

Prognostic and Diagnostic Monitoring of Complex Systems for Product Lifecycle Management: Challenges and Opportunities

Venkat Venkatasubramanian

Laboratory for Intelligent Process Systems

School of Chemical Engineering,

Purdue University

West Lafayette, IN 47907, USA.

Abstract

Modern technological advances have resulted in a myriad of complex systems, processes and products. These increasingly complicated processes, systems and products pose considerable challenges in their design, analysis, manufacturing and management for successful operation and use over their life cycles. In the process industries, for example, the maintenance and management of complex process equipment and processes, and their integrated operation, play a crucial role in ensuring the safety of plant personnel and the environment as well as the timely delivery of quality products. In the discrete parts industries, such as the auto industry, many product malfunctions are due to unanticipated dynamic interactions, due to repeated use or misuse of components. These interactions thrive in complex systems when the combined effects of uncertainty and operational adversity are not properly addressed either in design or in operation. Given the size, scope, and complexity of the systems and interactions it is becoming increasingly difficult for plant personnel to anticipate, diagnose and control serious abnormal events in a timely manner. Hence, it should come as no surprise that human operators tend to make erroneous decisions and take actions which make matters even worse, as reported in the literature. All these cost the companies and consumers in billions of dollars every year in product lifecycle management. Businesses and federal organizations are increasingly required to manage their entire products' life cycles to avoid costly failure or degradation in performance through service/maintenance, more robust design and control, and so on. These product life cycle management (PLM) issues present us with both major challenges and opportunities. There exist considerable incentives in developing appropriate prognostic and diagnostic methodologies for monitoring, analyzing, interpreting, and controlling such abnormal events in complex systems and processes. People in the process and product industries view this as the next major challenge in control systems research and application. In this paper, we will present an overview of these challenges and the opportunities. Recent progress has promising implications on the use of intelligent systems for product lifecycle management applications in the chemical, petrochemical, pharmaceutical and discrete parts

manufacturing industries.

1. Introduction

Recent technological advances have resulted in increasingly complicated processes, systems and products that pose considerable challenges in their design, analysis, manufacturing and management for successful operation and use over their life cycles. In the process industries, for example, the maintenance and management of complex process equipment and processes, and their integrated operation, play a crucial role in ensuring the safety of plant personnel and the environment as well as the timely delivery of quality products. Given the size, scope, and complexity of the systems and interactions it is becoming difficult for plant personnel to anticipate, diagnose and control serious abnormal events in a timely manner. In a large process plant, there may be as many as 1500 process variables observed every few seconds leading to information overload. Furthermore, the measurements may be insufficient, incomplete and/or unreliable due to a variety of causes such as sensor biases or failures. In addition, the emphasis on quick diagnosis aggravates the situation by causing psychological strains on plant personnel. In the discrete parts industries, one has similar problems. In the auto industry, for example, many product malfunctions are due to unanticipated dynamic interactions, due to repeated use or misuse of components. For example, unanticipated interactions between the power train and chassis of a car due to wet weather variations in the engine mount properties can cause the exhaust hangers to fail. Manufacturers then recall this car to fix the problem, but also unnecessarily recall thousands of other cars with the same exhaust system with no potential for failure. These interactions thrive in complex systems when the combined effects of uncertainty and operational adversity are not properly addressed either in design or in operation. Complex systems are challenging because they are highly interconnected among subsystems and components. It is their interconnectedness that makes them fragile when the cumulative effects of multiple abnormalities can cause systemic failures.

In the pharmaceuticals industry, the product life cycle management issues take on a somewhat different twist. Figure 1 illustrates the different stages a drug product passes through from discovery-to-delivery product pipeline. For instance, in contrast with continuous process manufacturing, the level of prognostic and diagnostic monitoring, quality control and automation in pharmaceuticals development and manufacture is rather limited. Hence, the industry makes do with limited computer-based assistance to acquire, manage, analyze and interpret complex product and processing information with enormous amounts of human intervention. This naturally leads to inefficiencies, uncertainties, costs, delays, and product quality concerns all along the product pipeline. Thus, it usually takes an enormous amount of time, effort and money to develop a new drug. A recent Tufts University study (Wall Street Journal, 2001) reported that the average cost of developing a new drug has more than doubled since 1987 to \$802 million, taking an average 12 years from discovery to market, a significant increase from the 9 year average reported in the

1980s. The need for more “robust” processes and better quality control were highlighted in a recent front page article in the *Wall Street Journal* (2003). In this business, the first to market captures the advantage of establishing a market position. For a successful pharmaceutical, a day gained or lost in the commercial life of a product typically represents \$1-40 million in revenue. Thus, the really compelling drivers are *speed to market* and *getting it right the first time*.

Given such difficult conditions, it should come as no surprise that human operators tend to make erroneous decisions and take actions which make matters even worse, as reported in the literature. Industrial statistics show that about 70% of the industrial accidents are caused by human errors. These abnormal events have significant economic, safety and environmental impact as seen in Union Carbide's Bhopal, India, accident and Occidental Petroleum's Piper Alpha accident. In the pharma industries, for example, there were more than 354 prescription drugs recalls in 2002, up from 176 in 1998. Recent events have shown that such large-scale plant accidents are not just things of the past but continue to haunt us even today. In this regard, we would like to bring your attention to three important major incidents. The first was the explosion at the Kuwait Petrochemical's Mina Al-Ahmedhi refinery in June of 2000. Fortunately, the human casualties were relatively low but financially the incident ranks among the top ten worst accidents with an estimated damage of about \$400 million. The next major recent incident was the explosion at the offshore oil platform of Petrobras, Brazil, and its subsequent sinking into the sea in March 2001. The estimated losses in this case are about \$5 billion. And last, in Sept 2001, the AZF chemical plant, Toulouse, had a large explosion that killed dozens of people and hundreds were treated for various kinds of injuries. It also disrupted the operations of the local university and its chemical engineering department which were situated near the plant site. Further, industrial statistics show that even though major catastrophes and disasters from chemical plant failures may be infrequent, minor accidents are very common, occurring on a day to day basis, resulting in many occupational injuries, illnesses, and costing the society billions of dollars every year (Bureau of Labor Statistics, 1998; National Safety Council, 1999).

