Purdue University Reactor (PUR-1) conducted during the first quarter was aimed to understand the system structures and eventually be able to identify some key components for investigations. The latter task will be performed during the next stage.

Purdue University Reactor (PUR-1), operated under the direction of the School of Nuclear Engineering, is the first and only operating reactor in the state of Indiana. PUR-1 is a 1 kW swimming pool reactor that is primary dedicated to research and education.

The current instrumentation at PUR-1 consists of four channels. The electronics underwent a major upgrade during 1999 and 2001 to replace the vacuum tubes technique with more advanced solid state circuits.

2.5 Review of analytical tools

The outcome of the project heavily relies on early detection of the signal trend. For this purpose, two potential tools have been identified, wavelets and Hilbert-Huang Transform.

2.5.1 Wavelet tool

Wavelets analysis plays an important role in nonlinearity detection. A wavelet transform decomposes the demand signal into one approximation group and several detailed groups. The continuous and discrete wavelet transforms are defined respectively:

$$C(a,b) = \int_R \frac{1}{\sqrt{a}} \Psi\left(\frac{t-b}{a}\right) dt$$

$$C(j,k) = \sum_{n \in \mathbb{Z}} s(n) g_{j,k}(n)$$

Where $s(t)$ is the signal; Ψ is the mother wavelet; a,j and b,k are scaling and shifting factors respectively. The wavelet transform is a multi-resolution analysis technique, which is capable of locating localized features. The multi-resolution analysis capability of wavelet provides us lots of power for signal processing.

Wavelet transform is capable of characterizing local features. Wavelet coefficient is a function of translation factor, thus temporal information is keep among wavelet coefficients.

Wavelet transform is a multi-resolution (or multi-scale) analysis. Wavelet coefficient is also a function of scaling factor. From the viewpoint of filtering, wavelet functions are some band-pass filters with different center frequencies and bandwidths determined scaling factor. Thus wavelet transform is equivalent to filtering a signal using a series of band pass filters. The wavelet function corresponding to a small scaling factor a has narrow support in time domain but wide support in frequency domain. This wavelet function then has coarse resolution in frequency domain. On the contrary, a large scaling factor generates a wavelet function having wide support in time domain but narrow support in frequency domain. Such a wavelet function then has relatively high resolution in frequency domain. This property is illustrated in Figure 1. Time-frequency space of different wavelet functions is shown in Figure 1. By selecting different scales, a wavelet transform is analyzing the signal with different resolutions. This can be simulated by the behavior of human being. If he/she wants to gather an overall image of an object, he/she will use a coarse resolution with big view angel to observe. Suppose he/she would like to focus on some details, he/she then needs to move closer to the object to obtain a finer resolution.

The quality of wavelet functions is a constant. The quality of a band-pass filter is defined as value of its central frequency dived by it bandwidth. As clearly shown in Figure 5, as the central frequency increases, the bandwidth increases proportionally. Thus the quotient remains constant. This is an important property because it agrees with the characteristic of membrane in human's ear, which is essentially a constant-Q band-pass filter.

