

**Intelligent Process Supervision via  
Automated Data Validation and Fault Analysis:  
Results of Actual CPI Applications**

Richard J. Fickelscherer<sup>1</sup>, Douglas H. Lenz<sup>1</sup>, Daniel L. Chester<sup>2</sup>

<sup>1</sup> FALCONEER Technologies Company, LLC, Williamsville, NY

<sup>2</sup>Department of Computer & Information Sciences, University of Delaware, Newark, DE

Prepared for Presentation at the

AICHE 2003 Spring National Meeting  
New Orleans, Louisiana  
March 30 – April 3, 2003

Using Information Technology for Increased Profitability and Productivity

Copyright © R.J. Fickelscherer, D.H. Lenz, D.L. Chester  
April 2003  
Unpublished

AICHE shall not be responsible for statements or opinions contained in papers or printed  
in its publications.

## ABSTRACT

Process Fault Analyzers are computer programs that can monitor process operations to identify the underlying cause(s) of operating problems. A general method for creating process fault analyzers for chemical and nuclear processing plants has been sought ever since the incorporation of computers into process control. The motivation has been the enormous potential for improving process plant operations in terms of safety and productivity. Automated process fault analysis should help process operators

1. prevent catastrophic operating disasters such as explosions, fires, meltdowns, toxic chemical releases, etc.;
2. reduce downtime after emergency process shutdowns;
3. eliminate unnecessary process shutdowns; and
4. maintain better quality control of the desired process products.

The **Method of Minimal Evidence (MOME)** is a model-based diagnostic strategy. MOME directly analyzes for various possible underlying process problems based on the behavior of engineering models evaluated with measured process data. It does so by providing a uniform framework for examining models of normal process operation and their associated modeling assumptions. MOME also can be directly used to diagnose many multiple fault situations, to determine the optimal placement of process sensors to facilitate fault analysis, and to determine the optimal division of a large process system for distributing fault analyzers.

MOME has now been used to hand compile the **Data Validation and Fault Analysis (DV&FA)** diagnostic rules necessary to competently perform such analysis in two commercial-scale facilities: one an adipic acid plant owned and operated by DuPont in Victoria, Texas and the other a electrolytic sodium persulfate plant owned and operated by FMC in Tonawanda, New York. The MOME strategy provides a means to directly perform continuous, on-line, real-time fault analyses at these facilities.

## **I. Introduction**

The measures taken in the past to help operators perform process fault management have not been able to provide them the support that they need to totally eliminate process accidents. Typically these accidents have very simple origins (Kletz, 1985) (Lieberman, 1985). The reasons that these accidents have occurred is because the number of possible process failures which need to be considered and the amount of process data which needs to be analyzed commonly exceed those that an operator can always effectively cope with in emergency situations. This makes automating process fault analysis an attractive proposition; however, to date this technology is still largely unused by the CPI (Venkatasubramanian 2001, IFAC 1995, Dhurjati 1992, De Heer 1987, Lees 1983).

The targeted use of automated fault analysis should exploit the relative advantages of automated analysis versus that which can be done by humans. Computers can outperform humans in doing numerous, precise, and rapid calculations, and in making associative and inferential judgements (Sheridan, 1981). On the other hand, humans are better at those functions that cannot be standardized. They are also better at decision-making that has not been adequately established (i.e., creative thought) and in co-ordinations that involve the integration of a great many factors whose subtleties or non-quantifiable attributes defy computer implementation (Lefkowitz, 1982). Thus, the computer offers a means to rapidly analyze process information in a systematic and predetermined manner. Automating such analysis in this fashion should make the information reaching the operators more meaningful, consequently reducing their cognitive load during process fault situations.

## **II. Automatic, Continuous Data Validation & Fault Analysis**

Humans have certain inherent limitations that cause their performance as process operators to always be potentially unreliable. One of these limitations is known as “vigilance decrement”, or the number of things that go unnoticed increases the longer a human performs a monitoring task (Eberts, 1985). Another limitation of human operators is a phenomenon called “mind set” (Kletz, 1985), which is also known as “cognitive lockup” (Sheridan, 1981), “tunnel vision” (Lees, 1983), and the “point of no return”

(Rasmussen, 1981). This means that sometimes when an operator becomes sufficiently certain as to the cause of abnormal process behavior, (s)he becomes exclusively committed to that particular hypothesis and acts upon it accordingly. A third human limitation is the phenomenon known as “cognitive overload”. That is, even when the detection of system failures is automated, the sheer number of alarms in the first few minutes of a major process failure may bewilder the process operators (Bailey, 1984, Fortin et al., 1983). A fourth limitation of human operators is that the situation confronting them may require knowledge that is either beyond their ability to understand (Goff, 1985), that is outside the knowledge they have gained from their experience and training, or that they have forgotten. This creates a somewhat paradoxical situation of the need for highly trained personnel to run “automated plants” (Visick, 1986). The final human limitation is that, even in the best situations, humans make errors. Despite efforts intended to reduce such errors, human errors can never be totally eliminated (Kletz, 1994).

Counteracting these limitations requires a means of for correctly analyzing the current process state and diligently and relentlessly monitoring process sensors that manipulates the controllers. Since the system performing this monitoring and analysis would always be aware of the actual process state, such a system would maximize the time available to the decision-maker when process operating problems arose.

Counteracting the above limitations also requires a means for examining all of the available evidence in a rational, unbiased manner. With such a system, all plausible fault hypotheses consistent with the evidence can be continuously derived, ranked, and updated according to how well they explained the observed process behavior. In addition, there needs to be a means for rapidly, rationally, and consistently analyzing the process state, regardless of how abnormal it is or how quickly it is changing, and focusing the operators’ attention on the most likely causes of the observed behavior.