All these concerns, and the lessons learnt from these accidents, have led the federal agencies in the U.S. to create tighter safety, health and environmental regulations. The Occupational Safety and Health Administration (OSHA) passed its PSM standard Title 29 CFR 1910.119, which requires all major chemical plant sites to perform process hazards analysis (PHA) (OSHA, 1992). In addition, EPA instituted the Risk Management Program (RMP) in 1995. Similar regulations are emerging in Europe as well (Laser, 2000). All these require the systematic identification of process hazards, their assessment and mitigation. Process Hazards Analysis is the systematic identification, evaluation and mitigation of potential process hazards which could endanger the health and safety of humans and cause serious economic losses. The importance of performing a comprehensive PHA is illustrated by Kletz (1988; 1991) with examples of industrial accidents that could have been prevented if only a thorough PHA had been performed earlier on that plant. AEM and PHA are two sides of the

same coin. Both are concerned with process safety: AEM is concerned about diagnosing abnormal causal origins of adverse consequences while PHA deals with reasoning about adverse consequences from abnormal causes. Intelligent, real-time, operator support systems are seen as a way to address both AEM and PHA. The automation of process fault detection and diagnosis forms the first step in automating supervisory control and AEM.

All these cost the companies and consumers in billions of dollars every year. It is estimated that the petrochemical industry in the U. S. incurs approximately \$20 billion in losses due to poor management of equipment and processes which lead to such abnormal situations. The cost is much more when one includes similar situations in other industries such as pharmaceutical, specialty chemicals, power, desalination and so on. Recently, Nucor Corporation Inc. paid \$100 million towards fines and remedies in a pollution control lawsuit. Similarly, accidents cost the British economy up to \$27 billion every year. U.S. manufacturers spend over \$7 billion annually recalling and renewing over 2,000 defective products. A far greater amount is spent on legal fees to fight lawsuits and warranty claims. All of these costs are associated with product lifecycle management and are on the rise. Businesses and federal organizations are increasingly required to manage their entire products' life cycles to avoid costly failure or degradation in performance through service/maintenance, more robust design and control, and so on. These product life cycle management (PLM) issues present us with both major challenges and opportunities. There exist considerable incentives in developing appropriate prognostic and diagnostic methodologies for monitoring, analyzing, interpreting, and controlling such abnormal events in complex systems and processes. People in the process and product industries view this as the next major challenge in control systems research and application.

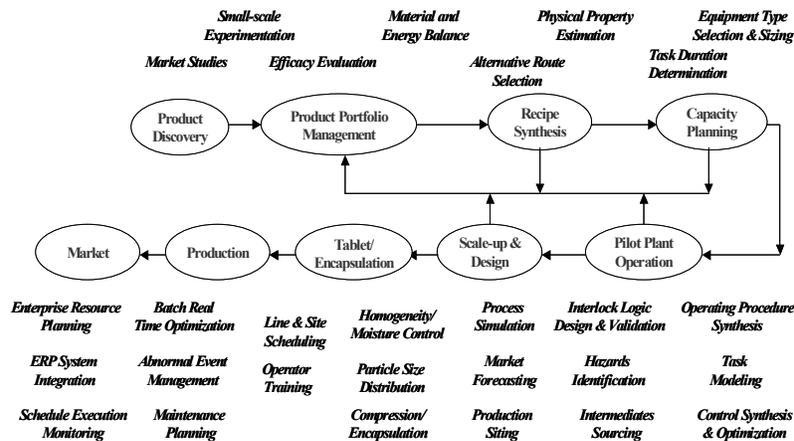


Figure 1: Product Lifecycle in the Pharmaceutical Industry

In the process industries, there are two different, but related, components of the overall abnormal events management (AEM) problem. One is process safety during real-time operations, and the other is safety in design. These two problem areas, with the application of intelligent systems concepts and tools, are now poised to play a dominant role in defining the course of process systems research and application for the coming decade. In this paper, we will present an overview of these two problem areas, the challenges and the opportunities. Due to the volume of literature on this subject and the limited length of this paper, this review is necessarily brief. For a more exhaustive treatment, the reader is referred to the review papers by Venkatasubramanian *et al.* (2003a; 2003b; 2003c). Recent progress has promising implications on the use of intelligent systems for product lifecycle management in the chemical, petrochemical, pharmaceutical and discrete parts manufacturing industries for better product quality, inherently safer design, operator training, abnormal events management and optimal process operations.

2. Prognostic and diagnostic systems: background

The term *fault* is generally defined as a departure from an acceptable range of an observed variable or a calculated parameter associated with a process (Himmelblau, 1978). This defines a fault as a process abnormality or symptom, such as high temperature in a reactor or low product quality and so on. The underlying cause(s) of this abnormality, such as a failed coolant pump or a controller, is (are) called the *basic event(s)* or the *root cause(s)*. The basic event is also referred to as a *malfunction* or a *failure*. Since one can view the task of diagnosis as a classification problem, the diagnostic system is also referred to as a diagnostic classifier. Figure 2 depicts the components of a general fault diagnostic framework. The figure shows a controlled process system and indicates the different sources of failures in it. In general, one has to deal with three classes of failures or malfunctions as described below: *Gross parameter changes in a model*: Parameter changes arise when there is a disturbance entering the process from the environment through one or more exogenous variables. An example is the change in the heat transfer coefficient due to fouling of a heat exchanger. *Structural changes*: Structural changes refer to changes in the model itself. They occur due to hard failures in equipment. An example is a controller failure which would imply that the manipulated variable is no longer functionally dependent on the controlled variable. *Malfunctioning sensors and actuators*: Gross errors usually occur with actuators and sensors. These could be due to a fixed failure, a constant bias (positive or negative) or an out-of-range failure.

Outside the scope of fault diagnosis are unstructured uncertainties, process noise and measurement noise. Unstructured uncertainties are mainly faults that are not modeled *a priori*. Process noise refers to the mismatch between the actual process and the predictions from model equations, whereas, measurement noise refers to high frequency additive component in the sensor measurements.

