

Soft Computing Applications in Equipment Maintenance and Service

Piero P. Bonissone and Kai Goebel

GE Corporate Research and Development
One Research Circle, Niskayuna, NY 12309, USA

{bonissone, goebelk}@crd.ge.com

Abstract

We present methods and tools from the Soft Computing domain, which are used within the diagnostics and prognostics framework to accommodate imprecision of real systems. Soft Computing (SC) is an association of computing methodologies that includes as its principal members fuzzy, neural, evolutionary, and probabilistic computing. These methodologies enable us to deal with imprecise, uncertain data and incomplete domain knowledge typically encountered in real-world applications. We outline the advantages and disadvantages of these methodologies and show how they can be combined to create synergistic hybrid SC systems. We conclude the paper with a description of three successful SC case study applications to equipment diagnostics: 1) *Prediction of Paper Web Breakage in Paper Mill Using Neural Nets and Induction Trees*; 2) *Method for automated tuning of a raw mix proportioning controller in cement plants*; 3) *Adaptive Classification for Gas Turbines Anomalies*.

1. Introduction

In the industrial world, we encounter a wide range of modeling problems that require the analysis of uncertain and imprecise information. Usually, an incomplete understanding of the problem domain further compounds this modeling problem. For example, to support the service of equipment, we need to generate models that can analyze the equipment data, interpreting their past behavior and predicting their future one. These problems pose a challenge to traditional modeling techniques and represent a great opportunity for the application of soft computing methodologies.

In an effort to yield higher margins, many manufacturing companies are shifting their operation to the service field. Therefore, diagnostics and prognostics play a significant role in this paradigm shift. A typical example of this service focus is the use

of long-term service agreement (LTSA) contracts with guaranteed uptime. These contracts strongly motivate the service provider to keep equipment in working order as opposed to performing a maintenance action once a failure has occurred. As a consequence, service should only be performed when necessary, ideally before a failure has occurred, while avoiding superfluous maintenance. This objective can be accomplished by using tools that measure the system state and indicate incipient failures. Such tools must have a high level of sophistication and must be able to incorporate monitoring, fault detection, decision making about possible preventive or corrective action, and execution monitoring. Because of the complexity of the task, AI and in particular Soft Computing have been leveraged in the implementation of these tools. Some application of Soft Computing techniques in support of service tasks, such as anomaly detection and identification, diagnostics, prognostics, estimation and control, have been reported in [1-3]. In this paper we will briefly describe the components of Soft Computing and illustrate some of their most successful application to equipment service.

2. Soft Computing

The term Soft Computing (SC), a sub-field of Artificial Intelligence, was originally coined by Zadeh as an association of computing methodologies that "...exploit the tolerance for imprecision, uncertainty, and partial truth to achieve tractability, robustness, low solution cost, and better rapport with reality"[4]. According to Zadeh, Soft Computing "includes as its principal members fuzzy logics (FL), neuro-computing (NC), evolutionary computing (EC) and probabilistic computing (PC)"[5]. It should be noted however that we have not reached a consensus yet as to the exact scope or definition of SC (for instance, see [6].)

A reason for Soft Computing's success is the synergy derived from its components. SC's main characteristics are its intrinsic capability to create hybrid systems that are based on a loose (or tight)

integration of these technologies. This integration provides complementary reasoning and searching methods that allow the combination of domain knowledge and empirical data to develop flexible computing tools and solve complex problems.

Typically, real-world problems are ill-defined systems, difficult to model and with large-scale solution spaces. In these cases, precise models are impractical, too expensive, or non-existent. Our solution must be generated by leveraging two kinds of resources: *problem domain knowledge* of the process/product that we want to improve/fix and *field data* that characterize the behavior of such process/product. The relevant available domain knowledge is typically a combination of first principles and experiential/empirical knowledge, and is usually incomplete and sometimes erroneous. The available data are typically a collection of input-output measurements, representing instances of the system's behavior, and are usually incomplete and noisy. These two kinds of resources determine the two main approaches found in Soft Computing: *knowledge-driven* reasoning systems (such as Probabilistic and Multivalued Systems) and *data-driven* search and optimization approaches (such as Neuro and Evolutionary Computing). This taxonomy, however, is soft in nature, given the existence of many hybrid systems that span across more than one field.

The interaction of knowledge and data is illustrated in the three case studies described in the next section. In addressing these real-world problems, sometimes we need to tune *knowledge-derived models*, while other times we need to control the search in *data-derived models*. In the former case, we first translate domain knowledge into an initial structure and parameters and then use global or local data search to tune the parameters. In the latter, we first use global or local search to derive the models (structure + parameters), we embed knowledge in operators to improve global search, and then we translate domain knowledge into a controller to manage the solution convergence and quality of the search algorithm. Extensive coverage of this topic can be found in [7-8] and [3].

