AGENT-BASED MONITORING, FAULT DETECTION, DIAGNOSIS AND CONTROL OF SPATIALLY DISTRIBUTED PROCESSES

Sinem Perk* and Ali Çınar*

* *Illinois Institute of Technology, Department of Chemical and Environmental Engineering, 10 West 33rd Street, Chicago, IL, 60616, USA*

Abstract: There is a trend in the process industries towards the use of multiple small reconfigurable production units rather than a single large unit to increase production, product range and operation flexibility. Monitoring, supervision and control of these processes is difficult because traditional methods typically do not handle well the increased heterogeneity, nonlinearity and complexity. In this work, statistical multiblock process monitoring techniques and an agent-based diagnosis and control methodology are integrated. The performance of this combined monitoring-diagnosis-control system is illustrated with a case study using a network of spatially distributed reactors.

Keywords: Multiblock Monitoring, Fault Detection and Diagnosis, Agent-based Simulation, Optimization

1. INTRODUCTION

As networked applications in process industries have increased in the recent years, existing methods for monitoring, supervision and control of the processes need to be improved or supplemented with new techniques in order to handle the complexity, heterogeneity and nonlinearity of these networked systems. In this study, a monitoring-diagnosis-control sequence that uses existing tools from literature and supplements those with agent-based decision-making is proposed. The agent-based approach is recommended for use especially in networked systems since it is a powerful tool for the management of distributed systems. Although agent-based modeling has found wide applications in social sciences and it has been proposed for a few control applications in chemical engineering, its application to fault detection and diagnosis has not been reported.

Having a single model for all different phases or stages of a system with different covariance structures may not give a sufficient explanation of the system behavior and fault detection and diagnosis can be more challenging with increasing model size. Multiblock methods have been recently proposed to improve the capabilities of the existing statistical process monitoring techniques. A comparison of the popular multiblock methods is provided (Smilde *et al.*, 2003). These algorithms have been applied to monitoring, fault detection and diagnosis of continuous processes and definitions for control limits for multiblock algorithms have been developed (Qin *et al.*, 2001; MacGregor *et al.*, 1994).

In a previous work, the multiway multiblock principal components analysis (PCA) has been applied to a penicillin fermentation process, where different phases of growth were modeled as separate blocks, to reflect the different covariance structures between phases in the model and to localize the fault in the batch process (Perk and Çınar, 2006). Since a network of interconnected continuous stirred tank reactors (CSTRs) is being considered, multiblock monitoring is employed in this study.

The control problem considered is the restoration of normal process operation after a disturbance hits the reactor network. Several disturbances such as a step change in feed flow rate, a sensor fault on a concentration reading, or the introduction and propagation of a different species that competes for resource utilization, in one of the reactors or a part of the network are introduced to test the effectiveness of the multiblock monitoring methods in detection and diagnosis and the control structure in restoring the system to its expected behavior. In this paper, the effect of a change in the feed properties to one of the reactors is reported.

Although the mathematical tools used in this study have been in use for over a decade, they have not been integrated under the roof of agent-based system before. Agent-based decision-making capability reduces the need for human intervention to the system and automates the monitoring, fault detection, diagnosis and control sequence.

In the following sections, multiblock PCA, agent-based fault detection and diagnosis, optimization and control, the autocatalytic CSTR network and the software framework are discussed. And at the end, a case study is presented.

2. MULTIBLOCK PRINCIPAL COMPONENTS ANALYSIS

Traditionally, PCA is used to form the statistical model based on the covariance structure of the normal operating data and the new observations are tested against this model. Monitoring large-scale distributed systems is difficult since many different structures constitute the data and treating the data as if it is coming from a single structure and trying to come up with a single statistical model that perfectly explains the system behavior is usually not possible. Even if the single model is powerful enough to detect any variation from in-control observations, the detection of fault may be delayed.

The benefit of multiblock algorithms in localizing and isolating the fault is better experienced in processes involving many processing units with many process variables. With multiblock methods, the overall process and also each different unit or subsections of a unit can be monitored. This enables the isolation of the processing unit in which the deviation occurred and detection of the major contributors to the event.

The consensus PCA algorithm for multiple blocks, based on a series of NIPALS steps is given in (Westerhuis *et al.*, 1998). The method was introduced to compare several blocks of variables measured on the same objects. The data are divided into B blocks. A column of one of the blocks is selected as a starting super score and this vector is regressed on all block data to find the block loadings, from which the block scores for all blocks are calculated. All block scores are augmented in a super block. The super score is then regressed on the super block to give the super weight. The super weight is normalized and used to calculate a new super score vector. If this new super score converges to a predefined criteria, the iteration stops. Then, each block is deflated using the super score and the procedure repeats for the next principal component dimension. Otherwise, the iteration continues until the super score vector converges. For monitoring purposes, the statistics can be calculated for both the super level and for lower block level.