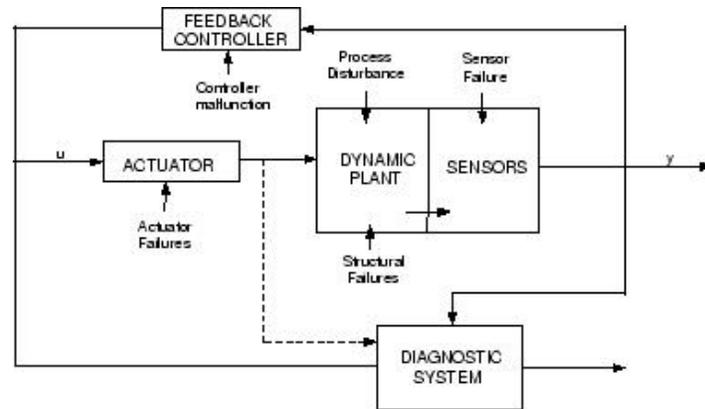


Figure 2. A general diagnostic framework

3. Desirable features of a diagnostic system

In order to compare various diagnostic approaches, it is useful to identify a set of desirable characteristics that a diagnostic system should possess. Then the different approaches may be evaluated against such a common set of requirements or standards. Though these characteristics will not usually be met by any single diagnostic method, they are useful to benchmark various methods in terms of the *a priori* information that is needed, reliability of solution, generality and efficiency of computation. In this context, one needs to understand two important concepts: completeness and resolution, before proceeding to the characteristics of a good diagnostic classifier. Whenever an abnormality occurs in a process, a general diagnostic classifier would come up with a set of hypotheses or faults that explains the abnormality. Completeness of a diagnostic classifier would require the actual fault(s) to be a subset of the proposed fault set. Resolution of a diagnostic classifier would require the fault set to be as minimal as possible. (1) *Quick detection and diagnosis*: The diagnostic system should respond quickly in detecting and diagnosing malfunctions. However, quick diagnosis and tolerable performance during normal operation are two conflicting goals (Willsky, 1976). A system that is designed to detect a failure (particularly abrupt changes) quickly will be sensitive to high frequency influences. This makes the system sensitive to noise and can lead to frequent false alarms during normal operation, which can be disruptive. This is analogous to the trade-off between robustness and performance noted in the control literature. (2) *Isolability*: Isolability is the ability of the diagnostic system to distinguish between different failures. Under ideal conditions free of noise and modeling uncertainties, this means that the diagnostic classifier should be able to generate output that is orthogonal to faults that have not occurred. There is also a trade-off between isolability and rejecting modeling uncertainties. (3) *Robustness*: One would like the diagnostic system to be robust to various noise and uncertainties. Its performance should degrade gracefully instead of failing totally and abruptly. This implies that the thresholds should be tuned conservatively. However, as

noted earlier, this can affect performance. (4) *Novelty Identifiability*: One of the minimal requirements of a diagnostic system is to be able to decide, given current process conditions, whether the process is functioning normally or abnormally, and if abnormal, whether the cause is a known malfunction or an unknown, novel, malfunction. This criterion is known as novelty identifiability. (5) *Classification error estimate*: An important practical requirement for a diagnostic system is in building the user's confidence on its reliability. This could be greatly facilitated if the diagnostic system could provide *a priori* estimates on classification error that can occur. Such error measures would be useful to project confidence levels on the diagnostic decisions by the system giving the user a better feel for the reliability of the recommendations by the system. (6) *Adaptability*: Processes in general change and evolve due to changes in external inputs or structural changes due to retrofitting and so on. In order to be useful and practical, the diagnostic system should be adaptable to changes. (7) *Explanation Facility*: A diagnostic system should also provide explanations on how a fault originated and propagated to the current situation. This is very important for building credibility with operators. One would like the system to not only justify why certain hypotheses were proposed but also explain why certain other hypotheses were not proposed. (8) *Modeling Requirements*: The amount of modeling required for system development is an important issue. For fast and easy deployment, the modeling effort should be as minimal as possible. (9) *Storage and Computational Requirements*: Usually, quick real-time solutions would require algorithms and implementations which are less computationally complex, but might entail high storage requirements. One would prefer a diagnostic system that is able to achieve a reasonable balance on these two competing requirements. (10) *Multiple Fault Identifiability*: The ability to identify multiple faults is an important but a difficult requirement. It is a difficult problem due to the interacting nature of most faults. In a general nonlinear system, the interactions would usually be synergistic and hence a diagnostic system may not be able to use the individual fault patterns to model the combined effect of the faults. On the other hand, enumerating and designing separately for various multiple fault combinations would become combinatorially prohibitive for large processes.

4. Classification of diagnostic approaches

The two main components in a diagnostic classifier are: (i) the type of knowledge used and (ii) the type of diagnostic search strategy. Diagnostic search strategy is usually strongly dependent on the knowledge representation scheme which in turn is largely influenced by the kind of *a priori* knowledge available. Hence, the type of *a priori* knowledge used is the most important distinguishing feature in diagnostic systems. In this review, we classify diagnostic systems based on the *a priori* knowledge used. The basic *a priori* knowledge that is needed for fault diagnosis is the set of failures and the relationship between the observations (symptoms) and the failures. A diagnostic system may have them explicitly (as in a table lookup), or it may be inferred from some source of domain knowledge. The *a priori* domain knowledge may be developed from a fundamental understanding of the process using first principles knowledge. Such knowledge is referred to as deep, causal or model-based

knowledge (Milne, 1987). On the other hand, it may be gleaned from past experience with the process. This knowledge is referred to as shallow, compiled, evidential or process history-based knowledge.

The model-based *a priori* knowledge can be broadly classified as qualitative or quantitative. The model is usually developed based on some fundamental understanding of the physics of the process. In quantitative models, this understanding is expressed in terms of mathematical functional relationships between the inputs and outputs of the system. In contrast, in qualitative model equations, these relationships are expressed in terms of qualitative functions centered around different units in a process. In contrast to the model based approaches where *a priori* knowledge about the model (either quantitative or qualitative) of the process is assumed, in process history based methods only the availability of large amount of suitably annotated historical process data is required. There are different ways in which this data can be transformed and presented as *a priori* knowledge to a diagnostic system. This is known as feature extraction. This can proceed as either quantitative or qualitative feature extraction. In quantitative feature extraction one can perform either a statistical or non-statistical feature extraction. This classification of diagnostic systems is shown in Figure 3.