3. Case Studies of SC Applications in Diagnostics and Maintenance

3.1 Prediction of Paper Web Breakage in Paper Mill Using Neural Nets and Induction Trees

Problem. Paper mills have enormous size of often times a hundred meters in length or more. Making

paper involves running miles of wet and dry webs over and between large cylinders that convert pulp into the final product. The speed of the web can reach 60 mph. Under those conditions, it is not surprising that the web breaks about once a day. Breakage results in a standstill of the machine of about ninety minutes, which in turn causes plant-wide revenue loss of several million dollars per year. It is therefore desirable to prevent breakage by predicting imminent breakage and being able to take preventive action.

Breakage occurs when the roll draw is larger than permissible. Draw is necessary to stretch the paper to acquire desired properties and the force exerted is ideally held between certain minimum and maximum values. These values are determined by the tendency of the web to stick to a roll and the maximum force, which depends on non-constant web properties (tear strength, tensile strength, and dry content) and external factors (e.g., temperature, lubricant, roughness, ash content, and retention). For maximum yield, it is desirable to run the machines at the upper operating point [2].

Solution. A number of approaches were considered, such as fuzzy causal modeling, principal component analysis, and learning vector quantization. After an initial down select, two approaches were chosen because of their complementary nature: 1) Neural nets with fuzzy accelerators; and 2) induction trees derived from decision trees. Both help in predicting some aspect of the web breakage. The former is more accurate and provides a finer estimate of the break tendency while the latter is more useful for diagnosis and for finding relevant features related to breakage. The two approaches were then combined to give different types of information about the system state (Fig. 1). The neural net output was used to indicate the possibility of a break in the short term using a continuous number. The output of the induction tree was interpreted as a description of the sensitivity relationships between the current break tendency and the current values of the input sensor readings.

Results. The neural net was trained via backpropagation algorithm using normalized input sensor data. Output was a quantized measure before breakage. This measure was designed such that the last ten readings before breakage attained a value of 0.9, the next last 20 readings a value of 0.5 and the 10 readings before that a value of 0.1.

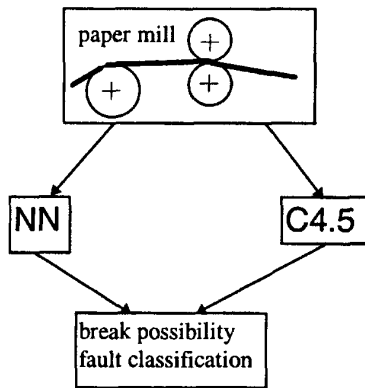


Fig. 1. Hybrid System for Breakage Detection

The interpretation could be little, medium, and long time to breakage. Training was done using fuzzy accelerators that adaptively adjust *momentum*, *learning rate*, and *steepness* parameters, after each iteration. These accelerators react to the rate at which learning commences. For example, if the error derivative was the same over several learning epochs and the decrease of the error was small, the learning rate was increased. The other parameters were adjusted in a similar fashion whereby the condition was evaluated using fuzzy reasoning. During training a correct classification of 99.1% was achieved for the neural network while the validation set achieved 86.8% correct classification. For the induction tree C4.5 was used. It was trained and tested with the same data as the neural net. Classes were assigned the linguistic label “low”, “medium”, and “high”. A subset of the classification rules was then used in the on-line detection as a fault isolator to explain the current situation. A typical classification rule is:

IF *Wire water pH* < 0.104
 AND *Retention aid flow* > 0.260
 THEN *classification* = high

This method achieved a 95.3% and 87.2 % correct classification for training and testing, respectively. The supplemental nature of neural nets and decision trees allows a high classification rate with simultaneous explanation of the outcome by aggregating the result. Further details on this application can be found in [9].

3.2 Method for automated tuning of a raw mix proportioning controller in cement plants

Problem. A cement plant basically grinds limestone, sandstone, and sweetener in a mill, and processes the

resulting powder in a kiln. To ensure the correct mix and proportions of chemical elements for making cement, it is critical to control the Raw Mix Proportioning (RMP) by keeping the output of the raw mill to be close to specified set points of physical properties (i.e., Lime Saturation Factor (LSF), Alumina Modulus (ALM), and Silica Modulus (SIM)). These properties are all functions of the fractions of four metallic oxides present in the material and can be picked up by sensors (IMACON and Quarcon) with a time delay of a few minutes. The control objective is then to regulate the values of LSF, ALM, and SIM, as they appear after the raw mill, so that they are close to the set points and fluctuate as little as possible. This can be done by continually changing the proportion in which the three raw materials are mixed before they enter the raw mill.

