SICOP Round Tables on Asset Management, International Workshop on Solving Industrial Control and Optimization Problems, Cramado, Brazil, April 6-7, 2006.

TOWARDS OPTIMAL MPC PERFORMANCE: INDUSTRIAL TOOLS FOR MULTIVARIATE CONTROL MONITORING AND DIAGNOSIS

Biao Huang [*,1] **Sien Lu** [*] **Fangwei Xu** [*]

[*] *Department of Chemical and Materials Engineering,University of Alberta, Edmonton AB T6G 2G6, Canada*

Abstract: Univariate control performance assessment (PA) was developed in late 1980s and it has been widely applied in industry. Multivariate control performance assessment was developed in 1990s, but its application has been limited. While the algorithm of univariate PA is rather straightforward to write, the multivariate control performance assessment (MVPA) is much more difficult to program. In addition, conventional MVPA algorithms alone appear not to be sufficient or sometime even irrelevant for evaluating model predictive control (MPC) systems. With demands from industries, an industrial toolbox for MVPA including variance based performance monitoring, economic performance assessment, process modeling and other diagnosis algorithms has been developed. Synthesized application of this toolbox makes MVPA highly relevant in MPC monitoring. This paper gives an overview of the toolbox developed for industrial applications and elaborates the key algorithms including model-based algorithm for MVPA, newly developed data-driven model-free approach to MVPA, and MPC economic performance assessment. It is then pointed out there is a lack of systematic and synthetic performance diagnosis means in current literature. The difficulty of developing systematic diagnosis tools is addressed. The solution strategy using Bayesian graphic network is then proposed. The proposed performance diagnosis framework is illustrated through some illustrative graphic examples.

Keywords: Control monitoring, Performance monitoring, Advanced process control, Model predictive control, MATLAB toolbox, Linear matrix inequality, Performance diagnosis, Bayesian network

1. INTRODUCTION

There are hundreds to thousands of control loops in a typical plant. It is not possible for process control engineers to routinely evaluate performance of each controllers. Even though the parameters of a controller may be tuned very well when commissioned, there is no guarantee that this controller will sustain the same performance forever. A computer-aided tool that can assess the performance of the controllers automatically and routinely is in a great demand.

The minimum variance control (MVC) benchmark has been widely used since the early work of Harris (1989) . With this benchmark, the performance index (PI, also known as Harris Index) of a single-input single-output (SISO) system can be calculated as the ratio of the output variance

[1] Corresponding author; Tel: +1(780)492-9016, Fax: +1(780)492-2881, E-mail: biao.huang@ualberta.ca

under minimum variance control and the actual output variance. For the multi-input multi-output (MIMO) system, it is the ratio of the traces of output covariance.

While univariate PA has been widely applied in industry, the practical application of multivariate PA has been limited, despite there is a great demand of it. This is, in part, due to the difficulty to develop a general algorithm to deal with all formats of so called interactor matrix (no such general program is available to the best of our knowledge). More importantly, the classical MVPA only deals with variance based performance measures, which appears not to be sufficient for model predictive control systems. In view of this, to aid practical applications, we have developed a toolbox for MVPA. The toolbox consists of several functions including general interactor matrix calculation, interactor based and interactor-free multivariate PA, economic performance assessment, and robust system identification, among many other functions. The two multivariate PA functions are FCOR algorithm [Huang and Shah, 1999] and a recently developed data driven model-free algorithm. A data driven model-free approach without the knowledge of the interactor matrix has been developed by [Huang et al., 2004]. The advantage of this method is that it gives an explicit "one-shot" solution. The performance index can be directly estimated from input/output data. Therefore, no concepts, such as transfer function matrix, interactor matrix, Markov parameters, etc. are needed for MVPA.

While the research has been focusing on control performance monitoring, other developments include sensor monitoring, actuator monitoring, oscillation detection, model validation [Huang et al., 2003], model predictive control monitoring [Kesavan and Lee, 1997; Gao et al., 2003], covariance based performance measure [McNabb and Qin, 2005], PCA based diagnosis for model error and disturbance effects [Lee et al., 2004], nonlinearity detection [Choudhury et al., 2004]. These methods targeted specific problem in a control system and successful case studies have been reported. However, the interrelations between these monitoring algorithms have rarely addressed. The common practice was that one detection algorithm was developed for specific problem and then tested with the targeted problem being occurred in a simulation or in a practical problem. Obviously, different problems can result in similar symptom and affect more than one detection algorithms. There is clearly a need to consider all detection problems simultaneously in a systematic manner.