Therefore, an innovative, automatic, continuous sensor data validation and fault analysis program was incorporated into the process control systems of two chemical plants. These knowledge-based systems (KBS) are referred to as **FALCON** for the DuPont adipic acid process and **FALCONEER** for the FMC persulfate process. **FALCON / FALCONEER** are acronyms for *F*ault *A*na*L*ysis *C*ONsultant via *E*ngineering *E*quation *R*esiduals. They are classified as knowledge base systems because they base their

conclusions on fundamental conservation principles and engineering concepts, such as mass and energy balance models. These models are evaluated with actual process data and, depending upon how well they close, directly provide evidence of the current operating state of the process system, and validate the information provided by the sensors for control.

Using models to perform data validation and fault analysis in real time proves the assertion that “Models are the means by which data can be converted to meaningful information” (Kramer & Mah, 1993). These models are based on a fundamental understanding of normal operating behavior of the given process system. They generate an unimpeachable source of information for logically inferring conclusions about the process being modeled. Automatically performing this inference after each update of process sensor data allows such KBS’s to perform “intelligent supervision” of the daily operations of their associated process systems.

The KBS system design first determines the current operating state of the operating process. This is essential in order for the KBS to perform the proper analysis on the collected process sensor data. The program is able to be turned on at any time and then automatically determine the current process state and monitor for all state transitions possible from that state. This helps minimize “nuisance alarms”. Examples of possible process operating states are:

- ✍ The process is being started up.
- ✍ The process is in production mode and is running within all Standard Operating Conditions (SOC’s) of its key control points.
- ✍ The process is in production mode but is not running within all SOC’s of its key control points or is rapidly approaching one or more interlock shut down regions of operation.
- ✍ The process is shutdown. (All other key raw material feeds have been shut off).

The type of programmed logic *FALCONEER* uses is called the Method of Minimal Evidence (MOME) (Fickelscherer, 1990, Fickelscherer, 1993). MOME is a diagnostic strategy based upon the behavior of engineering models describing normal operation of the plant process system. It has been demonstrated to

be competent in an Adipic Acid plant owned and operated by DuPont in Victoria, Texas (Dhurjati, et al. 1988, Rowan, 1992).

The chief advantage of the Method of Minimal Evidence (MOME) is that it provides a uniform framework for examining process models and their associated modeling assumption variables. In this framework, each process model represents the relationship that exists between modeling assumption variables during normal process operation. Process engineers provide the primary process models, which are all linearly independent of each other. The MOME methodology derives and adds additional, secondary process models from combining primary models, which improve the discrimination ability of the process model set. Any significant residuals resulting from the evaluation of these models directly indicate that one or more of the associated modeling assumption variables are deviating. All modeling assumption deviations can be classified into one of three possible categories. These three categories are distinguished from one another by their effect on the residuals of the constraints they violate. These classifications permit the derivation of standard diagnostic rule formats for each category, thus allowing the fault analyzer development to be very systematic.

Another major advantage of the diagnostic strategy centers around the diagnostic rule formats it uses. These formats have been chosen to ensure that the diagnostic knowledge bases will always perform competently; i.e., it will only make the correct fault diagnoses or not any diagnoses. These formats also ensure that diagnostic knowledge bases can diagnose the faults with the best possible diagnostic sensitivity and diagnostic resolution. Each diagnostic rule contains only the minimal amount of diagnostic evidence required to uniquely discriminate its associated fault situations from all of the other likely process operating events. Using just the minimal amount of diagnostic evidence in each diagnostic rule in this manner also allows many multiple fault situations to be diagnosed directly. The diagnostic strategy can also be used to determine the optimal sensor placement for performing process fault analysis. It can also be used to optimally distribute fault analyzers throughout a large process system.

The basic logic behind MOME is as follows. All unevaluated process models describing normal process operation can be characterized as functions of the following quantities:

$$O = f(i) \text{ (sensor variables, parameters, modeling assumptions, time)}$$

Once evaluated with actual process data and standard or extreme values of unmeasured parameters, a residual results (i.e.,  $\delta(i)$  below) which is just a function of process sensor noise and any currently occurring modeling assumption deviations, e.g.,

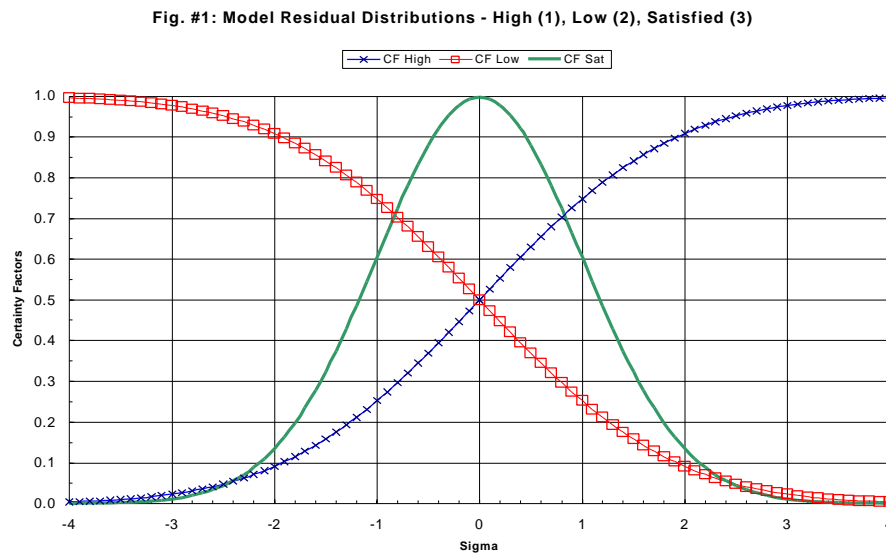
$$\delta(i) = f(i) \text{ (sensor noise, modeling assumption deviations)}$$