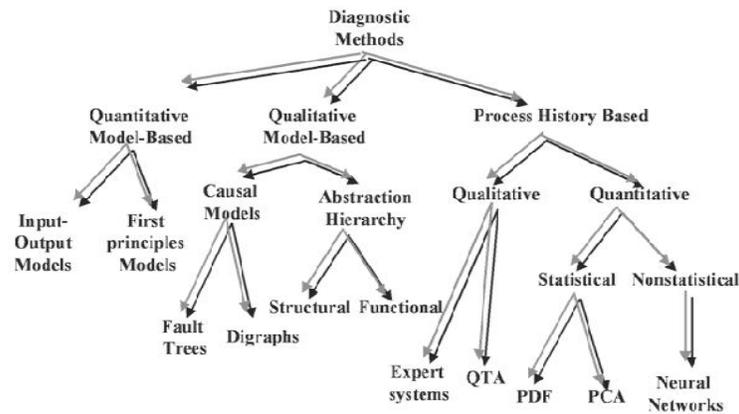


Figure 3. Classification of diagnostic methods

5. Quantitative model-based approaches

This section briefly reviews quantitative model-based fault diagnosis methods. We briefly summarize and evaluate the most frequently-used fault detection and isolation approaches, such as observers, parity relations, Kalman filters and parameter estimation. For the sake of brevity, we have included only some of the many techniques available and therefore the references listed in this review paper are by no means exhaustive. Once again, for a more complete review the reader is referred to Venkatasubramanian et al. (2003a; 2003b; 2003c). In automatic control, change/fault detection problems are known as model-based fault

detection and isolation (FDI). Relying on an explicit model of the monitored plant, all model-based FDI methods (and many of the statistical diagnosis methods) require two steps. The first step generates inconsistencies between the actual and expected behavior. Such inconsistencies, also called *residuals*, are “artificial signals” reflecting the potential faults of the system. The second step chooses a decision rule for diagnosis.

The check for inconsistency needs some form of redundancy. There are two types of redundancies, hardware redundancy and analytical redundancy. The former requires redundant sensors. It has been utilized in the control of such safety-critical systems as aircraft, space vehicles and nuclear power plants. However, its applicability is limited due to the extra cost and additional space required. On the other hand, analytical redundancy is achieved from the functional dependence among the process variables and is usually provided by a set of algebraic or temporal relationships among the states, inputs and the outputs of the system (Chow and Willsky, 1984; Basseville, 1988). The essence of analytical redundancy is to check the actual system behavior against the system model for consistency. Any inconsistency, expressed as residuals, can be used for detection and isolation purposes. The residuals should be close to zero when no fault occurs but show “significant” values when the underlying system changes. To generate the diagnostic residuals requires an explicit mathematical model of the system. The model may be obtained either analytically using first principles or empirically as a black-box model. Also, statistical methods are often required for the decision making.

Most of the work on model-based diagnostic systems has so far been concentrated in the aerospace, mechanical and electrical engineering literature. There has not been much work on the application of observers for fault diagnosis in chemical process systems. This may be due to the fact that the objectives of state/parameter estimation techniques when used in process control are different from the objectives of fault diagnosis as pointed out by Watanabe *et al.* (1994). Further, the unavailability and/or the complexity of high fidelity models and the essential nonlinear nature of these models for chemical processes render the design of diagnostic observers for such systems quite difficult. Gertler and Luo (1989) presented a design procedure to generate isolable parity equation models chosen from a multitude of suitable models on the basis of sensitivities with respect to different failures and robustness relative to uncertainties in selected parameters. They illustrated the application of the technique on a distillation column. The application to EKF-based FDI system for fault diagnosis in the Model IV FCCU case study involving DAEs was reported by Huang *et al.* (2000). The application of three unknown input observers (UIO), one a *linear*, second an *extended linear* and the third a *non-linear* UIO on a CSTR case study is discussed in Dash *et al.* (2001; 2002). In their work, the performance of these three observers is evaluated through extensive simulation studies.

The type of models the analytical approaches can handle are limited to linear, and in some cases, to very specific nonlinear models. For a general nonlinear model, linear

approximations can prove to be poor and hence the effectiveness of these methods might be greatly reduced. Another problem in this approach is the simplistic approximation of the disturbances that include modeling errors. In most cases, the disturbance matrix includes only additive uncertainty. However, in practice, severe modeling uncertainties due to parameter drifts come in the form of multiplicative uncertainties. This is a general limitation of all the model-based approaches that have been developed so far. In addition to difficulties related to modeling, they do not support an explanation facility owing to their procedural nature. Further, *a priori* estimation of classification errors can not also be provided using these methods. Another disadvantage with these methods is that if a fault is not specifically modeled (novelty identifiability), there is no guarantee that the residuals will be able to detect it. Adaptability of these approaches to varying process conditions has also not been considered. When a large-scale process is considered, the size of the bank of filters can be very large increasing the computational complexity, though, with the recent increase in computational power and the essential linear nature of these problems, this might not be a serious bottle-neck.

6. Qualitative model-based methods

In this part of the paper, we briefly review qualitative model representations and search strategies used in fault diagnostic systems. Qualitative models are usually developed based on some fundamental understanding of the physics and chemistry of the process. Various forms of qualitative models such as causal models and abstraction hierarchies have been developed. In terms of search strategies, one can broadly classify them as topographic and symptomatic search techniques. Topographic searches perform malfunction analysis using a template of normal operation, whereas, symptomatic searches look for symptoms to direct the search to the fault location.