For the real-time simulation of a highly nonlinear heterogeneous system, where a small disturbance may have instant drastic effects over the system behavior, timely detection of faults is crucial. Moreover, detecting the parts of the system that were most affected and the variables that contribute to the fault the most is equally important. Multiblock monitoring based on consensus PCA algorithm is used in the statistical model formation where each CSTR in the network is represented as a separate block in the model.

The multiblock PCA method was especially suitable for use since the system is a network of spatially distributed reactors where each reactor is dominated by different species and monitoring of each reactor is of special interest. The block data consists of concentrations of species in the reactor and the resource concentration. Each block has the same dimensions of I observations and J variables. Multiblock monitoring enables the monitoring of each reactor as well as the monitoring of the system at the super level.

Considering CSTRs in the network as different entities with different structures helps to build a more reliable, realistic model, prevents loss of necessary information during the dimension reduction stage of single PCA model and more importantly, reduces the challenge of identifying the parts of the system that were affected the most by localizing the fault.

3. FAULT DETECTION AND AGENT-BASED DIAGNOSIS

The statistics used for monitoring multivariate continuous processes are the statistical distance

T^2 and the squared prediction error (SPE). T^2 shows the variation between the new observation and the previous observations and SPE reveals if the new observation can still be explained by the statistical model. If a new observation goes out of the confidence limits, it is flagged as an out-of-control observation. Persistent out-of-control signals show a fault has occurred in the system (Nomikos and MacGregor, 1995).

These two statistics are complementary to each other and must be used together. Sometimes only the T^2 chart signals out-of-control showing an observation is different than the others, however, SPE may not signal if the variation can still be explained by the model. Most of the time, a fault is signalled in both charts.

The above statistics detect a fault, but give no information about the cause of the fault. Contribution plots are used to show the contribution of each variable to the statistic calculated. After fault is detected, the variables that were significantly affected from the fault or significantly contributed to the fault can be identified in the contribution plots. In (Westerhuis *et al.*, 2005), the control limits for the contribution plots were introduced. A variable may have higher contribution than others at every step but this does not always mean it is the cause of the fault. The limits are calculated using in-control data, so if a variable's contribution is exceeding the limits, then it is accepted as contributing to the fault.

The false alarms and missed alarms associated with the above statistical techniques is typically a major problem in fault detection and diagnosis. Having too many false alarms and missed alarms decrease the credibility of the statistical technique used. Decision becomes difficult when different methods start alarming at different times for different units.

In the proposed structure, there are four agent types. Whenever a statistic goes outside of the confidence limits, each corresponding agent sends an out-of-control signal. For this continuous process, whenever a shift occurs in SPE or T^2 to move outside the limits, this shift is detected by the corresponding two types of agents using a cumulative sum (CUSUM) based change detection algorithm (Gustafsson, 2002). Multiblock PCA enables the monitoring of each individual block and block contributions to fault as well. Contribution limits are calculated for each block. The third type of agent watches the block contributions and signals whenever a block contribution is outside the block confidence limits. The fourth type of agent monitors the SPE and alarms when the SPE statistic exceeds the confidence limits. These agents compete for the same task of fault detection, are rewarded for correct and early detection and are penalized for false and missed alarms and are employed in the simulation based on their reliability. The messages sent by the agents reveal the faulty unit, and the time of fault.

Simulations are run for multiple times with known faults offline to test the detection and diagnosis algorithms. The location of fault and time of introduction is known, and agent-based fault detection is tested on these known faults. An agent earns a full positive reward if it correctly identifies the fault at the exact time of occurrence. If there is a lag between the time of diagnosis and the actual fault time, the agent receives a reduced reward, proportional to the duration of the lag. If the agent completely misses a known fault or gives a false alarm, a negative reward is deducted from its total rewards. In this way, the methods are ranked based on their performance in catching a fault. The method with a higher reward is trusted more. During the real-time simulation whenever an out-of-control signal is issued, the issuing agent's total reward is compared to the opposing agents' rewards and a fault is declared at the time of agreement.

4. CONTROL AND OPTIMIZATION

After having a consensus on the occurrence of a fault, its time of occurrence, and its location in the network by the diagnostic agents, that part of the network should be isolated in order to prevent the spread of the fault to the unaffected parts of the network, and the normal operation should be restored.