If the residual (i.e.,  $\delta(i)$ ) of an evaluated model is significantly high or low, then it can be inferred that at least one or more of the possible modeling assumptions' deviations could cause such a residual is occurring. If the residual (i.e.,  $\delta(i)$ ) of an evaluated model is not significant, then the following logic applies. Either there are no modeling assumption deviations, one or more such deviations are occurring but at magnitudes or rates of change that are below the sensitivity of that model to discriminate such deviations, or two or more significant assumption deviations are interacting in an opposing fashion.

The patterns of expected residual behavior that result from applying this method (e.g., the diagnostic rules) contain the minimum patterns required to diagnose each of the possible fault situations. This directly maximizes the sensitivity of the fault analyzer for these various faults, maximizes the resolution (discrimination between various possible faults and non-fault events) of that analysis, and optimizes its overall competence when confronted with multiple assumption deviations.

Rather than using Boolean Logic, this program uses Certainty Factors to make its decisions in order to combat a problem referred to as "diagnostic instability" (Kramer, 1987). This occurs when Boolean states toggle between true and false, creating a "chatty" situation. All Primary and Secondary Models' residuals (i.e.,  $\delta(i)$ ) are converted to three Certainty Factors for each model (low, satisfied, and high). The Certainty

Factors for high and low model states were originally calculated as a “one-tailed Z test with a confidence level of three sigma” where sigma is based on the variance calculated using the process models’ residual’s normal distributions. The Certainty Factor for the normal situation is determined from the normalized Gaussian distribution of those residuals (i.e.,  $\sigma$ ). The possible Certainty Factors associated with these functions are shown in Figure #1.



Perfect resolution between different process fault hypotheses is not always possible or is possible only at larger magnitudes of the specified fault. Trading lower diagnostic resolution for higher diagnostic sensitivity in this fashion allows the fault analyzer to narrow down the potential faults occurring into a reasonable number, which can then be checked out to determine the actual fault present. It is important to emphasize that because of the extensive use of OR Logic by this KBS, whenever more than one hypothesis is announced, it could actually be that one or the other or both is the actual fault occurring in the process. None the less, because there exists many cross checks between different data validation and diagnostic rules, as many such OR Logic situations as possible will be eliminated by the conclusions of the corresponding data validation rules. In other words, by the validation rules finding those assumptions satisfied and, since the validation rules use AND Logic, the satisfied hypothesis will always override all other high or low hypotheses as the best determination of the associated assumption’s current state.

In a nutshell, the MOME Strategy for fault analysis compares patterns of residual behavior expected to occur during the various possible assumption deviations with the patterns of those residuals currently present in the process. It uses the minimum unique patterns required for correctly doing this analysis, allowing for many of the possible multiple assumption deviations to be directly identified. This is important because this methodology does not discern only process operating faults, but all possible assumption deviations, fault and non-fault events alike. The logic is designed to venture diagnoses only if it is highly certain of the underlying problem. This is advantageous because it should not confuse its users with incorrect diagnoses at times when the actual process operating state is in flux.