Solution. The proposed controller uses fuzzy logic supervisory control to minimize set point tracking error, while providing a smooth control action and insuring that there is no sudden control change that imposes shock to the system. The fuzzy logic supervisory controller (FLSC) uses the tracking error and its change-in-error to recommend a change in the control output that modifies the current control settings. Specifically, the FLSC consists of three pairs of fuzzy proportional integral (FPI) controllers. In the method described here, the FPI is tuned offline using genetic algorithms (GA) that modify FPI's most sensitive parameters: scaling factors (SF) and membership functions (MF).

The overall scheme is shown in Figure 2. It consists of FLSC closing the loop around the plant simulation (PSIM), using only the current setpoints as its state input. The actual setpoints are compared with the desired setpoints to generate an error as input to the FLSC, that in turn outputs control actions back to the simulator. Offline, a GA uses the setup for evaluating various FPI parameters. The design of the control algorithm requires a hierarchical system for handling multiple, possibly conflicting goals. The latter arise because the three properties (LSF, SIM, and ALM) have to be controlled simultaneously, and each may require a different proportion to be accurately controlled. In addition, LSF control has the highest priority of the three, and this priority could be numerically changed. The hierarchical controller module is illustrated in Figure 3 where the low level controllers will control the output to minimize the error in each of the three set points of LSF, SIM, and

ALM, while the supervisory controller will determine the best compromise among these control actions.

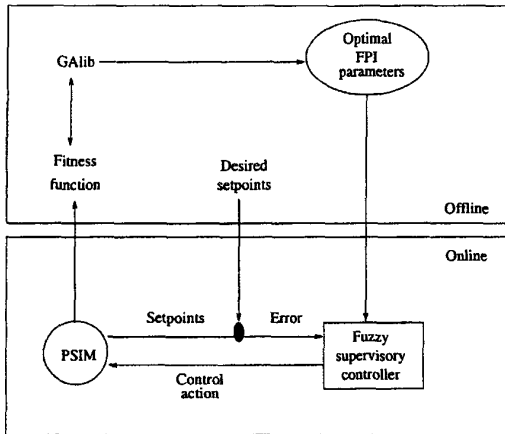


Figure 2. System schematic for automatic tuning of a fuzzy supervisory controller in cement plants

This hierarchical control scheme permits the decomposition of complex problems into a series of smaller and simpler ones. As these simpler problems are solved, typically by using low-level controllers, they are recombined to address the larger problem. A fuzzy logic supervisory controller that performs soft switching between different modes of operation governs this recombination. The soft switching allows more than one mode of operation (with its corresponding controller) to be active at any one time. By assigning a linear combination of low-level controllers to a given mode, the engineer can trade off safety and efficiency against performance.

Results. The GA parameters were population size = 50, number of generations to evolve = 25, crossover rate = 0.6, and mutation rate = 0.001. All structures in each generation were evaluated and an elitist strategy was used to guarantee monotone convergence. It was shown that this approach resulted in a controller that was superior to the manually designed one, and with only modest computational effort. This makes it possible to customize automated tuning to a variety of different cement plants. The tests demonstrated that GAs are powerful search methods and are very suitable for automated tuning of FPI controllers.

The Genetic Algorithms were able to come up with near-optimal FPI controllers within a reasonable amount of time according to different search criteria.

In addition, the tests were also designed to demonstrate that parameters tuning should be performed in the order of their significance. That is, tuning of scaling factors should come first since they have global effects on all the control rules in a rule base. Tuning membership functions will only give marginal improvements for a FPI with tuned scaling factors.

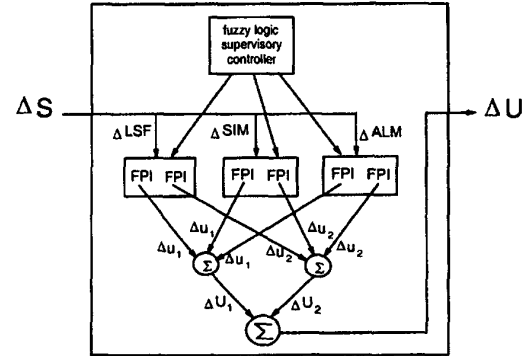


Figure 3. Fuzzy Logic Supervisory Controller module

The tuning of the Supervisory Fuzzy Controller, in conjunction with a plant simulator, had the objective of verifying that the system could exhibit stable operation for more than a week under severe external disturbances (in the oxide content of the raw material) as well as under parameter noise. The control system was tested in the Gujarat Ambuja Cement Limited (GACL) in Kodinar in India. The control system is commercially available from IMA, the manufacturer of the IMACON and Quarcon sensors. The interested reader can obtain further details in references [10-12].