The problem becomes an agent-based optimum control problem, where the difference between the current states and the normal operation states is minimized by agents via adjusting the flow between reactors, giving minimum disturbance to the other parts of the network. As a whole, the diagnostic agents report if there is a fault in the system and its approximate time of occurrance and the control agents take action to reestablish the normal operation throughout the network.

When a fault is flagged, an offline simulator, which is a clone of the main simulator, is generated and it is given the current states of the system as a starting point. This clone is used by the optimization routine to simulate the process behavior with changing interaction flowrates *and/or* the feed flowrate. After the flowrates that satisfy the objective are found, they are set in the main simulator to continue the simulation with these updated values.

All improving search methods in optimization, starting with a feasible initial solution, consider neighbors of current solution and try to advance to one that is feasible and superior in objective value. If no feasible neigbor is improving, the process stops with local optimum. In order to produce more robust algorithms, the improving solution search idea can be extended to allow nonimproving feasible moves to escape a local optimum. One method of introducing nonimproving moves into improving search is termed simulated annealing, referring to the analogy to the annealing process of slowly cooling metals to improve strength. Simulated annealing algorithms control infinite cycling to a solution recently visited by accepting nonimproving moves according to probabilities tested with computer-generated random numbers. Improving and accepted nonimproving moves are pursued; rejected ones are not. The 'temperature' parameter in simulated annealing is used to control the randomness of the search. The searches usually begin with a relatively high temperature parameter and decrease it after every few iterations. As the temperature gets lower, the probability of accepting bad moves decreases dramatically (Rardin, 1998). In this work, simulated annealing is used as the optimization routine. The execution time associated with the search is negligibly small.

Rather than searching for the flowrates that would restore the system defaults, a historical database, where the states of the system were stored, could have been formed. After the detection of the fault the states could have been set back to those in the database. Since the system is highly nonlinear with multiple steady states, a very small disturbance can force the system to a totally different regime, making the historical database useless. Because of the system characteristics, it may be impossible to go back to the reference state. This brings in the need for a search algorithm to identify the search surface, attainable regions and to determine how close we can get to our reference state.

5. AUTOCATALYTIC CSTR NETWORK

In this study, the system of interest is a CSTR network. The static and dynamic behavior of an autocatalytic reaction with decay was studied in networks of coupled CSTRs (Tatara *et al.*, 2004). It was shown that the number of steady states of the network increases with heterogeneity, thereby allowing those autocatalytic species to exist in the network that would normally not exist in the homogeneous environment of a single CSTR. The heterogeneity of the networks is influenced by the number of reactors as well as the network topology. The existence of multiple steady states and the nonlinearity of the system provides a challenging supervision and control problem. Tatara *et al.* (2005) proposed an agent-based control algorithm to this challenging problem, whose work is extended in this paper by applying additional agent layers for process monitoring and fault detection.

The network consists of interconnected CSTRs, each having feed and exit streams, as well as multiple connections to their neighbors. Each reactor has as many inlet as outlet interconnection streams that are always equal. The feed flowrates and the interconnection flowrates can be treated as manipulated variables. In the case study presented later, the feed flowrate is used as the manipulated variable.

6. THE SOFTWARE FRAMEWORK

The framework is built in Java using RePast toolkit and COLT distribution. The reactor network model and agent-based control system is implemented with the open source agent modeling and simulation environment RePast (Collier *et al.*, 2003). The RePast toolkit is a java-based framework for agent simulation and provides features such as an event scheduler and visualization tools. The control agents created with RePast interact with virtual representations of the physical reactor network. The virtual network objects map the states of the physical system to objects that can be manipulated by the control objects. The ordinary differential equations that describe the autocatalytic reactions in each CSTR are solved numerically using the CVODE solver. The solver code is written in C and linked with RePast via the JavaNative Interface (JNI).

COLT distribution is used to implement the statistical model building and statistical testing methods and is connected to the agent simulation for online monitoring using RePast event scheduler. The COLT distribution consists of several free Java libraries, for user convenience bundled up under a single name and provides an infrastructure for scalable scientific and technical computing in Java. Its powerful matrix operations enables technical computing with Java.