### **III. Process Descriptions**

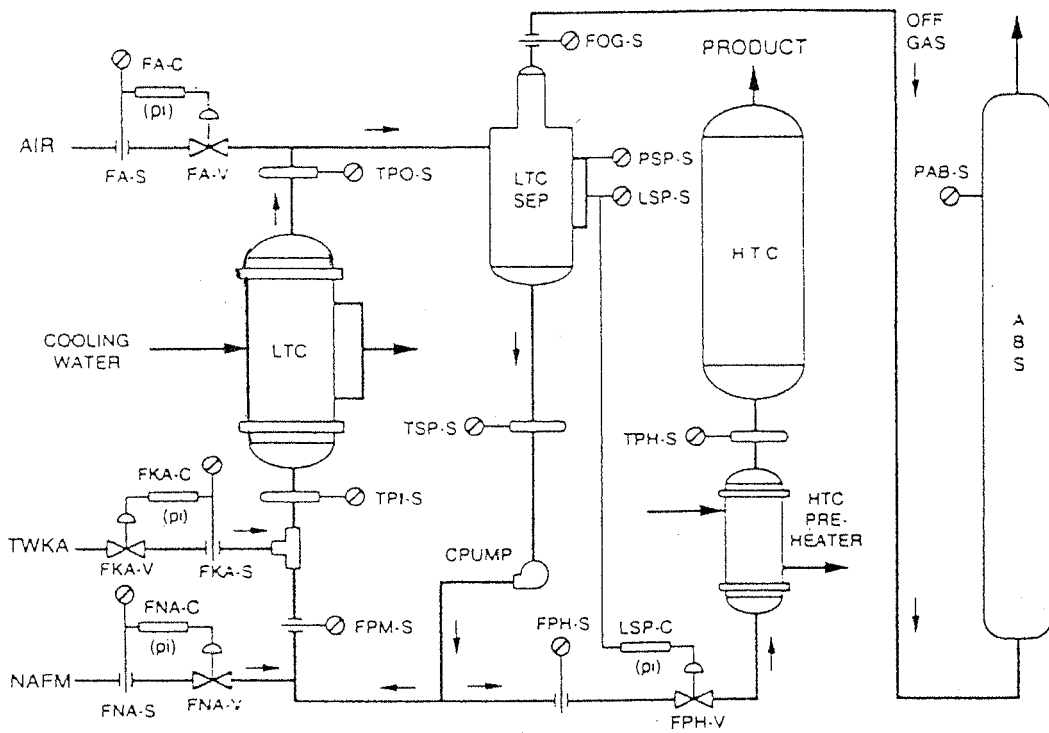
#### **III.A. Adipic Acid Process**

DuPont selected the adipic acid process as the target process system for the FALCON Project mainly because it was an established, highly developed, and widely known technology. Adipic acid is the monomer reactant in the polymerization of nylon-6,6. Since DuPont is firmly established as one of the dominant producers and users of adipic acid worldwide, the threat posed by proprietary information inadvertently being released as a result of the project was minimal. Moreover, serious process fault situations had previously occurred in these process systems. Consequently, developing an operable fault analyzer for this process system could potentially have some direct safety benefits. It was decided to focus the initial efforts on one particular process subsystem, the adipic acid process's Low Temperature Converter (LTC) Recycle Loop. This subsystem is the main reactor of the process and met the requirements for the target process.

A schematic of the process side of the LTC Recycle Loop is given in Figure 2. The main process unit in this system is the Low Temperature Converter itself (LTC). The LTC is a water cooled, plug flow reactor. Its main function is to remove the heat of reaction evolved from the oxidation via nitric acid (referred to as NAFM) of a mixture of cyclohexanone and cyclohexanol (referred to as TWKA) into adipic acid. The heat of reaction is removed from this highly exothermic reaction by cooling water flowing through the shell-side

of the LTC. This reaction also produces a relatively large quantity of process offgas. Air is injected into the process stream at the LTC's outlet to dilute the process offgas in order to reduce the chances that it can ignite. The resulting gas/liquid mixture is separated in a unit called the LTC Separator. The gas stream is sent to an absorber where the nitric oxides are recovered and recycled. The process liquid is pumped out of the separator, with the bulk of that liquid being recycled back to the LTC. The residual reactant in the product stream is oxidized in a unit known as the High Temperature Converter (HTC). The product stream is then sent to crystallizers where the adipic acid is removed.

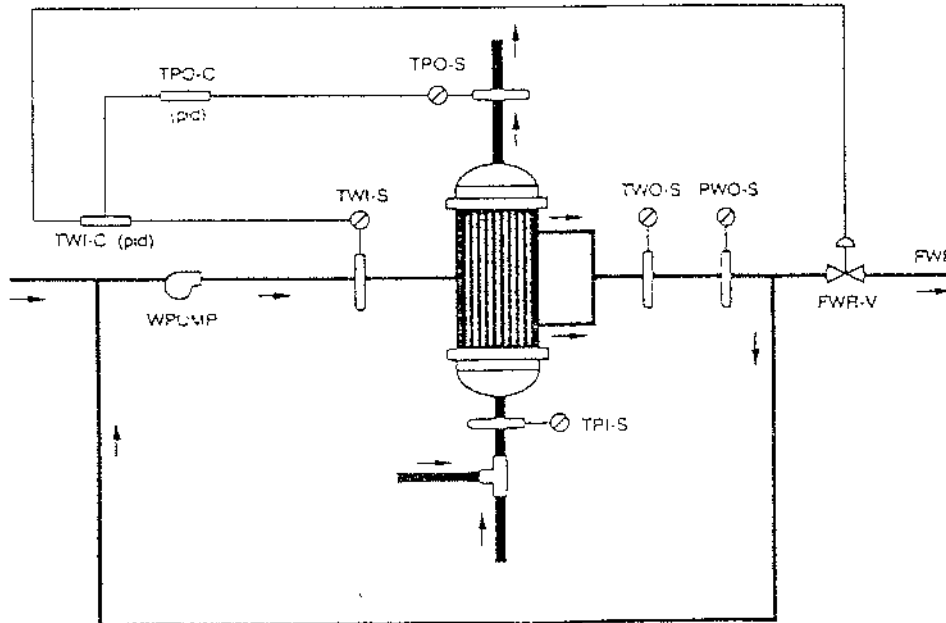
**FIGURE #2: ADIPIC ACID PROCESS LTC RECYCLE LOOP**



The normal operation of the LTC occurs within a very narrow range of process temperatures. If the process temperature gets too high, the process liquid will degas, causing the process pressure to increase to the point where it will rupture the process equipment. If the process temperature gets too low, the dissolved adipic acid will reach its saturation point and crystallize out of solution. When this occurs, the LTC's tubes

become plugged and only a partial oxidation takes place in LTC. This operating problem is known as "**frosting.**" The remaining reactants are converted in the downstream units not designed to remove the heat of reaction. This situation could also cause the process liquid to degas and consequently rupture the process equipment. To minimize the chances of such faults, the process temperature is maintained within the narrow range of temperatures with an elaborate temperature control system. A schematic of the LTC's cooling system is given in Figure 3.