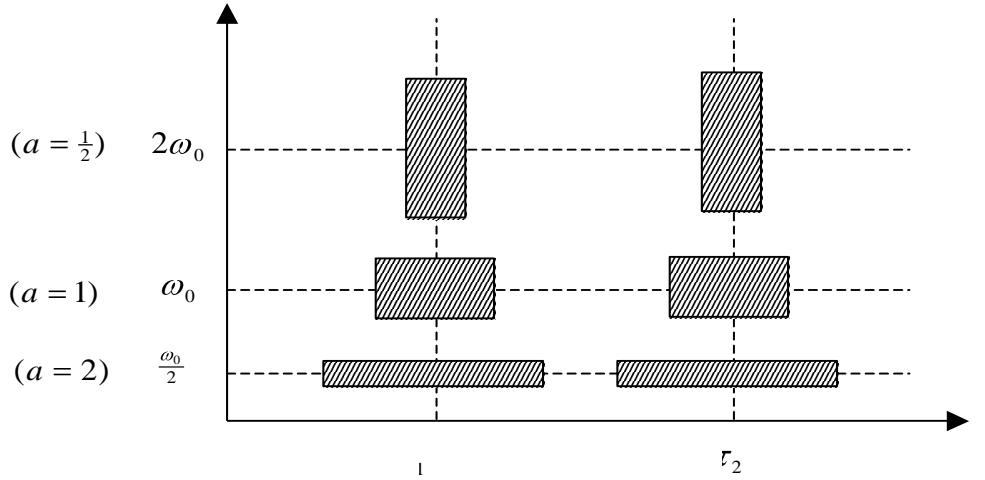


Figure 5: Time-frequency space of wavelet functions

One of the outstanding characteristics of wavelets is their capability of detecting singularities in a noisy signal. This can partially explained by its close relation to a mathematic term that is used to describe the degree of singularity: *Lipschitz Exponent*. If a signal $x(t)$ satisfies the following inequality around t_0 :

$$|x(t_0 + h) - P_n(t_0 + h)| \leq A|h|^\alpha$$

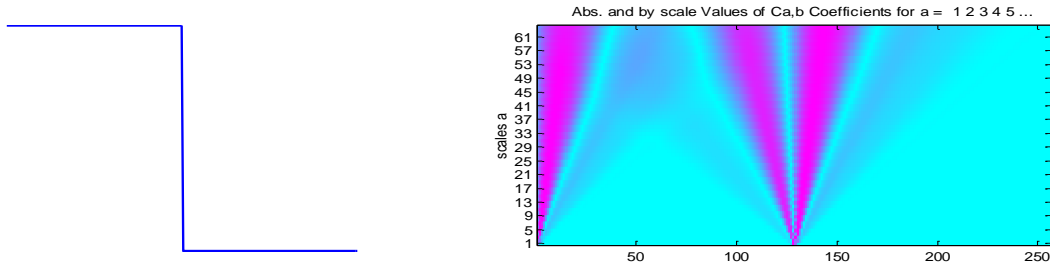
Where h is a sufficiently small number and P_n is a n -th order polynomial that passes t_0 , then the Lipschitz Exponent of $x(t)$ at t_0 is α .

A low Lipschitz Exponent means high irregularity. It is very nice that Lipschitz Exponent can be characterized with wavelet coefficients as shown in equation.

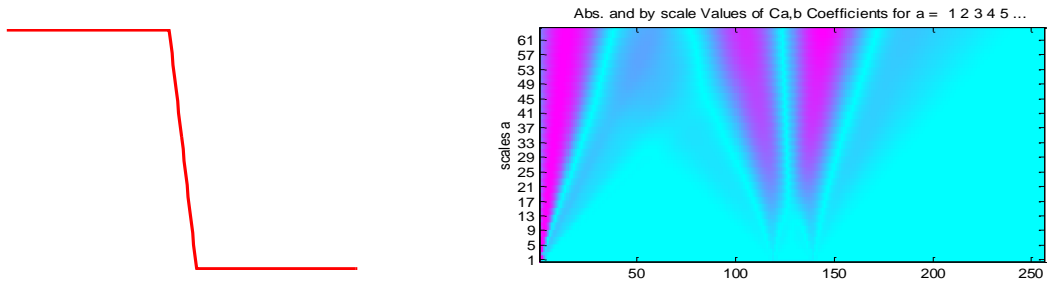
$$\log_2 |WT_{2^j} x(t)| \leq \log_2 K + j\alpha$$

Where K is a constant. The above inequality tells us the maximum value of wavelet coefficients at a particular scale is constrained by the product of scale factor and Lipschitz Exponent. To be more applicable, it is rephrased as the following: if the L. E. of a signal is less than 0 (rather irregular), the maximum value of its wavelet coefficients decreases while scale factor increases; if the L. E. is greater than zero (rather regular), opposite situation happens. This property can be used to detect different types of irregularity, as illustrated in figure 6. As shown in figure 6, the wavelet spectrum of a step function can be readily distinguished from that of a ramp function.