3.3 Adaptive Classification for Gas Turbines Anomalies

Problem. Gas Turbines are used in many applications. They can range from stationary power plants to the use in airplanes or helicopters. Depending on the particular use, the design and size of the gas turbine will vary. On a coarse level, however, gas turbines use the same operating principles. To detect incipient failures, service providers have long tracked the behavior of the engine by measuring many system parameters and by using trend analysis to detect changes that might be indicative of developing failures [13]. Challenges of these schemes are that faults may not be recognized distinctively due to large amounts of noise as well as changing operating conditions that constantly move the estimate for "normal" operation.

The former is caused in part by changing environmental conditions for which corrections with first principle models or regression models work only to some extent. The latter is due to changes of schedules, maintenance, etc. which is not necessarily known to the analyst.

Solution. The solution used employed fuzzy clusters, a digital filter, and allowed the cluster to adapt to changing environments. Evaluation in multi-variate feature space helped distinguishing some faults because system variables correlated. For example, a bleed problem, where air leaks from the compressor, will result in a less efficient engine. However, in case of an aircraft engine, the controller demands a certain thrust level and will cause more fuel to be injected into the combustors. This in turn will raise both the turbine speed as well as the exhaust gas temperature. Any particular change may be too small to be picked up alone. However, in a three-dimensional space, the change is more pronounced and can be detected more easily. However, there is still the potential for high misclassification and a crisp classifier will not work very well. The approach chosen uses fuzzy clusters that have a lower degree of membership in the region of overlap. Figure 4 shows fuzzy clusters in the exhaust gas vs. turbine speed space.

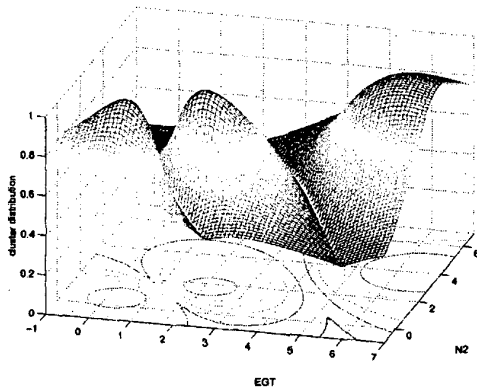


Figure 4: Fuzzy clusters in multi-dimensional space

Results. The system was tested with historical gas turbine data. It showed superior behavior compared to traditional trending approaches, which was manifested in a low false negative rate (none observed with the data available) and a much-reduced false positive rate, which improved, by several orders of magnitude. Figure 5 shows the adaptation of a cluster during operation. The “x” denote the location of the centroid

of cluster “normal” and the squares denote the location of a fault cluster. During a software change, the normal region changes considerably in one dimension as seen by the vertical jump of the operating point in the graph. Because there were no changes in the other two dimensions, the algorithm adapts correctly to the new location while retaining the ability to detect faults.

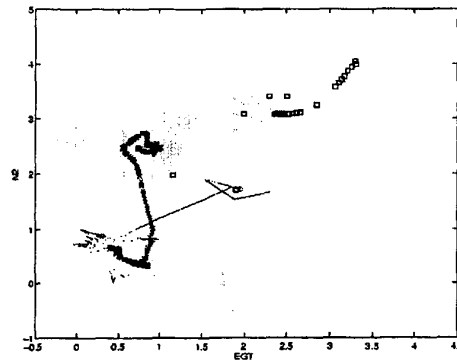


Figure 5. Adaptation of clusters during operation

4. Final Remarks

Soft computing (SC) is having an impact on many diagnostic industrial and commercial operations. It provides alternative approaches to traditional knowledge-driven reasoning systems or pure data-driven systems and it overcomes their shortcomings by synthesizing a number of complementary reasoning and searching methods over a large spectrum of problem domains. We have reviewed soft computing’s main components and surveyed some of their successful combinations that have lead to the development of hybrid SC systems.

These systems leverage the tolerance for imprecision, uncertainty, and incompleteness, which is intrinsic to the problems to be solved, and generate tractable, low-cost, robust solutions to such problems. The synergy derived from these hybrid systems stems from the relative ease with which we can translate problem domain knowledge into initial model structures whose parameters are further tuned by local or global search methods. This is a form of complementary or “tight” hybridization.