**FIGURE #3: LTC COOLING SYSTEM RECYCLE LOOP**

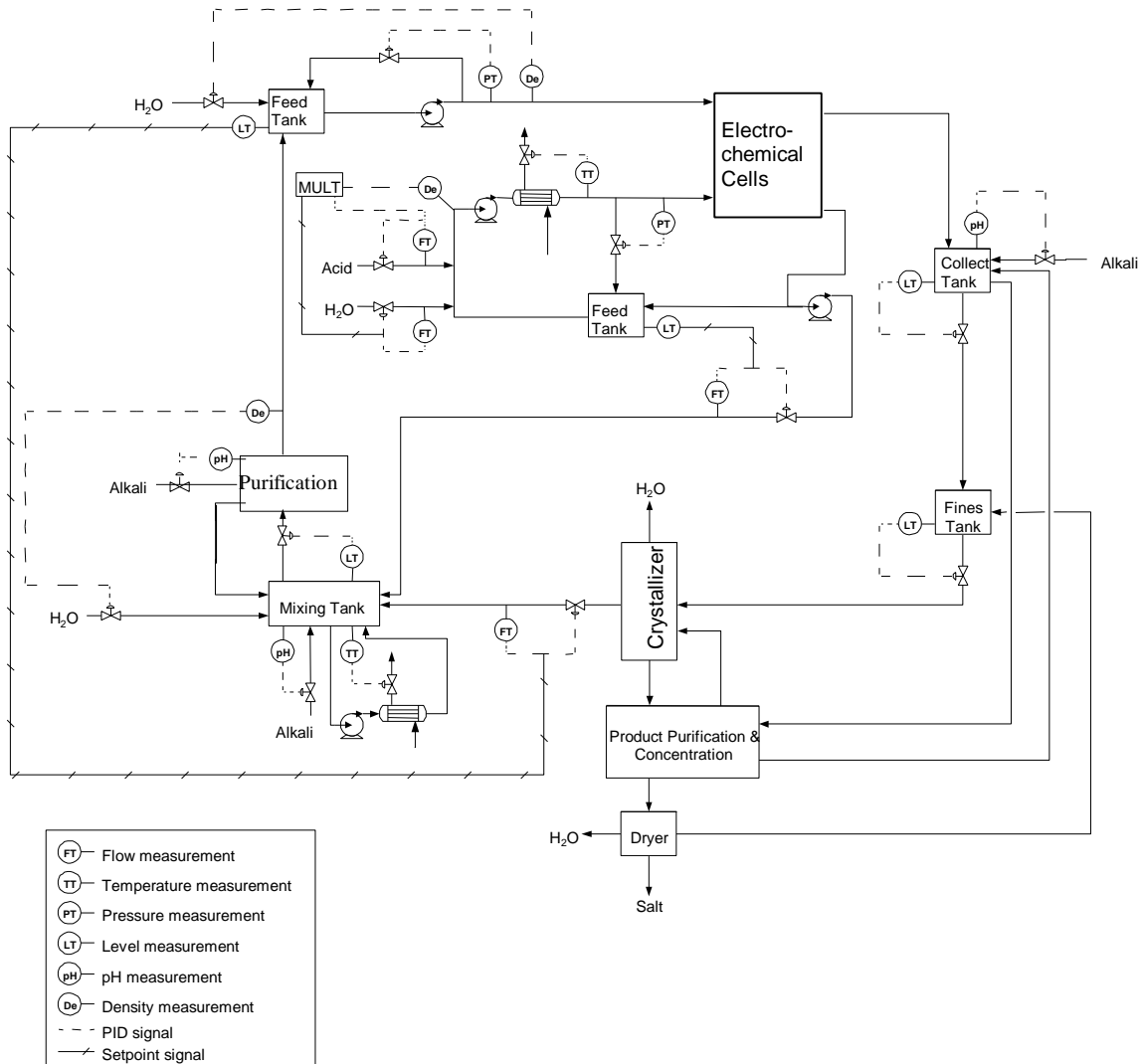


### III.B. Sodium Persulfate Process

FMC's electrolytic process contains water and 4 inorganic salts, which in solution dissociates into 7 ions. The main unit operations in the plant are the electrolytic cells and the crystallizer. In addition a series of tanks (Anolyte Feed, Anolyte Collect & Mixing tank) are used to maintain the correct chemistry. On the anodic side of the electrolytic cells, anions react to form the anions needed in the final salt while the raw material is added to the cathodic side, which is transferred to the anodic side through the membrane and through the Mixing tank. In the crystallizer, water evaporation occurs under vacuum, which saturates the

solution and precipitates the product salt. Salts formed in the crystallizer are treated in a special system that ensures the formation of a pure final salt. A simplified layout of the process is given in Figure #4.

**FIGURE 4: SIMPLIFIED PERSULFATE PROCESS WITH CONTROL STRUCTURE**



A Honeywell TDC 3000 is the distributed control system that runs the electrochemical plant. The critical control loops highlighted in Figure #4 are described briefly below.

✍ *Mixing tank control loop:* Prepares the solution that is sent to the cells by controlling pH and density and by specifying a catholyte forward flow.

✍ *Anolyte Feed tank control loop:* Keeps the feed flow rate to the cells constant to maintain required electrochemical cell operation.

- ✍ *Crystallizer control loop*: Maintains constant evaporation and extraction rates in order to precipitate and grow the salt crystals.
- ✍ *Electrolyte cell control loop*: Constant amperage is maintained to achieve a constant production rate.
- ✍ *Product purity control loop*: Determines the purity specification of the final product/salt.

#### IV. FALCON and FALCONEER KBS Comparisons & Performance Results

The **FALCON** System was tested during its development with 260 simulated fault situations and 500 hours of selected process data which contained 65 process operating events (e.g., emergency shutdowns, startups, production changes, etc.), including 13 actual process fault situations. During its off-line field test, the **FALCON** System monitored over 5,000 continuous hours of process data in real time. This data included 22 process operating events, 8 of which were actual process fault situations. The **FALCON** System was then tested on-line for three months. DuPont independently rated its performance at better than 95% correct responses during that test (Rowan, 1992).