In the causal models approach, people have developed systems based on signed digraphs (Iri *et al.*, 1979; Venkatasubramanian and Rich, 1988), qualitative simulation (Kuipers, 1985), qualitative process theory (Forbus, 1984), causal or precedence ordering techniques (Iwasaki and Simon, 1986) and so on. There are two main problems with qualitative models: ambiguities and spurious solutions. Ambiguities can be resolved completely only through the use of actual quantitative values. Frameworks for reasoning about relative orders of magnitudes have been proposed by (Raiman, 1986; Mavrovouniotis and Stephanopoulos, 1987). In these frameworks, influence magnitudes are related using relations such as A is negligible compared to B, A is close to B and A is the same order of magnitude as B. A set of inference rules then generates a partial ordering of values into groups significantly different in magnitude (Ungar and Venkatasubramanian, 1990). Spurious solutions refer to the generation of physically unrealizable solutions by a qualitative process. This problem can be alleviated to a reasonable extent by modeling the system from different perspectives (Kuipers, 1985; Kay and Kuipers, 1993).

Hierarchical abstraction of mass and energy flow structures at different levels of function (called Multilevel Flow Models), *i.e.*, functional abstraction hierarchy, has been used by Lind (1991). An MFM model is a normative description of a system, a representation of its function in terms of what should be done, how should it be done and with what should it be done. This leads to three basic concepts in MFM: (i) Goals, (ii) Functions and (iii) Physical components. Three types of connective relations such as (i) achieve relations, (ii) condition relations and (iii) realize relations are used to connect objects. Diagnostic reasoning strategies based on the MFM model can be found in (Larsson, 1994).

Though qualitative models have a number of advantages as noted above, the major disadvantage is the generation of spurious solutions. Considerable amount of work has been done in the reduction of spurious solutions while reasoning with qualitative models. In SDGs, this is done using generation of latent constraints and similar techniques have been proposed for qualitative physics based models such as QSIM. The search strategies can be classified as either topographic or symptomatic search. Clearly, for a given qualitative representation, different search strategies could be used for diagnosis. Hence, one can view the methods proposed in the literature as different combinations of the qualitative models and search strategies.

7. Process history-based methods

In this section, fault diagnosis methods based on historic process knowledge and data are reviewed. In contrast to the model based approaches where *a priori* knowledge (either quantitative or qualitative) about the process is needed, in process history based methods, a large amount of historical process data that is suitably annotated is needed. There are different ways in which this data can be transformed and presented as *a priori* knowledge to a diagnostic system. This is known as feature extraction. This extraction process can be either qualitative or quantitative in nature. Two of the major methods that extract qualitative history information are the expert systems (Davis, 1984; Rich and Venkatasubramanian, 1987) and trend modeling methods (Cheung and Stephanopoulos, 1990; Bakshi and Stephanopoulos, 1992; Rengaswamy and Venkatasubramanian, 1995; Vedam and Venkatasubramanian, 1997). Methods that extract quantitative information can be broadly classified as non-statistical or statistical methods. Neural networks are an important class of non-statistical classifiers (Leonard and Kramer, 1990; Kavuri and Venkatasubramanian, 1994). PCA/PLS and statistical pattern classifiers form a major component of statistical feature extraction methods (Kramer, 1991; Nomikos and MacGregor, 1994; Dong and McAvoy, 1996; Dunia *et al.*, 1996).

It is impossible to adequately review the abundant literature on expert systems for fault diagnosis here. A more detailed review of this is presented elsewhere (Venkatasubramanian *et al.*, 2003c). Most of the successful expert systems applications have been rule-based systems for structured selection or heuristic classification perspective of diagnosis,

particularly for medical diagnosis. They often use some kind of probabilistic reasoning such as certainty factors, Bayesian methods or fuzzy logic to handle the uncertainties. Such systems, while quicker to develop and implement, are of limited scope, valid only for the processes for which they were developed.

In the trend representation approach, Cheung and Stephanopoulos (1990) presented a formal framework for the representation of process trends. They introduce triangulation to represent trends. Janusz and Venkatasubramanian (1991) propose a comprehensive set of primitives using which any trend can be represented. They use a finite difference method to calculate the first and second derivative of the process trend changes and based on these values, the primitives are identified. This qualitative formalism readily lends itself to hierarchic representations as well. Rengaswamy and Venkatasubramanian (1995) have shown how primitives can be extracted from raw noisy sensor data by treating the problem of primitive identification as a classification problem using neural networks. Vedam and Venkatasubramanian (1997) proposed a wavelet theory based adaptive trend analysis framework and later proposed a dyadic B-Spline based trend analysis algorithm (Vedam *et al.*, 1998). Recently, Rengaswamy *et al.* (2001) have discussed the utility of trend modeling in control loop performance assessment.

Multivariate statistical methods such as PCA and PLS have been used in diagnosis with considerable success (Qin and McAvoy, 1992; Nomikos and MacGregor, 1994; Dong and McAvoy, 1996; Vedam and Venkatasubramanian, 1997; Yoon and MacGregor, 2001). Overviews of using PCA and PLS in process analysis and control, fault detection and diagnosis were given by MacGregor *et al.* (1991; 1994) and MacGregor and Kourti (1995).

In an earlier work, Kresta *et al.* (1991) laid out the basic methodology of using multivariate statistical process control procedure to handle large numbers of process and quality variables for continuous process. Later on, Nomikos and MacGregor (1994) extended the use of multivariate projection methods to batch processes by using multiway PCA. To deal with nonlinearity, Qin and McAvoy (1992) proposed a neural net PLS approach that incorporated feedforward networks into the PLS modeling. In order to handle nonlinearity in batch processes, Dong and McAvoy (1996) utilized a nonlinear principal component analysis method. To facilitate the diagnosis procedure in very large processes, new hierarchical multivariate monitoring methods based on multiblock PLS algorithm was presented by MacGregor *et al.* (1991; 1994). Raich and Cinar (1996) proposed an integral statistical methodology combining principal component analysis and discrimination analysis techniques. Based on angle discriminants, a novel disturbance diagnosis approach (Raich and Cinar, 1997) was later introduced that provides better results for cases in which distance based discrimination is not accurate enough. Recently, Yoon and MacGregor (2000) have discussed the use of statistical and causal model-based methods for fault diagnosis.