There are, however, a number of challenging issues while considering detection problems symutanesouly: 1) While the source of the problem may be unique (e.g. sensor decalibration), its symptoms can be similar to that resulted from different problem sources (e.g. process model change). The model validation algorithm that is designed to detect plant-model mismatch can not immune from sensor decalibration. Thus, while each detection algorithm may work well when only the targeted problem occurs, relying on a single detection algorithm can be misleading in general. For example, the sensor decalibration may be diagnosed as process model problem if a single model validation algorithm is applied. 2) To resolve this issue, one needs to investigate how problem sources can affect each others' detection algorithm. Sensitivity analysis of all detection algorithms to all problem sources has to be performed in order to achieve this objective. 3) All processes operate, to certain degree, in an uncertain world. The occurrence of a problem, its symptom, and its interconnection with other problems/symptoms all have certain uncertainties. A solution has to be built upon a probability framework.Thus, a joint probability distribution among all problem sources and observed/calculated symptoms need to be established. As elaborated shortly, the computation of the probabilities and statistical inferences grows exponentially with the number of problem sources and observed/calculated symptoms. 4) Most of the existing performance monitoring methods are data based. While they have clear advantages of simplicity, it is obvious that certain priori knowledge of process is not only helpful but also necessary when multiple problem sources are considered. For example, a process flow chart indicating components interrelationship may be available and should be considered in making a meaningful diagnosis and decision. It is of a considerable challenge to integrate data based algorithms with *a priori* knowledge. A Bayesian network model also known as graphical model [Murphy, 2002], elaborated in this paper, is one of the most promising frameworks for solving such problems.

The remainder of this paper is organized as follows: The architecture of the toolbox is illustrated in Section 2. The MVPA based on interactor matrix is presented in Section 3, and the model-free MVPA in Section 4. Economic performance assessment is overviewed in Section 5. A new framework for control performance diagnosis and troubleshooting is addressed in Section 6, followed by conclusion in Section 7.

2. PROCESS ANALYTICAL TOOLBOX

Advanced process control (APC) applications, such as model predictive control (MPC) technology, have been widely accepted and applied in process industries. Monitoring of APC performance especially economic performance has been a great interest both in academia and in industry. This motivated us to develop state-of-the-art algorithms that can be used to monitor the performance of APC applications and facilitate the task of APC maintenance. The software package that we have recently developed is called Performance Analysis Technology and Solutions (PATS).

The package of PATS was written in MATLAB, which is widely used in academia by researchers. MATLAB itself is a very powerful language in computation with large amount of speciality toolboxes in different research areas. Moreover, the MATLAB data link feature makes it possible to retrieve real-time process data directly from DCS system via standard OPC interface, which is available in almost all different kind of DCS systems in process industries. We have employed these features in our development of PATS package. On the other hand, it is possible for researchers to run their applications on line with direct real-time process data. This makes it a very convenient shortcut for the technology transfer from academia to process industries.

PATS is composed of application components and interface components. Two interfaces, PI Driver and Data Collector, are provided to retrieve process data, either historical process data off-line from the PI server or real-time process data on-line from the DCS system. They are designed to provide process data to the application components from different data sources. The main application components include Cluster Analysis, Process Model Identification, Univariate Time-Varying Controller Performance Assessment, Multivariate Controller Performance Assessment, Robust Minimum Energy Control with Output Covariance Constraints, MPC Economic Performance Analysis and Tuning Guidelines. The output results of these application components provide process engineers useful information on the investigated process or controller directly based on the process data, most of which are routine operating data. The outcome will be process models, controller performance of time-varying processes, multivariate controller performance, optimal designed controller, MPC economic performance, MPC tuning guidelines, and so on. The synthesis of variety of results makes this toolbox a powerful package for process and control engineers. For example, the controller performance indices obtained from multivariate controller performance monitoring component can be imported into MPC economic performance analysis and tuning component. It can then give the possible MPC economic benefit potential as well as some corresponding MPC tuning guideline. These two application components together with Data Collector will make a plant oriented solution for MPC performance monitoring. Considering