The abovementioned relations give us a practical way to estimate the degree of singularity using wavelet coefficients. The importance of singularity is that it can be used to locate the characteristic points of the signal, which can be subsequently used to identify the temporal or spatial relation of two signals.



(a) continuous wavelet decomposition of a step signal



(b) continuous wavelet decomposition of a ramp signal

Figure 4: Continuous wavelet decompositions of two signals (a) step (b) ramp

2.5.2 Hilbert-Huang Transfer (HHT)

The dynamics of a complex system such as a nuclear reactor are often characterized being both nonlinear and non-stationary. Unfortunately for long time we don't have effective tools to analyze processes that are both nonlinear and non-stationary. Conventional Fourier Transforms were developed for linear and stationary processes. Wavelet Analysis, an extension to the FT, can be used on non-stationary but linear processes. There are other tools available for nonlinear signals but cannot handle non-stationarity at the same time. The lack of analytical tools severely limits the capability of diagnostic systems to identify faults quickly and accurately.

The Hilbert-Huang Transform (HHT) recently emerges as a tool for handling both non-linear and non-stationary processes. With HHT, it is possible to characterize the dynamics of a nonlinear and non-stationary system in real time. For example, a conventional spectra analysis on a window of time only provides the “overall” behavior of the system during the selected window. However, the HHT is able to calculate the so call “instantaneous frequency” to characterize instant dynamics of the system at any time. It is hard to underestimate the value of such information for any diagnostic/prognostics systems.

2.6 Gaussian Process for Machine Learning as Prediction Tool

The Gaussian distribution is a well known probability distribution which describes one, or more, random variables. In case of one variable it is called *univariate*, whereas for more variables it is termed as *multivariate*.

In order to get a view of GP regression, the linear regression is examined initially. Thus, linear regression is formulated as:

$$y = w^T \phi(x) \quad (2.6.1)$$

where w is the vector of weights and $\phi(x)$ the vector of the basis functions. Both vectors are of length L . Next, it is assumed that weights have a prior distribution that is of Gaussian form with zero mean and variance $\sigma^2 I$. The equation below shows the prior distribution in a vector form.

$$p(w) = N(w | 0, \sigma^2 I) \quad (2.6.2)$$

where I is the identity matrix. At this point, to simplify the equations, the following notation is introduced:

$$y = \Phi w \quad (2.6.3)$$

where Φ is a matrix with elements $\Phi_{nm} = \phi_n(x_m)$. It should be mentioned that since y is a linear combination of Gaussian variables, has also a Gaussian distribution. Taking into consideration equation 2.6.2, the parameters are computed as shown in equations 2.6.4 and 2.6.5.

$$E[y] = E[\Phi w] = \Phi E[w] = 0 \quad (2.6.4)$$

$$Cov[y] = E[yy^T] = E[\Phi ww^T \Phi^T] = \Phi E[ww^T] \Phi^T = \sigma^2 \Phi \Phi^T \quad (2.6.5)$$

Of special interest is the expansion of the covariance term. Specifically, the last term of equation 2.6.5 can be denoted as K which is known as Gram matrix:

$$K = \sigma^2 \Phi \Phi^T \quad (2.6.6)$$

The elements of the Gram matrix can be expressed with the help of a kernel $k(x_n, x_m)$ as in equation 2.6.7:

$$K_{nm} = \sigma^2 \phi(x_n)^T \phi(x_m) = k(x_n, x_m) \quad (2.6.7)$$

Thus, in terms of Gaussian processes, the output of the linear regression y has a random distribution which is normal with mean zero and variance K :

$$p(y) = N(y | 0, K) \quad (2.6.8)$$

A careful observation of equation 2.6.8 yields the fact that the marginal distribution of y depends on the kernel function that composes the Gram matrix. That dependence provides us with a lot of flexibility since for different types of kernels, the result varies. Overall, the form of the covariance function is the one that determines the behavior of the GP regression.