Considerable interest has been shown in the literature in the application of neural networks for the problem of fault diagnosis (Himmelblau, 1986; Venkatasubramanian and Chan, 1989; Kavuri and Venkatasubramanian, 1994; Bulsari, 1995). Neural networks have been proposed for classification and function approximation problems. In general, neural networks that have been used for fault diagnosis can be classified along two directions: (i) the architecture of the network such as sigmoidal, radial basis and so on and, (ii) the learning strategy such as supervised and unsupervised learning. The most popular supervised learning strategy in neural networks has been the back-propagation algorithm. There are a number of papers that address the problem of fault detection and diagnosis using back-propagation neural networks. In chemical engineering, Venkatasubramanian (1985), Watanabe *et al.* (1989) and Venkatasubramanian and Chan (1989), Ungar *et al.* (1990) and Hoskins *et al.* (1991) were among the early researchers to demonstrate the usefulness of neural networks for fault diagnosis. Later, a more detailed and thorough analysis of the learning, recall and generalization characteristics of neural networks was presented by Venkatasubramanian *et al.* (1990) and Vaidhyanathan and Venkatasubramanian (1992). A hierarchical neural network architecture for the detection of multiple faults was proposed by Watanabe *et al.* (1994).

8. A comparison of various approaches

We have reviewed so far the three conceptually different frameworks for process fault diagnosis. In this section, we provide a comparative evaluation of these different frameworks against a common set of desirable characteristics for a diagnostic system that we proposed in section 3. The evaluations are summarized in Table 1. It is clear from the table that no single method is adequate to handle all the requirements for a desirable diagnostic system. Though all the methods are restricted, in the sense that they are only as good as the quality of information provided, it is seen that some methods might better suit the knowledge available than others. It is our view that some of these methods can complement one another resulting in better diagnostic systems. Integrating these complementary features is one way to develop hybrid methods that could overcome the limitations of individual solution strategies. Hence, hybrid approaches where different methods work in conjunction to solve parts of the problem are attractive.

Combination of methods allows one to evaluate different kinds of knowledge in one single framework for better decision making. A blackboard-based cooperative problem-solving framework where different diagnostic methods work in conjunction to perform collective process fault diagnosis has been proposed by (Mylaraswamy, 1996; Mylaraswamy and Venkatasubramanian, 1997; Vedam and Venkatasubramanian, 1999). The blackboard architecture, called Dkit, in which different diagnostic methods analyze the same problem, and a scheduler which regulates the decision-making of these methods, is the central concept in this framework. The utility of such a hybrid framework for solving real-time complex fault diagnosis problems is illustrated through the use of a diagnosis study on the Amoco Model IV Fluid Catalytic Cracking Unit (FCCU). This framework was adopted by the Honeywell

ASM Consortium for the development of a commercially-viable, next generation, intelligent control systems called AEGIS and MSEP.

Table 1. Comparison of various diagnostic methods

	Observer	Digraphs	Abstraction Hierarchy	Expert Systems	QTA	PCA	Neural Networks
Quick Detection and Diagnosis	✓	?	?	✓	✓	✓	✓
Isolability	✓	×	×	✓	✓	×	✓
Robustness	✓	✓	✓	✓	✓	✓	✓
Novelty	?	✓	✓	×	?	✓	✓
Identifiability							
Classification Error	×	×	×	×	×	×	×
Adaptability	×	✓	✓	×	?	×	×
Explanation Facility	×	✓	✓	✓	✓	×	×
Modelling Requirement	?	✓	✓	✓	✓	✓	✓
Storage & Computation	✓	?	?	✓	✓	✓	✓
Multiple Fault Identifiability	✓	✓	✓	×	×	×	×

 FAVORABLE
 NOT FAVORABLE
 SITUATION DEPENDENT
 NOT APPLICABLE

9. Intelligent systems for process hazards analysis

As noted earlier, AEM and PHA are closely related activities that deal with process safety and operability issues and hence have an impact on supervisory control decisions. Again, concerns about process safety have led the federal agencies in the U.S. to create stricter safety, health and environmental regulations that require the performance of process hazards analysis. An estimated \$5 billion is spent annually by the chemical process industries (CPI) to perform PHAs and related activities. The estimated cost of process hazards reviews in the CPI is about 1% of sales or about 10% of profits.

Given the enormous amounts of time, effort and money involved in performing PHA reviews, there exist considerable incentives to develop intelligent systems for automating the process hazards analysis of chemical process plants. An intelligent system can reduce the time, effort and expense involved in a PHA review, make the review more thorough, detailed, and consistent, minimize human errors, and free the team to concentrate on the more complex aspects of the analysis which are unique and difficult to automate. Also, an intelligent PHA system can be integrated with CAD systems and used during early stages of design, to identify and decrease the potential for hazardous configurations in later design phases where making changes could be economically prohibitive. It would facilitate automatic

documentation of the results of the analysis for regulatory compliance. Also these PHA results can be made available online to assist plant operators during diagnosis of abnormal situations as well as to train novice operators.