Although it preformed competently on-line, maintaining and improving **FALCON**'s diagnostic knowledge base (written in Common Lisp based data structures) proved to be impractical for anyone other than the original developer (i.e., the University of Delaware) (Rowan & Taylor, 1989). Being a research project, maintainability was not given as high a priority as was **FALCON**'s performance with actual process data. From a research viewpoint, generalizing the underlying logic of this model-based diagnostic strategy was paramount, allowing future such development project activities to be as streamlined as possible. This effort led directly to the formulation of the Method of Minimal Evidence (**MOME**) (Fickelscherer, 1990).

The **FALCONEER** system developed for FMC's complex process system incorporated ~100 sensor measurement inputs and ~110 potential determinable fault hypotheses. It performed process state identification before applying the data validation and fault analysis rules. **FALCONEER** used FUZZY Logic / certainty factor calculations to attach credence to its conclusions, and used diagnostic rules anticipating all levels of diagnostic resolution possible. At this time, it does not use filtered residuals in its calculations, and does not perform interlock failure analysis.

The biggest benefit of the original **FALCON** Project was the derivation of the Method of Minimal Evidence. It allows automated process fault analyzers to be developed systematically and eliminates the need for exhaustive testing to demonstrate that the resulting KBS's are competent. The **FALCON** and **FALCONEER** Systems were developed for process systems of comparable complexity and the same order of magnitude number of both sensor measurement inputs (~30 for **FALCON**; ~100 for **FALCONEER**) and potential fault hypotheses determinable (~160 for **FALCON**, ~60 of which can be announced; ~110 for **FALCONEER**, all of which can be announced). Both also perform process state identification before applying the data validation and fault analysis rules.

They were also different in some respects: **FALCON** used Boolean logic exclusively, used both raw and filtered calculations of model residuals in its diagnostic rules, did interlock failure analysis, and only used the highest level of diagnostic resolution for announcing its conclusions; **FALCONEER** uses Certainty Factor calculations to attach credence to its conclusions, uses diagnostic rules anticipating all levels of diagnostic resolution possible, does not currently use filtered residuals in its calculations, and does not as of yet perform interlock failure analysis.

Respectively, **FALCON** contained approximately 850 production rules (~ 250 base propositions and 600 fault diagnostic rules) requiring ~10,000 lines of Common LISP code to implement while **FALCONEER** consists of approximately 1,050 data validation and fault analysis rules requiring ~18,000 lines of Visual Basic code to implement: approximately the same size and complexity all things being considered. However, the development effort required for the original **FALCON** System was much more than an order of magnitude greater than that required for **FALCONEER** (>15 man-years for **FALCON** versus 9 man-months for the enhanced **FALCONEER**). The **FALCON** Project experience allowed us to stream-line the development activities and eliminate the nearly exhaustive testing that DuPont required before the **FALCON** System was put on-line.

The development of FMC's *FALCONEER* System (Skotte et al., 2001) proved to be a systematic application of MOME to their Electrolytic Sodium Persulfate process. The resulting quality of program code is as optimal as possible for its method of implementation (i.e., hand compile the DV&FA rules based on the expected behavior of the Primary and Secondary Models and then code them manually).

Developing the engineering models, which describe normal process operation, evaluating them with sufficient process data to determine their normal variances and off-sets, and then hand compiling the DV&FA diagnostic rules consumed the vast majority of the development effort. The development and implementation of *FALCONEER* even provided initial dividends. These dividends included: (1) several process sensors routinely used to control the process were found to be in error; (2) a higher than expected unmeasured flow in one unit operation was discovered by evaluating the primary models with actual process sensor data; and (3) some of these model residuals also provided supporting evidence for expected, but unmeasured, process phenomenon such as fines destruction and crystallizer boiling point elevation.

*FALCONEER* has now been continuously operating on-line at FMC's Tonawanda New York plant since mid-February 2001. To date, *FALCONEER* has competently monitored its associated process for possible process fault situations as a diligent and relentless watchdog.

## **V. Criteria for Optimal Process Sensor Placement**

Currently, process plants are instrumented so that their operation can be more easily monitored and controlled. Normally, only little thought is given about the impact this has upon the operator's ability to effectively perform process fault management. Systematic approaches are not normally used to analyze proposed configurations of process instrumentation for performing optimal fault analysis. In the past, poorly designed configurations have been a contributing cause of many major process accidents (Kletz, 1985), including the near meltdown of the nuclear reactor at the Three Mile Island power plant (Lees, 1983).

Systematic approaches to process fault analysis, such as the Method of Minimal Evidence, can be used directly to analyze the impact of process instrumentation upon the ability to effectively perform fault analysis (Raghuraj et al., 1999). In fact, such analyses were used twice during the development of the

**FALCON** KBS to examine the instrumentation of the adipic acid system. The first analysis performed became the basis for a recommendation that DuPont collect 15 additional process measurements (7 pairs of controller outputs and setpoints, 1 thermocouple reading). This recommendation was accepted. The second analysis was performed to anticipate the effects that an impending process system modification would have upon the **FALCON** System's performance. This formed the basis of a recommendation that DuPont leave in place a pressure sensor that it had originally planned to remove. This recommendation was also accepted. It was also used in the development of the *FALCONEER* KBS to successfully recommend to FMC that two additional flow meters be added to their Electrolytic Sodium Persulfate System.

In general, process instrumentation should be added only if it will appreciably improve the fault analyzer's diagnostic sensitivity or its diagnostic resolution for a particular fault situation. Moreover, since adding and maintaining instrumentation is expensive, it must be justified by improvements in process safety. Fault situations that have potentially severe consequences would most likely qualify as those that should have all the instrumentation necessary for performing effective fault analysis.