In order to make prediction of the future, data of past denoting the system's behavior are necessary. In terms of learning systems, training sets are required for adjusting the model parameters. A training set is inserted into GP through the covariance term and more specifically through the selected kernel function.

$$(x_i, f_i) \text{ for } i=1, \dots, m \quad (2.6.9)$$

$$\Sigma p = \begin{bmatrix} K(X, X) & K(X, X_T) \\ K(X_T, X) & K(X_T, X_T) \end{bmatrix} \quad (2.6.10)$$

where X_T denotes the training data and X is the input data. Obtaining an output for a new incoming value requires the evaluation of the posterior distribution according to the so far observed data. This is simply done by conditioning equation 2.6.10 on the observed data. Accomplishing that a normal distribution is taken (equation 2.6.12) with mean and variance presented below.

$$P_{t|X_T, X} = N(M, V) \quad (2.6.12)$$

$$M = K(X_T, X) K(X, X)^{-1} P_T \quad (2.6.13)$$

$$V = K(X_T, X) - K(X_T, X) K(X, X)^{-1} K(X, X_T) \quad (2.6.14)$$

P_T expresses the joint distribution of the training data. Overall, GP regression yields a distribution around the prediction point for each input.

Gaussian Processes can be embedded in the processing module of intelligent agents and make predictions for a component's state. Updated predictions are sent by the agent to the central unit of the plant.

2.7 Neural Network as Diagnostic Tool

The NN has as many inputs as the sensors of the distributed system. If N is the number of sensors, then the inputs are also N . On the other side the number of perceptrons needed is $N-1$ (see Figure 7). Each perceptron has two inputs and one output. The inputs are the measured velocities from two sequential sensors and the output is either zero or one. Zero output is taken in case the sum of inputs is zero and one in other case. The activation function for each sensor is similar to that of a perceptron but not the same:

Activation function:

$$F(x) = 0, \quad \text{if } x=0 \quad (2.7.1)$$

$$F(x) = 1, \quad \text{if } x>0 \text{ or } x<0. \quad (2.7.2)$$

Each input is connected to two consequent perceptrons. The respective weights are 1 and -1. In other words, for each perceptron its first input is multiplied with 1 and the second with -1. Their sum goes through the activation function and the respective output is derived.

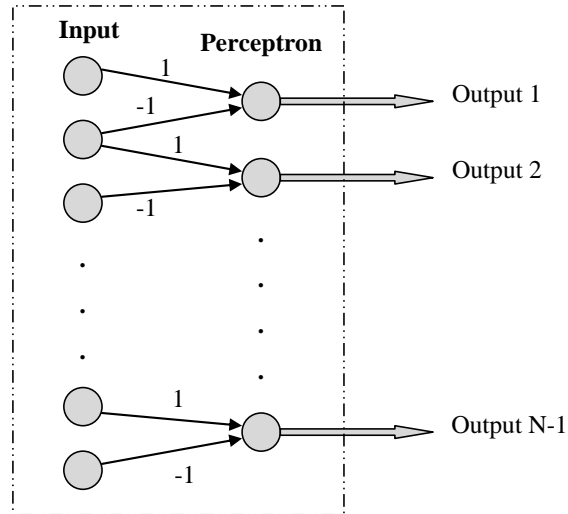


Figure 7: Proposed Neural Network structure for processing of data arriving from flow meter sensors.

The output of the perceptron depends on the sum of the weighted inputs. A zero sum denotes that the two measurements are equal, which means that the flowmeters obtain the same value. The latter implies that flow has not changed in the component between the two sensors. In case of a non-zero sum, the perceptron outputs the number one which indicates the change of flow between the two sensors. As a result, the neural network diagnosis a change in the flow inside the pipe. Additionally the system is able to locate and indicate the area in which the abnormality is detected.

3 Test Case Developments

3.1.1 Neural Network Diagnostic case

In this case the multi agent system is comprised of sensors which are of the same type. Specifically, acoustic sensors are adopted and deployed in order to check the circulation of the water flow in the nuclear plant.

Acoustic sensors are deployed across the area of the flow (pipes). In that case sensors consist of two parts: a transmitter and a receiver. The transmitter gives an ultrasonic signal and the receiver gets the train of the pulses.