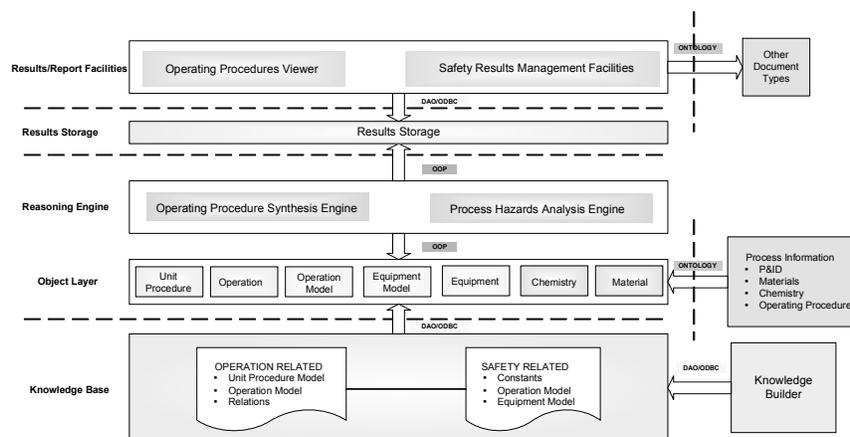


Figure 4. Information flow and overall structure of PHASuite

Several attempts for automating PHA have been made in the past two decades. A detailed survey of the literature on intelligent systems for HAZOP analysis can be found in Venkatasubramanian et al. (2000). Based on our previous efforts in LIPS on intelligent systems for PHA (Vaidhyanathan et al. 1996, Srinivasan & Venkatasubramian 1998a, 1998b), we developed PHASuite, Process Hazards Analysis Suite (Zhao, 2002). PHASuite was built based on a comprehensive knowledge engineering framework. From function point of view, the framework consists of four main parts: information sharing, representation, knowledge base and reasoning engine. Ontology based information sharing schemes are developed to share process information and results with other systems. Colored Petri Nets based representation, created from process information, is used for representing chemical processes as well as the methodology for HAZOP analysis. In this framework, a process is decomposed into two abstraction levels, operation level and equipment level, which are bridged using functional representation. Analysis is carried out on both of the levels, i.e. integration of operation-centered analysis and equipment-centered analysis. Knowledge used in this system is divided into operation knowledge and safety knowledge, both of which are stored in models. Models, general model and specific model, which are created based on general models to incorporate the experience system gains from analyzing different processes, form the case base. Case-based techniques are adopted for knowledge management. Knowledge base is stored externally in structured databases. A two-level, two-layer reasoning engine is designed to operate on the Petri Nets representation of the process using the knowledge base to perform HAZOP analysis. Software engineering methodologies have been applied to guide the design and development of PHASuite in order to achieve the goal of efficiency, flexibility and high quality. Layered and

repository software architecture have been designed to handle the complex information flow and the multipart knowledge base in the system. Objected-oriented and component-oriented methodologies have been adopted in the design and implementation. Unified modeling language is used in design and documentation of development details. Figure 4 illustrates the information flow and overall structure of PHASuite.

10. Conclusions and future directions

The complexity of the modern processes, systems and products pose considerable challenges in their design, analysis, manufacturing and management for successful operation and use over their life cycles. In the process industries, for example, the maintenance and management of complex process equipment and processes, and their integrated operation, play a crucial role in ensuring the safety of plant personnel and the environment as well as the timely delivery of quality products. The costs of these product lifecycle management problems runs into billions of dollars. These concerns can be effectively addressed by the emerging intelligent prognostic and diagnostic monitoring and control systems paradigm which has the goal of automating supervisory control tasks currently handled manually by human operators. The basic aim of this paper was to give a broad overview of the various approaches to automated fault diagnosis and describe the state-of-the-art efforts in terms of industrial applications in the field. We also presented the relevance of automated process hazards analysis to abnormal events management in product lifecycle management. The intelligent systems, such as PHASuite, developed at Purdue University are now well beyond proof of concept and are ready for industrial applications and commercial exploitation. In addition to routine PHA, such intelligent systems can facilitate HAZOP reviews at an early stage of process development and design. This means that problems can be identified and rectified during detailed design or while formulating operating procedures. Making changes once a plant is built are very expensive compared with changes at the design stage (Skelton, 1997). Early identification of hazards will also lead to effective avoidance or control of such hazards.

The application of the intelligent systems framework for complex problems such as abnormal situation management and process hazards analysis, which were formerly solved only by human teams, has come a long way since its modest beginnings in the 1980s. Intelligent systems are now well poised to make significant contributions to AEM, PHA and PLM in real-life industrial settings and revolutionize the prognostic and diagnostic monitoring of complex systems in the coming decade in a wide variety of industries.

References

- ASM Consortium Web Site: <http://www.iac.honeywell.com/pub/absitmang/>
Bakshi, B. and G. Stephanopoulos (1992). *IFAC Symposium on Online Fault Detection and Supervision in the Chemical Process Industries*.
Basseville, M. (1988). *Automatica*, **24** (3), 309-326.

- Bulsari (Editor), A. B. (1995). *Neural networks for chemical engineers*. Elsevier Science.
- Bureau of Labor Statistics (1998). Occupational injuries and illnesses in the United States by industry. Government Printing Office, Washington, DC.
- Cheung, J. T. and G. Stephanopoulos (1990). *Comput. & Chem. Engng.*, **14**, 495-510.
- Chow, E. D. and A. S. Willsky (1984). *IEEE Trans. Automat. Contr.* **29** (7), 603-614.
- Dash, S., S. Kantharao, R. Rengaswamy and V. Venkatasubramanian (2001). *ESCAPE-11*, 853-858.
- Dash, S., P. Vachhani, R. Rengaswamy and V. Venkatasubramanian (2002). *Chemical Engineering Science*.
- Davis, R. (1984). *Artif. Intell.* **24**, 347-410.
- Dong, D. and T. J. McAvoy (1996). *AIChE J.*, **42** (8), 2199-2208.
- Dunia, R., S. J. Qin, T. F. Edgar and T. J. McAvoy (1996). *AIChE J.*, **42**(10), 2797-2812.
- Forbus, K. D. (1984). *Artif. Intell.*, **24**, 85-168.
- Gertler, J. and Q. Luo (1989). *AIChE J.*, **31**(11), 1856-1868.
- Himmelblau, D. M. (1978). Fault detection and diagnosis in chemical and petrochemical processes. Elsevier Press, Amsterdam.
- Himmelblau, D. M. (1986). *IFAC Kyoto Workshop on fault detection and Safety in Chemical Plants*. Kyoto, Japan.
- Hoskins, J. C., K. M. Kaliyur and D. M. Himmelblau (1991). *AIChE J.*, **37**, 137-141.
- Huang, Y., S. Dash, G. V. Reklaitis and V. Venkatasubramanian (2000). *SAFEPROCESS 2000, 14-16 June*. Budapest, Hungary.
- Iri, M., K. Aoki, E. O'Shima and H. Matsuyama (1979). *Comput. & Chem. Engng.*, **3**, 489-493.
- Iwasaki, Y. and H. Simon (1986). *Artificial Intelligence*, **29**, 3-32.
- Janusz, M. and V. Venkatasubramanian (1991). *Engng. Applic. Artif. Intell.*, **4**, 329-339.
- Kavuri, S. N. and V. Venkatasubramanian (1994). *Int. J. Control*, **59**(3), 767-792.
- Kay, H. and B. Kuipers (1993). *Proceedings of AAAI-93*, 606-613.
- Kletz, T. A. (1988). *What Went Wrong?* 2nd edition, Gulf Publishing, Houston, Texas.
- Kletz, T. A. (1991). *Journal of Loss Prev. Process Ind.*, **4**, 128-129.
- Kramer, M. A. (1991). *AIChE J.*, **37** (2), 233-243.
- Kresta, J. V., J. F. MacGregor and T. E. Marlin (1991). *Can. J. Chem. Eng.*, **69**, 35.
- Kuipers, B. (1985). *Proceedings of Ninth Joint International Conference on Artificial Intelligence*.
- Larsson, J. E. (1994). *Automatica*, **30** (5), 775-787.
- Laser, M. (2000). *Trans IchemE* **78**(B), 419-422.
- Leonard, J. A. and M. A. Kramer (1990). *AIChE Annual Meeting, Chicago*.
- Lind, M. (1991). *IFAC Workshop on Computer Software Structures Integrating AIKBS Systems in Process Control*.
- MacGregor, J. F., T. E. Marlin, J. Kresta and B. Skagerberg (1991). *Chemical Process Control, CPCIV* (Y. Arkun and W. H. Ray, Eds.). 79-100.
- MacGregor, J. F., J. Christiana K. Costas and M. Kotoudi (1994). *AIChE J.*, **40**(5), 826-838.
- MacGregor, J. F. and T. Kourti (1995). *Control Engng. Practice*, **3** (3), 403-414.