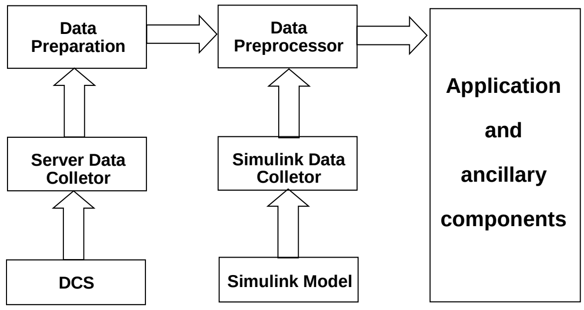

Fig. 1. PAT data collection architecture

that data from simulations may also be of interest to process control engineers, a Simulink data collector has also been integrated with PATS. The schematic diagram of the PATS data collection architecture is shown in Fig.1.

3. MVPA – MODEL BASED APPROACH

A MIMO process can be modeled as

$$y_t = Tu_t + Na_t, \tag{1}$$

where T and N are proper, rational transfer function matrices for the plant and noise, respectively; y_t is an output vector and u_t an input vector. For stochastic systems, a_t represents a white noise vector with zero mean and covariance matrix Σ_a.

Furthermore, if T is a proper, full rank transfer function matrix, a unitary interactor matrix D can be evaluated and $DT = \tilde{T}$, where $\tilde{T}$ is the delay-free transfer function matrix of T. Therefore, Equation (1) can be expressed as

$$y_t = Tu_t + Na_t = D^{-1}\tilde{T}u_t + Na_t. \tag{2}$$

Premultiplying both sides of Equation (2) by $q^{-d}D$, where d is the order of interactor matrix D, gives

$$q^{-d}Dy_t = q^{-d}\tilde{T}u_t + q^{-d}DNa_t. \tag{3}$$

Let $\tilde{y}_t = q^{-d}Dy_t$ and $\tilde{N} = q^{-d}DN$, Equation(3) becomes

$$\tilde{y}_t = q^{-d}\tilde{T}u_t + \tilde{N}a_t. \tag{4}$$

[Huang and Shah, 1999] showed that since D is a unitary interactor matrix, the minimum variance control law which minimizes the objective function of the interactor-filtered variable $\tilde{y}_t$, $J_1 = E(\tilde{y}_t^T\tilde{y}_t)$, also minimizes the objective function of the original variable y_t, $J_2 = E(y_t^T y_t)$, and $J_1 = J_2$, which means that $E(\tilde{y}_t^T\tilde{y}_t) = E(y_t^T y_t)$. From $\tilde{y}_t$, performance indices can be estimated.

Two general Matlab functions, *interactor* and *fcor*, have been programmed for this algorithm. The function *interactor* is for calculation of interactor matrices while *fcor* is for estimation of performance indices. Simulation examples designed for the test of *fcor* function is illustrated next. The example is from [Huang and Shah, 1999].

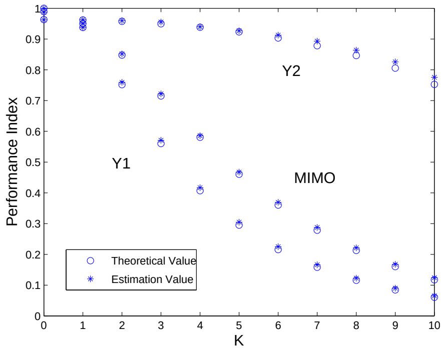

Fig. 2. Performance assessment result by the FCOR algorithm

$$T = \begin{bmatrix} \frac{q^{-1}}{1-0.4q^{-1}} & \frac{Kq^{-2}}{1-0.1q^{-1}} \\ \frac{0.3q^{-1}}{1-0.1q^{-1}} & \frac{q^{-2}}{1-0.8q^{-1}} \end{bmatrix}$$

$$N = \begin{bmatrix} \frac{1}{1-0.5q^{-1}} & \frac{-0.6}{1-0.5q^{-1}} \\ \frac{0.5}{1-0.5q^{-1}} & \frac{1.0}{1-0.5q^{-1}} \end{bmatrix}$$

$$Q = \begin{bmatrix} \frac{0.5-0.20q^{-1}}{1-0.5q^{-1}} & 0 \\ 0 & \frac{0.25-0.200q^{-1}}{(1-0.5q^{-1})(1+0.5q^{-1})} \end{bmatrix}$$

T is a 2×2 MIMO process, Q is the controller and N is the disturbance transfer function matrix. The white noise excitation, a_t, is a two-dimensional normally-distributed white noise sequence with $\Sigma_a = I$.