We have illustrated this synergy by describing several applications in diagnostics. These studies stem from real-world, high-impact business problems, such as gas

turbine service and diagnosis, voltage breakdown prediction, generator diagnostics, and paper web breakage prediction, and use a number of different means from the toolbox of soft computing.

The payoff for a conjunctive use of techniques is a more accurate and robust solution than a solution derived from the use of any single technique alone. This synergy comes at comparatively little expense because typically the methods do not try to solve the same problem in parallel but they do it in a mutually complementary fashion. Another way to say this is that the model needs a structure and parameters, and a search method to discover them, and no single technique should be expected to be the best for all problems.

A step in further improving system performance is the exploitation of parallel systems. These systems may be designed to rely to the maximum amount on non-overlapping data and use different techniques to arrive at their conclusions. In *information fusion*, the outputs of these heterogeneous models will be compared, contrasted, and aggregated.

The future appears to hold a lot of promise for the novel use and combinations of SC applications. The circle of SC's related technologies will probably widen beyond its current constituents. The push for low-cost solutions combined with the need for intelligent tools will result in the deployment of hybrid systems that efficiently integrate reasoning and search techniques.

On the application front we will likely see a drive towards prognostic and autonomous capabilities. With an increase of service-related operations, it will be increasingly attractive to be able to forecast anomalous trends and conditions, and correct them before their effects are fully developed. In addition, remotely monitored systems will bear the need to operate autonomously, thus requiring intelligent agents to regulate their operations.

In the future, we expect that the combination of soft computing with advances in the areas of computer vision, voice recognition, natural language processing, etc., will further improve and expand our problem-solving capability to a large spectrum of industrial and commercial problems.

5. References

- [1] P.P. Bonissone, V. Badami, K.H. Chiang, P.S. Khedkar, K. Marcelle, and M.J. Schutten. "Industrial Applications of Fuzzy Logic at General Electric", *Proceedings of the IEEE*, (83)3:450-465, 1995.
- [2] Y-T. Chen and P.P. Bonissone, "Industrial Applications of Neural Networks at General Electric", *Technical Information Series*, 98CRD79, General Electric CRD, October, 1998.
- [3] P.P. Bonissone, Y-T. Chen, K. Goebel, and P. Khedkar, "Hybrid Soft Computing Systems: Industrial and Commercial Applications", *Proceedings of the IEEE*, (87)9:1641-1667, 1999.
- [4] L.A. Zadeh, "Fuzzy Logic and Soft Computing: Issues, Contentions and Perspectives", *IIZUKA'94: 3rd Int. Conf. Fuzzy Logic, Neural Nets and Soft Computing*, Iizuka, Japan, 1994, pp. 1-2.
- [5] L.A. Zadeh, "Some reflection on soft computing, granular computing and their roles in the conception, design and utilization of information/intelligent systems", *Soft Computing A Fusion of Foundations, Methodologies and Applications*, Springer-Verlag, (2)1:23-25, 1998.
- [6] D. Dubois and H. Prade, "Soft computing, fuzzy logic, and Artificial Intelligence", *Soft Computing A Fusion of Foundations, Methodologies and Applications*, Springer-Verlag, (2)1: 7-11, 1998.
- [7] B. Bouchon-Meunier, R. Yager, and L.A. Zadeh, *Fuzzy Logic and Soft Computing*. World Scientific, Singapore, 1995.
- [8] P.P. Bonissone, "Soft Computing: the Convergence of Emerging Reasoning Technologies", *Soft Computing A Fusion of Foundations, Methodologies and Applications*, Springer-Verlag, (1)1: 6-18, 1997.
- [9] P.P. Bonissone, Y-T. Chen, and P. Khedkar. *System and Method for Predicting a Web Break in a Paper Machine*, US Patent #5,942,689, 1999.
- [10] P.P. Bonissone, Y-T. Chen, *System and Method for Providing Raw Mix Proportioning Control in a Cement Plant with a Fuzzy Logic Supervisory Control*, US Patent #6,113,256, 2000.
- [11] P.P. Bonissone, Y-T. Chen, *System and Method for Providing Raw Mix Proportioning Control in a Cement Plant with Gradient-Based Predictive Controller*, US Patent #6,120,173, 2000.
- [12] P.P. Bonissone, Y-T. Chen, *System and Method for Providing Raw Mix Proportioning Control in a Cement Plant*, US Patent #6,120,172, 2000.
- [13] D. Doel, "The Role for Expert Systems in Commercial Gas Turbine Engine Monitoring", *Proc. Gas Turbine and Aeroengine Congress and Exposition*, Brussels, Belgium, 1990.