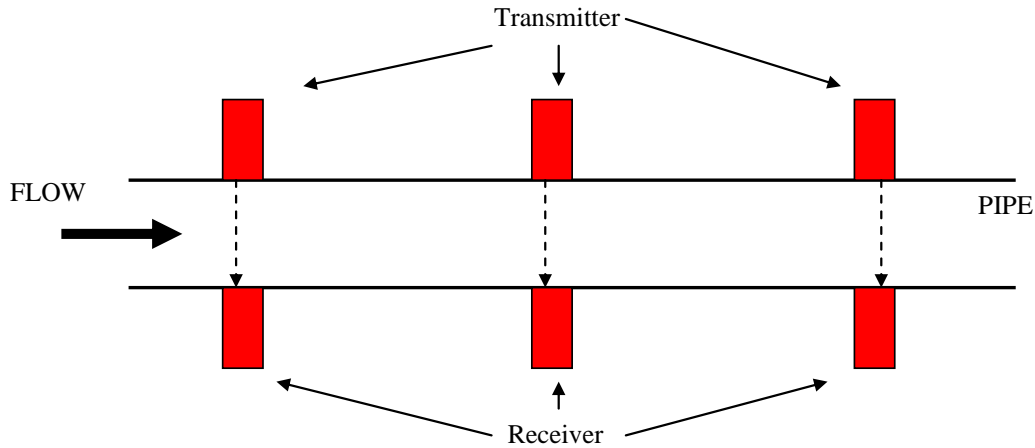


Figure 8: Ultrasonic measurements across a pipe

In this scenario the receivers get the respective ultrasonic signal and compute the ultrasonic velocity. If the material in the pipe is just water then we will get the ultrasonic value of water. In case there is water and steam (2-phase flow) then the received signal yields different value. This results in detection of the steam in the pipe. If we observe simultaneously the values coming from all sensors of the same pipe then we are able to check if the flow is uniform. In case some values are different than expected, this indicates the possible existence of a leakage (break).

Here what is under investigation is the case of flooding in a pipe: the injection is done between sensor one and sensor three. As a result the PNN captures the difference and yields the vector $[1, 0]$. The first element of the vector denotes the difference in measurements, something expected in our case. After a point, part of the water is being dragged upwards by the gas. Gradually, the descent flow of water stops and the water

moves up. The discrepancy is provided by the neural net which yields [1, 1]. At the final state of flooding the water is not driven by gravity any more. In that case the system measures a two phase flow above the injection point and a normal gas flow in the lower part.

3.1.1 Gaussian Processes embedded in agents for prognosis of blade degradation

Initially, a Gaussian process with its covariance function is trained. The obtained distribution after training ends is shown in figure 9. In this figure, we observe the mean distribution with its variance along the time scale. Although the variance includes negative values, these are not taken into consideration.

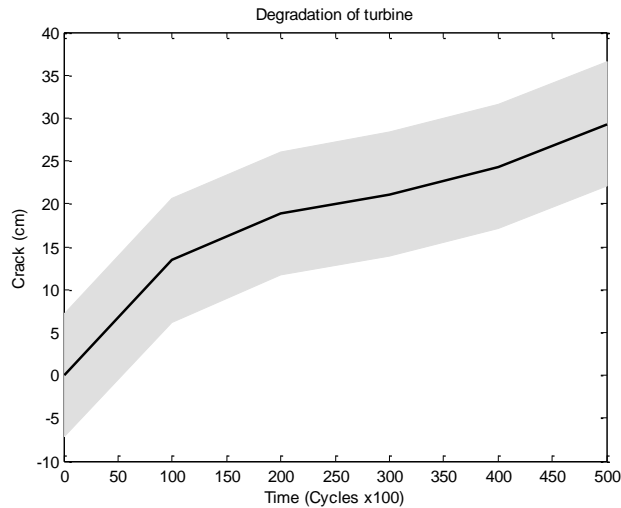


Figure 9. Mean distribution function and its variance obtained after training a Gaussian Process embedded in processing unit of an intelligent sensor.

It is observed from figure 5 that the prediction error of the updated distribution is less than the one given by the prior (trained) function. So, sequential update of prediction distribution decreases MSE. At this point it should be also mentioned that all measurements were in the range of the predicted variance for both cases (trained prior and posterior).