By the *interactor* function, the interactor matrix of T is calculated as

$$D = \begin{bmatrix} -0.9578q & -0.2873q \\ 0.2873q^2 & 0.9578q^2 \end{bmatrix}.$$

From the interactor matrix D and the noise model N, the theoretical values of the performance index can also be calculated. The performance assessment results from data are shown in Figure 2. From this figure, we can see that the theoretical values and the estimation values match well for different K.

4. MVPA – MODEL FREE APPROACH

When the interactor matrix is known a *prior*, the performance index can be estimated from a set of closed-loop routine operating data. Although the interactor matrix is a meaningful generalization of the time delay term for multivariate systems, both its concept and calculation algorithm are an obstacle, especially, for industrial users. Therefore, the challenge for the performance assessment of MIMO system is the calculation of the performance index without the knowledge of the interactor matrix.

Ko and Edgar (2001) developed a method which integrates the calculation of interactor matrix and the estimation of performance index without explicitly calculating the interactor matrix. However, this approach still needs to have the Markov matrices of the system transfer function matrix as *a priori*. Huang *et al.* (2004) developed a truly model-free and interactor-free method to calculate the performance index directly from input/output data without the knowledge of interactor matrix or Markov matrices.

4.1 Algorithm

Describe a linear time-invariant system with l-inputs, m-outputs and n-states using the innovations state space representation as

$$\begin{aligned} x_{t+1} &= Ax_t + Bu_t + Ke_t \\ y_t &= Cx_t + e_t \end{aligned} \tag{5}$$

where the dimensions of system state space matrices A, B, C and K are $n \times n, n \times l, m \times n$ and $n \times m$, respectively. K is the Kalman filter gain and e_k is an unknown innovation or white-noise sequence with covariance matrix S.

Stack the input y_t into two block Hankel matrices,

$$Y_p = \begin{bmatrix} y_1 & y_2 & \cdots & y_j \\ y_2 & y_3 & \cdots & y_{j+1} \\ \vdots & \vdots & \vdots & \vdots \\ y_N & y_{N+1} & \cdots & y_{N+j-1} \end{bmatrix} \tag{6}$$

and

$$Y_f = \begin{bmatrix} y_{N+1} & y_{N+2} & \cdots & y_{N+j} \\ y_{N+2} & y_{N+3} & \cdots & y_{N+j+1} \\ \vdots & \vdots & \vdots & \vdots \\ y_{2N} & y_{2N+1} & \cdots & y_{2N+j-1} \end{bmatrix} \tag{7}$$

where p denotes the past and f denotes the future.

$$H_N = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ CB & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ CA^{N-2}B & CA^{N-3}B & \cdots & 0 \end{bmatrix}$$

and

$$H_N^s = \begin{bmatrix} I_m & 0 & \cdots & 0 \\ CK & I_m & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ CA^{N-2}K & CA^{N-3}K & \cdots & I_m \end{bmatrix}$$

are the lower triangular Toeplitz matrices corresponding to the deterministic input u_k and the unknown stochastic input e_k, respectively, Subspace method of identification has two steps. The first step is the projection of certain subspace generated from the data in order to get the estimation of subspace matrices such as H_N^s and H_N. The second step is the estimation of system state space matrices A, B, C and K from matrices such as H_N^s and H_N . The method introduced in this section only needs the estimation of subspace matrices, without the knowledge of system state space matrices.

Define $J_{mvc} = Ey_t^T y_t|_{mvc}$. It is shown in [Huang et al., 2004] that, the MVC benchmark can be estimated as

$$\hat{J}_{mvc} = tr(I - \hat{H}_N \hat{H}_N^{\dagger})\hat{L}_{h,1}\hat{L}_{h,1}^T(I - \hat{H}_N \hat{H}_N^{\dagger})^T. \tag{8}$$

where $\hat{H}_N^{\dagger}$ is a pseudo inverse of $\hat{H}_N$, $L_{h,1}$ is the first block column of L_h, and

$$\hat{H}_N = (Y_f - Y_f/_{U_f} W_p)U_f^{\dagger}$$
$$\hat{L}_h = \frac{1}{\sqrt{j}}(Y_f^{cl} - Y_f^{cl} Y_p^{cl\dagger} Y_p^{cl})Q^T \tag{9}$$

Eqn.(9) is derived from following QR decomposition:

$$\hat{L}_h Q = \frac{1}{\sqrt{j}}(Y_f^{cl} - Y_f^{cl} Y_p^{cl\dagger} Y_p^{cl})$$

In the above equations, Y_f, Y_p, U_f are block Hankel matrices formed according or similar to eqn.(6) and (7) using open-loop experiment data; Y_p^{cl}, Y_f^{cl} are formed using closed-loop routine operating data; $W_p = [Y_p, U_p]^T$; $Y_f/_{U_f} W_p$ is oblique projection of row space of Y_f onto row space of W_p via U_f.