## **VI. Data Validation & Fault Analysis Diagnostic Rule Compiler Motivations**

A critical step in the development of a fault analyzer is verifying that the underlying diagnostic knowledge base performs competently. Such verification ensures that the diagnostic rules will always perform correctly. During the development of the **FALCON** System, this was accomplished by nearly exhaustively testing the fault analyzer with both actual and simulated fault situations. Thoroughly doing this turned out to be the most computationally intensive undertaking of the **FALCON** Project, with more than 5,500 hours of actual plant data and about 260 simulated process fault situations being analyzed before the **FALCON** System was installed on-line in the plant. The various phases of this verification effort required a period of approximately three years to complete.

Obviously, if such extensive verification efforts were a prerequisite for developing competent fault analyzers in general, these programs would never be widely used within the processing industries. Fortunately though, efforts similar to that expended in verifying the **FALCON** System are not required.

It is possible to dramatically reduce the effort required to verify a process fault analyzer based upon the Method of Minimal Evidence (MOME). If all of the process models contained within the declarative knowledge base are well-formulated, then any subsequent misdiagnoses made by the fault analyzer directly indicates that one or more of the diagnostic rules is incorrect. However, since MOME is a very systematic and logically structured procedure for creating diagnostic rules, it has been possible to develop an Artificial Intelligence (AI) driven compiler program, which automatically creates diagnostic knowledge bases based upon it. Such a program ensures that MOME is always applied correctly.

With this AI-driven compiler program, if all of the process models are indeed well-formulated then the resulting fault analyzer is guaranteed to perform competently. Consequently, verifying the correctness of various diagnostic rules would not be necessary: only the correctness of the process models used to create those rules needs to be verified. In the case of the **FALCON** System, this would mean the difference between verifying over 10,000 lines of Common Lisp code (representing the 800+ highly interdependent diagnostic rules and 50+ primary and secondary models) and verifying about 200 lines of FORTRAN code (representing just the 50+ primary and secondary models). In **FALCONEER**, only 30 Primary and 70 Secondary models had to be verified rather than the additional 1,050 DV&FA diagnostic rules. This allowed the **FALCONEER** KBS to be delivered with 1 person year of effort (even though we still hand-compiled the diagnostic rules) rather than the 15 person years required for the original **FALCON** KBS development effort.

Consequently, process engineers would only have to maintain the declarative knowledge base containing the various process models. This is advantageous for the following reason. As opposed to other knowledge representation schemes (such as production rules, frames, etc.), process models can be

represented as mathematical equations. Furthermore, since all of the primary process models within the diagnostic knowledge base are independent of each other, each can be independently added, modified, or removed as required. This would simplify the maintenance of the process fault analyzer immensely. With an AI-driven compiler, the fault analyzer can be incrementally improved with minimal effort as the process system's operating behavior became better understood.

Having such an AI-driven compiler will greatly improve the resulting knowledge based systems' transparency<sup>1</sup> by separating the domain specific knowledge (i.e., the various primary and secondary models of normal operation) from its general problem solving strategy (i.e., the MOME algorithm and exhaustive search of the resulting Data Validation and Fault Analysis (DV&FA) diagnostic rules).

## **VII. Conclusions & Recommendations**

### **VII.A Case Study Conclusions**

Using models to perform innovative data validation and fault analysis in real time proves the mantra that, “Models are the means by which data can be converted to meaningful information” (Kramer & Mah, 1993). Based upon the most fundamental understanding of both DuPont’s and FMC’s plant process systems available, these models coupled with a systematic logical evaluation represent an unimpeachable source of derived information that can provide an early warning for safety, quality, and/or operability related sensor problems or process leaks. This tool also provides background, diligent monitoring of key instrumented process variables to assist the control room operators in process monitoring, troubleshooting, and control decision making. Having this systematic logical inference done automatically after each update of process sensor data will allow these MOME knowledge-based systems to become an indispensable tool for performing “intelligent supervision” of the daily operations of the operating plant process system.

---

<sup>1</sup> Transparency is defined as the KBS's understandability to both that system's developer and targeted user.

## **VII.B. Recommendations**

Even the impressive improvement noted in the required development effort for *FALCONEER* is still an order of magnitude higher in time requirements than those truly necessary. We anticipate that this development time will be remarkably shortened by our compiler program that generates the data validation and fault analysis (DV&FA) rules based on MOME directly from the underlying models of normal process operation. Automating the creation of these DV&FA rules according to the MOME algorithm is straightforward. All of the development and maintenance effort to create such fault analyzers will then only be directed at the derivation of models of normal process operation (the declarative knowledge) in the KBS. This constitutes the data structure input into the compiler. Since all the primary models are linearly independent, they can be added, improved or deleted as need be and then the entire KBS knowledge base recompiled to include these improvements. Consequently, with such a compiler, the fault analyzer can be incrementally improved with minimal effort as the process system's operating behavior becomes better understood or the process system changes, allowing the DV&FA KBS to easily evolve along with its associated process system. This in turn will substantially reduce both the development and maintenance costs of such fault analyzers.

## **VIII. ACKNOWLEDGEMENTS**

The authors would like to thank FMC Corporation and their employees and Dupont Corporation and their employees for their support throughout these projects. We especially would like to thank the operators and manufacturing engineers who were / are the customers & direct end-users for their patience, advice, and assistance in the development, testing, and implementation of these knowledge based systems. We would also like to thank Profs. Dhurjati & Lamb of the University of Delaware and all their graduate students who worked on **FALCON** and the Foxboro Company for their financial support in this effort.