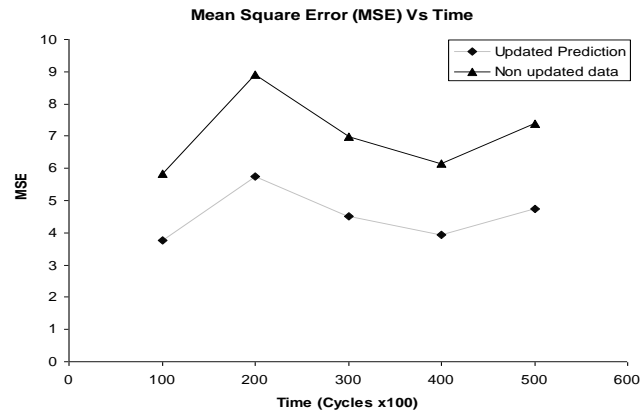


Figure 11. Mean Square Error for measurements between prior distribution and updated posterior.

4. Publications Related to the Project

In addition to the publications produced in the earlier years of the project and stated in previous annual technical reports, the following publications are thus far produced:

Alamaniotis, M., Gao, R. & Tsoukalas, L.H. 2009. Distributed Intelligence System for Online Action-Taking in Non Anticipated Situations in Nuclear Power Plants. In *Proceedings of ICAPS-09 Scheduling and Planning Applications Workshop, Thessaloniki, Greece, 18 September 2009*.

Alamaniotis, M., Youtsos, M., Gao, R. & Tsoukalas, L.H. 2009. Pseudo Neural Network based diagnostic system for two phase flow annular flow in nuclear power plants. In *Proceedings of the International Conference on Optimization using Exergy based Methods and Computational Fluid Dynamics, Berlin, Germany, 20-23 October 2009*.

Alamaniotis, M., Ikonomopoulos, A. & Tsoukalas, L.H. 2010. Gaussian Processes for failure prediction of slow degradation components in nuclear power plants. *To appear in proceedings of European Safety and Reliability Conference 2010, ESREL-2010, Rhodes, Sept 4-7, 2010*.

5. References

- Louis, A.K., Maass, P., Rieder, A., (1997). *WAVELETS: Theory and Applications*. New York: Wiley
- Fukunaga, K., (1997). *Statistical pattern recognition*. New York: Wiley
- Mallat, S. (1999). *A wavelet tour of signal processing*. San Diego: Elsevier
- Misiti, M., Misiti, Y., Oppenheim, G., Poggi, G.M., (2008). *Matlab Wavelet toolbox*. Natick: Mathworks Inc
- Tsoukalas, L., & Uhrig, R. (1997). *Fuzzy and neural approaches in engineering*. New York: Wiley
- Weiss, G. (2000). *Multiagent systems: an approach to distributed artificial intelligence*. Boston: MIT press Ltd
- W. Hines, D. Wrest and R. Uhrig, 'Plant Wide Sensor Calibration Monitoring' in *proceedings of 1996 IEEE International Symposium on Intelligent Control*, Dearborn, MI, pp. 378-383, 1996
- G. Jahn, S. McArthur, 'Staged Implementation of an Agent based advanced gas cooled reactor condition monitoring system', 2007 IEEE Power Engineering Society General Meeting, IEEE: Tampa, FL, pp: 427-431, 2007
- R. Schirru, C. Pereira, 'A real time artificially intelligent monitoring system for nuclear power plants operators support', *Real Time Systems*, vol. 27, Kluwer Academic Publishers, 2004
- P. Tipping, 'Lifetime and ageing management of nuclear power plants: a brief overview of some light water reactor component ageing degradation problems and ways of mitigation', *International Journal & Piping*, vol: 66, Amsterdam: Elsevier, pp: 17-25, 1996.
- Bishop, C.M. 2006. *Pattern Recognition and Machine Learning*. New York: Springer.
- MacKay, D.J.C. 1998. Introduction to Gaussian processes. In C. M. Bishop (ed.), *Neural Network and Machine Learning*: 133-166. New York: Springer.