- Mavrovouniotis, M. and G. Stephanopoulos (1987). *Proceedings of AAAI-8*.
- Milne, R. (1987). *IEEE Trans. Syst., Man, and Cybernetics*, **17**(3), 333-339.
- Mylaraswamy, D. (1996). *PhD Thesis*, Purdue University.
- Mylaraswamy, D. and V. Venkatasubramanian (1997). *Comput. & Chem. Engng* **21**, S935-S940.
- National Safety Council (1999). *Chem. Engng. Prog.*, **91**(9), 36-45.
- Nomikos, P. and J. F. MacGregor (1994). *AIChE J.*, **40**, 1361-1375.
- OSHA (1992). *Process safety management of highly hazardous chemicals; explosives and blasting agents; final rule*. 29 CFR 1910.119.
- Qin, S. J. and T. J. McAvoy (1992). *Comput. & Chem. Engng.*, **16** (4), 379-391.
- Raich, A. and A. Cinar (1996). *AIChE J.*, **42**, 995-1009.
- Raich, A. and A. Cinar (1997). *Comput. & Chem. Engng.*, **21** (6), 661-673.
- Raiman, O. (1986). *Proceedings of AAAI-86*.
- Rengaswamy, R. and V. Venkatasubramanian (1995). *Engng. Applic. Artif. Intell.*, **8**(1), 35-51.
- Rengaswamy, R., T. Hagglund and V. Venkatasubramanian (2001). *Engng. Applic. Artif. Intell.*, **14**(1), 23-33.
- Rich, S. H. and V. Venkatasubramanian (1987). *Comput. & Chem. Engng.*, **11** (2).
- Skelton, B. (1997). *Process safety analysis: an introduction*. Houston: Gulf Publishing.
- Srinivasan, R. and V. Venkatasubramanian (1998a). *Comput. & Chem. Engng.*, **22**(9).
- Srinivasan, R. and V. Venkatasubramanian (1998b). *Comput. & Chem. Engng.*, **22**(9).
- Ungar, L. H., B. A. Powell and S. N. Kamens (1990). *Comput. & Chem. Engng.* **14**.
- Ungar, L. H. and V. Venkatasubramanian (1990). *Artificial Intelligence in Process Systems Engineering: Knowledge Representation*. CACHE, Austin, Texas.
- Vaidyanathan, R. and V. Venkatasubramanian (1992). *Engng. Applic. Artif. Intell.*, **5** (1).
- Vaidyanathan, R., V. Venkatasubramanian and F. Dyke (1996). *Process Safety Progress* **15**(2).
- Vedam, H. and V. Venkatasubramanian (1997). *American Control Conference*, 309-313.
- Vedam, H., V. Venkatasubramanian and R. Bhalodia (1998). *Comput. & Chem. Engng.*, **22**.
- Vedam, H., S. Dash and V. Venkatasubramanian (1999). *Comput. & Chem. Engng.*, **23S**.
- Venkatasubramanian, V. (1985). *Second Conference on Artificial Intelligence Applications: The Engineering of Knowledge-Based Systems* Washington, DC, USA.
- Venkatasubramanian, V. and S. H. Rich (1988). *Comput. & Chem. Engng.*, **12** (9).
- Venkatasubramanian, V. and K. Chan (1989). *AIChE J.*, **35**.
- Venkatasubramanian, V., R. Vaidyanathan and Y. Yamamoto (1990). *Comput. & Chem. Engng.*, **14** (7).
- Venkatasubramanian, V., R. Rengaswamy, K. Yin and S. N. Kavuri (2003a). *Comput. & Chem. Engng.*, **27**(3), 293-311.
- Venkatasubramanian, V., R. Rengaswamy and S. N. Kavuri (2003b). *Comput. & Chem. Engng.*, **27**(3), 313-326.
- Venkatasubramanian, V., R. Rengaswamy, S. N. Kavuri and K. Yin (2003c). *Comput. & Chem. Engng.*, **27**(3), 327-346.

Wall Street Journal (2001). Dec. 2001.

Wall Street Journal (2003). Sept. 2001.

Watanabe, K., I. Matsura, M. Abe, M. Kubota and D. M. Himmelblau (1989). *AIChE J.*, **35** (11).

Watanabe, K., S. Hirota, L. Hou and D. M. Himmelblau (1994). *AIChE J.*, **40** (5).

Yoon, S. and J. F. MacGregor (2000). *AIChE J.*, **46**(9).

Yoon, S. and J. F. MacGregor (2001). *Journal of Process Control*, **11**.

Zhao, C., 2002, Ph.D. Thesis, Purdue University.