The algorithm developed above is based on the subspace method, by which H_N is replaced by a set of open loop experimental data and H_N^s by a set of closed-loop routine operating data. The final expression of the minimum variance lower bound is directly based on data. With this algorithm, there is no need to know interactor matrix, Markov parameters, or transfer function matrices. The only information needed is two sets of data, one open-loop experiment data and one closed-loop routine operating data.

4.2 Matlab Function

A Matlab function, *pass*, is programmed for this subspace method. *Pass* stands for Performance Assessment by SubSpace method.

The outputs of this function are performance indices of overall system, η, and that of individual output loop, η_n. Three kinds of performance index can be calculated by this function: estimation value, theoretical value and mixed value.

4.2.1. Estimation value As introduced, the performance index can be estimated directly from a set of open loop experimental data and a set of closed-loop routine operating data by this subspace method. Therefore, this performance index is named as estimation value.

For the estimation value, use command

$$[Eta, Eta_n] = pass(U_{open}, Y_{open}, Y_{close}, \text{`}est\text{'}).$$

The input parameters U_{open} and Y_{open} are a set of open loop experimental input/output data and Y_{close} is a set of closed-loop routine operating output data.

4.2.2. Theoretical value On the other hand, if the models of plant, controller and noise are available, the theoretical value of H_N and H_N^s are known because they are comprised of Markov parameter of plant and noise system, respectively. Then, the theoretical value of J_{mvc} can be calculated as

$$J_{mvc} = tr(I - H_N H_N^{\dagger}) H_{N,1}^s S (H_{N,1}^s)^T (I - H_N H_N^{\dagger})^T. \tag{10}$$

where S is the square root of the covariance matrix of white noise.

For the theoretical value, use command

$$[Eta, Eta_n] = pass(plant, controller, noise, S, \text{`}the\text{'}).$$

The input parameters $plant, controller, noise$ are models of plant, controller and noise, respectively, expressed in transfer function matrix.

4.2.3. Mixed value If only the plant model is available, "mixed value" of J_{mvc} can be calculated as

$$\hat{J}_{mvc} = tr(I - H_N H_N^{\dagger})\hat{L}_{h,1}\hat{L}_{h,1}^T(I - H_N H_N^{\dagger})^T, \tag{11}$$

where H_N is calculated from plant model and $\hat{L}_h$ is estimated from a set of closed-loop data.

For the mixed value, use command

$$[Eta, Eta_n] = pass(plant, Y_{close}, \text{`}mix\text{'}).$$

4.3 Simulation Example

The example that has been employed to illustrate the FCOR algorithm is used to test this subspace performance assessment method. Four kinds of performance index are shown in Figure 3. The theoretical and estimation values of FCOR algorithm are the same as those shown in Figure 2. For the subspace method, the theoretical and estimation values are calculated by the function *pass*. As we can see, they all match well. Both theoretical values are exactly same, which shows match of the two methods in theory. Compared with the

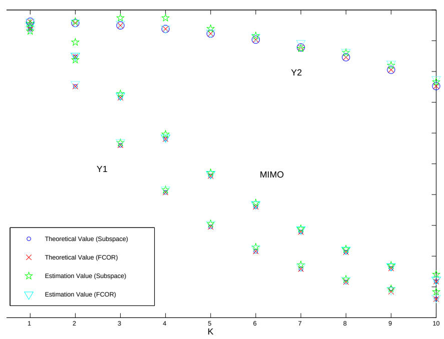

Fig. 3. Performance assessment results by the FCOR algorithm and subspace method

FCOR algorithm, moreover, this subspace method has obvious advantage. No concept and computation of, such as, interactor matrix, is needed. We can directly estimate the performance index from data, a set of open loop experiment data and a set of closed-loop routine operating data.