7. CASE STUDY: CHANGE IN THE FEED

In the case study, the effect of a change in the feed to one of the reactors is demonstrated. Multiblock PCA model is formed with two principal components and new data projection starts at time ($t = 2200$) (Figure 1). At time ($t = 2500$), the feed to the fifth reactor, which was pure resource before, is contaminated with a trace amount (0.06) of the dominant species in the reactor. The fifth reactor is the center reactor in a 9-CSTR network as shown in Figure 2, with three autocatalytic species. The dimensionless interconnection

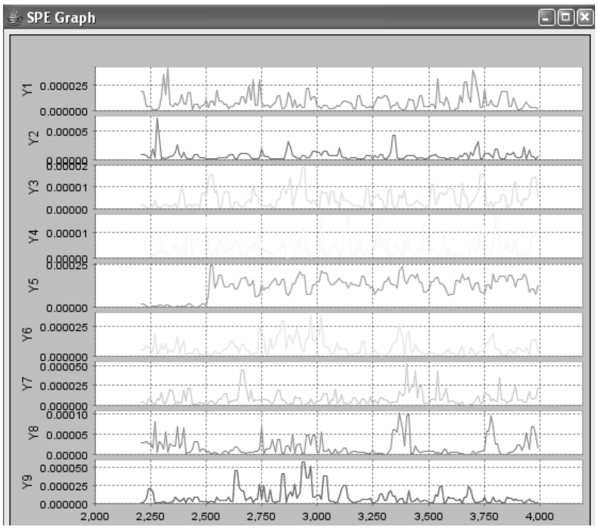

Fig. 1. SPE statistics showing the fault in reactor 5 after t = 2500.

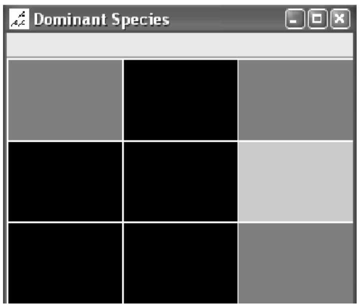

Fig. 2. The 3x3 reactor network grid with three species, black represents species 3 , dark gray represents species 2, and light gray represents species 1.

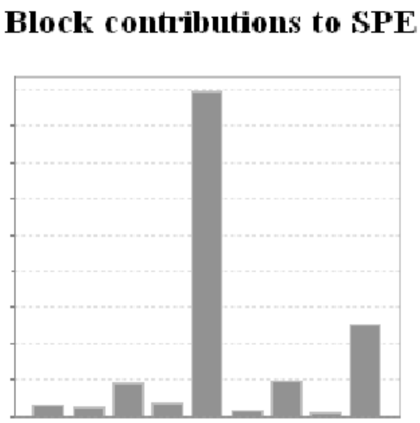

Fig. 3. Block contributions to the SPE statistic at the time of detection.

flowrate for each reactor is the same and is equal to 0.001. The default dimensionless feed flowrate of each reactor is 0.008. The three species have similar death and growth rates.

Figure 1 shows that there is a shift in the SPE statistic after the fault is introduced and the fault stays in the system until the end of the projection if no action is taken by the control agents. This shift is detected by the SPE agent at time (t = 2530), only three time steps after the fault was introduced.

The same fault is simulated again, this time with the control action activated. With random initial conditions, everything else staying the same, the multiblock model is formed, the diagnostic agents start watching the system at the same time the projections starts, and during projection the same fault is introduced to the system. After the diagnostic agents agree on the existence of a fault in the system and its location, they activate the control structure and call the optimization routine, single variable optimization to minimize the difference between reference concentration and the current dominant species concentration is done on the feed flowrate to that reactor and the system is restored to the reference state. The maximum number of iterations for the simulated annealing algorithm is set to 30, however, the optimum is usually found no later than the 7th iteration. The temperature parameter is set to 15 and is reduced 20% at every five iterations. Figure 4 shows the SPE statistics and the block contributions (Figure 3) that were used by the diagnostic agents. It is seen that after the control agents set the new feed flowrate, the system goes back to normal.

8. CONCLUSION

An agent-based automated process monitoring, fault detection, diagnosis and control structure is proposed in this study. It brings together powerful tools, like multiblock process monitoring, fault detection, diagnosis and simulated annealing search. And, it combines them with a relatively new yet again powerful approach, an agent-based system. Agent-based decision-making has been used in a few chemical engineering applications recently but its application to process monitoring and supervision is new.

When many different methods are used on the same data to gather information, in this case different fault detection methods, agents can be effectively used to clear the ambiguity in the information gathered from alternative methods, and help build a consensus.

A case study is performed in the network of reactors by introducing a disturbance to the network. The effectiveness of the proposed structure to detect the fault and take action to control the problem is demonstrated.