## REFERENCES

- Bailey, S. J., "From Desktop to Plant Floor, a CRT is the Control Operators Window on the Process", *Control Engineering*, 1984
- De Heer, L.E., "Plant Scale Process Monitoring and Control Systems: Eighteen Years and Counting", Proceedings of the 1<sup>st</sup> International Conference on Foundations of Computer-Aided Process Operations, ed. by G.V. Reklaitis and H.D. Spriggs, New York, Elsevier Science Publishers Inc., 1987, pp. 33-66.
- Dhurjati, P.S., D.E. Lamb and D.L. Chester, "Experience in the Development of an Expert System for Fault Diagnosis in a Commercial Scale Chemical Process", Proceedings of the 1<sup>st</sup> International Conference on Foundations of Computer-Aided Process Operations, ed. by G.V. Reklaitis and H.D. Spriggs, New York, Elsevier Science Publishers Inc., 1988, pp. 589-617.
- Dhurjati, P.S., ed., "On-Line Fault Detection and Supervision in the Chemical Process Industries," Proceedings of the International Federation of Automatic Control Symposium, Newark, Delaware, USA, 1992.
- Eberts, R.E., "Cognitive Skills and Process Control", *CEP*, December 1985, pp. 30-34.
- Fickelscherer, R. J., "Automated Process Fault Analysis," Ph.D. Thesis, University of Delaware, Newark, DE, 1990.
- Fickelscherer, R.J., "A Generalized Approach to Model-based Process Fault Analysis", Proceedings of the 2<sup>nd</sup> International Conference on Foundations of Computer-Aided Process Operations, ed. by D.W.T. Rippin, J.C. Hale, J.F. Davis, Austin, TX, CACHE, 1993, pp. 451-456.
- Fortin, D. A., T. B. Rooney, and H. Bristol, "Of Christmas Trees and Sweetie Palms", Proceedings of the Ninth Annual Advanced Control Conference, West Lafayette, Indiana, 1983, pp. 49-54.
- Goff, K.W., "Artificial Intelligence in Process Control", *Mechanical Engineering*, October, 1985, pp. 53-57.
- IFAC 1995, "IFAC Workshop on On-Line Fault Detection and Supervision in the Chemical and Process Industries," Proceedings of the International Federation of Automatic Control Symposium, Seattle, Washington, USA, 1995.
- Kletz, T.A., "What Went Wrong? Case Histories of Process Plant Disasters", Houston, TX, Gulf Publishing Co., 1985.
- Kletz, T. A., "Living with Human Errors on Computer Controlled Plants," Foundations of Computer-Aided Process Operations II, ed. by D. W .T. Rippin, J. C. Hale and J. F. Davis, Austin, TX, CACHE, Inc., 1994, pp. 95-107.
- Kramer, M.A., "Malfunction Diagnosis Using Quantitative Models and Non-Boolean Reasoning in Expert Systems," AIChE Journal 33, 1987, pp.130 – 147.
- Kramer, M. A. and R. S. H. Mah, "Model-Based Monitoring", Foundations of Computer-Aided Process Operations II, ed. by D. W .T. Rippin, J. C. Hale and J. F. Davis, Austin, TX, CACHE, Inc., 1993, pp. 45-68.
- Lees, F.P., "Process Computer Alarm and Disturbance Analysis: Review of the State of the Art", *Comp. And Chem. Eng.*, Vol. 7, No. 6, 1983, pp. 669-694.

Lefkowitz, I., "Hierarchical Control in Large Scale Industrial Systems", Studies in Management Science and Systems, Vol. 7, New York, North-Holland Publishing Co., 1982, pp. 65-98.

Lieberman, N.P., "Troubleshooting Process Operations", Tulsa, OK, PennWell Publishing Co., 1985.

Raghuraj, R., M. Bhushan, and R. Rengaswamy, "Locating Sensors in Complex Chemical Plants Based on Fault Diagnostic Observability Criteria", AIChE Journal, Vol. 45, No. 2, (1999), pp. 310 - 322.

Rasmussen J., "Models of Mental Strategies in Process Plant Diagnosis", Human Detection and Diagnosis of System Failures, ed. by J. Rasmussen and W.B. Rouse, New York, Plenum, 1981, pp. 251-258.

Rowan, D.A., "Beyond FALCON: Industrial Applications of Knowledge-Based Systems", IFAC Symposium – On-line Fault Detection and Supervision in the Chemical Process Industries, ed. by P.S. Dhurjati, Newark, DE, 1992, pp. 215-217.

Rowan, D. A., Ibid., 1992.

Rowan, D. A. and R. J. Taylor, "On-Line Fault Diagnosis: *FALCON* Project," Artificial Intelligence Handbook, Instrument Society of America 1989, Vol. 2, pp. 379-399.

Sheridan, T.B., "Understanding Human Error and Aiding Human Diagnostic Behavior in Nuclear Power Plants", Human Detection and Diagnosis of System Failures, ed. by J. Rasmussen and W.B. Rouse, New York, Plenum, 1981, pp. 19-35.

Skotte, R., D. Lenz, R. Fickelscherer, W. An, D. Lapham III, C. Lymburner, J. Kaylor, D. Baptiste, M. Pinsky, F. Gani and S. B. Jørgensen, "Advanced Process Control with Innovation for an Integrated Electrochemical Process," AIChE Spring National Meeting, Houston, TX, April, 2001.

Venkatasubramanian, V., "A Review of Process Fault Detection and Diagnosis: Past, Present and Future" submitted to Computers and Chemical Engineering, 2001

Visick, D., "Human Operators and their Role in Automated Plant", Chemistry and Industry, May 1986, pp. 199-203.