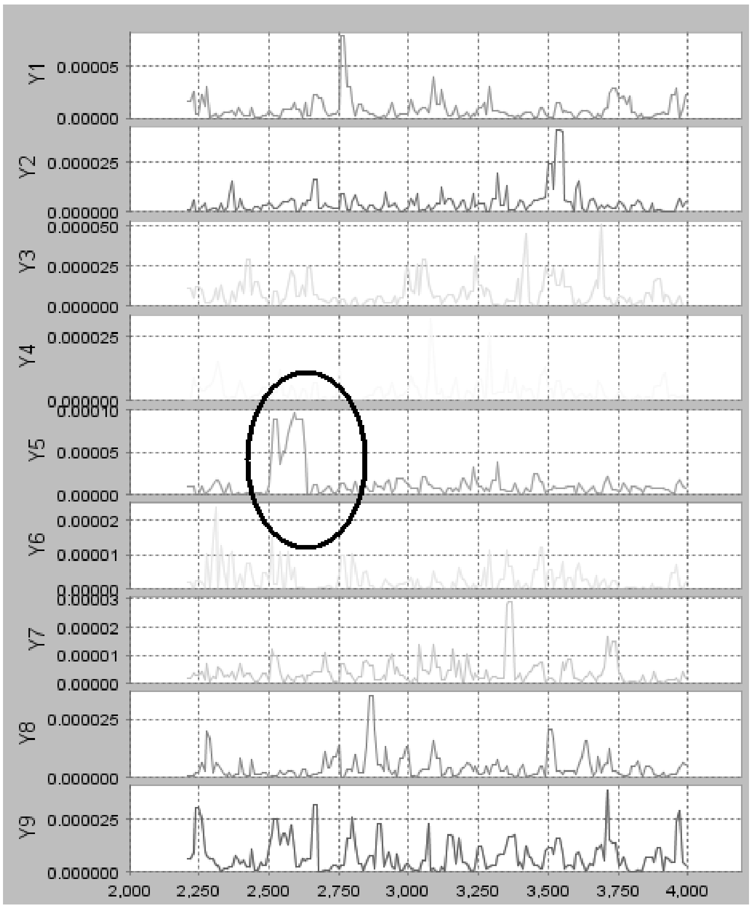

Fig. 4. SPE statistics showing the fault in reactor 5 after t = 2500, and corrective action is taken at t = 2600.

9. ACKNOWLEDGEMENTS

This material is based upon work supported by the National Science Foundation under Grant No. CTS-0325378 of the ITR program.

REFERENCES

Collier, N., T. Howe and M. North (2003). Onward and upward:the transition to repast 2.0. In: *First Annual North American Association for Computational Social and Organizational Science Conference, Pittsburgh, PA.*

Gustafsson, F. (2002). *Adaptive Filtering and Change Detection.* Wiley.

MacGregor, J.F., C. Jaeckle, C. Kiparissides and M. Koutoudi (1994). Process monitoring and diagnosis by multiblock PLS methods. *AIChE Journal* **40**, 826–838.

Nomikos, P. and J.F. MacGregor (1995). Multivariate SPC charts for batch processes. *Technometrics* **37**, 41–59.

Perk, S. and A. Çınar (2006). Batch process monitoring using multiblock multiway principal components analysis. In: *Prep. International IFAC Symposium. Advanced Control of Chemical Processes (ADCHEM), Gramado, Brazil.*

Qin, S.J., S. Valle and M.J. Piovoso (2001). On unifying multiblock analysis with application to decentralized process monitoring. *Journal of Chemometrics* **15**, 715–742.

Rardin, R. (1998). *Optimization in Operations Research.* Prentice Hall.

Smilde, A.K., J.A. Westerhuis and S. de Jong (2003). A framework for sequential multiblock component methods. *Journal of Chemometrics* **17**, 323–337.

Tatara, E., I. Birol, F. Teymour and A. Çınar (2004). Static and dynamic behavior of autocatalytic replicators in reactor networks. *Industrial Engineering and Chemistry Research* **43**, 3972–3993.

Tatara, E., I. Birol, F. Teymour and A. Çınar (2005). Agent-based control of autocatalytic replicators in networks of reactors. *Computers and Chemical Engineering* **29**, 807–815.

Westerhuis, J.A., S.P. Gurden and A.K. Smilde (2005). Generalized contribution plots in multivariate statistical process monitoring. *Chemometrics and Intelligent Laboratory Systems* **51**, 95–114.

Westerhuis, J.A., T. Kourti and J.F. MacGregor (1998). Analysis of multiblock and hierarchical PCA and PLS models. *Journal of Chemometrics* **12**, 301–321.