5. MPC ECONOMIC PERFORMANCE ASSESSMENT AND TUNING

Variance based MVPA as illustrated in the previous two sections appears not to be sufficient for model predictive control monitoring and diagnosis. An algorithm based on the Linear Matrix Inequality (LMI) for MPC economic performance analysis and for providing MPC tuning guidelines (named LMIPA) has been developed. Some mathematical details can be found in [Xu et al., 2006]. The MVPA algorithms are integrated with LMIPA to form key components of PATS. LMIPA calculates the potential of economic performance improvement and provides the tuning guidelines on how to materialize the economic potential. MVPA calculates the dynamic performance of MPC, such as how much variability can be reduced by tuning the control. Synthesis of LMIPA and MVPA is an achievement of software development which aims at transferring the research results into the process industry. The synthesized framework has been tested in an industrial process. The process of concern has a considerable size and is qualified as a great challenging problem (42 outputs and 15 inputs) to test the algorithms. We were provided with process step responses data, closed-loop routine operating data, and optimization coefficients of the current MPC application, based on which we can then draw conclusions on the potential of further economic improvement relative to the existing MPC application and make suggestions on how the further economic performance improvement of the MPC may be materialized. Some of sample conclusions are demonstrated below:

(1) *An absolute maximum in an ideal scenario:* If the disturbances of the process all turn to zero, the maximum possible increase of the yield can be calculated under the existing constraints on the variables according to LMIPA. This increase may be achieved by pushing the operating points to the optimum subject to constraints and model relation. LMIPA can determine the optimal operating point.
(2) *A maximum in a realistic scenario:* With the disturbances, the maximum possible increase of the yield without changing the constraints or trying to reduce process variability can be calculated according to LMIPA. This increase may be achieved by simply pushing the operating points to the optimum subject to constraints and model relation. LMIPA can determine how this can be done.
(3) According to (1) and (2), one can determine how much disturbance has costed the process even at the optimal tuning scenario, and certainly more lost is expected in non-optimal tuning. Reduction of this lost due to the variability is where the process control engineers should work on. LMIPA-MVPA provides such information.
(4) LMIPA-MVPA can be used to calculate the monetary incentive of reducing variability of certain variables by tuning. For example, if a control engineer offers to tune the controllers, say y_{10} be reducing variability by 1/3. Then the natural question is whether this effort can have any economic benefit. According to the calculation by LMIPA, it may turn out that there is no economic improvement at all relative to that achieved. On the other hand, if, say, y_1 be reduced variability by 1/3, then the calculation may indicate that there will be significant benefit.
(5) The calculation in (4) is passive. However, LMIPA-MVPA can also make an active calculation to search for the optimum and provide the tuning guidelines such as which variables should be reduced variability for certain economic benefit. This is discussed next.
(6) If, say, control engineers are confident that the variability of the process can be reduced by up to 20%, then LMIPA can estimate how much the yield could be potentially increased. LMIPA-MVPA can determine how this can be done and which variables should be reduced variability and by how much.
(7) If, say, the operators/engineers are allowed to increase the constraints on certain variables by, say, up to 5%, then LMIPA can determine how much the yield could be increased. LMIPA-MVPA can determine how this can be done and which variables should be increased the constraints and by how much.

(8) The maximum percentages of variability reduction and allowable range increase in (6) and (7) may be arbitrarily specified and allowed to be different in each variables, LMIPA-MVPA can deliver the conclusion and suggestions as in (6) and (7).
(9) If, say, managers want to set a target such that the yield be increased by $10,000/day, then LMIPA-MVPA will determine how this may be done and which variables should be reduced the variability and by how much. LMIPA also optimizes the result by minimizing the effort to decrease the variability.
(10) Alternatively, LMIPA also determines how the constraints may be increased to gain the $10,000/day benefit and which variables should be increased the constraints and by how much. LMIPA also optimizes the result by minimizing the increase of the constraints.
(11) The monetary increase of the yield in (9) and (10) can be arbitrarily specified. LMIPA-MVPA delivers the conclusion and suggestions as in (9) and (10) if they are practically feasible. If not, LMIPA-MVPA can also indicate otherwise.
(12) Synthesis of LMIPA, MVPA, and other components of PATS will tell how much variability of process can be reduced, and how much economic benefits may be achieved under the optimal variability reduction, and how they should be done. This result extends the current research activities in achievable (variance) performance limit to achievable economic performance limit.
(13) Finally PATS has an open framework for including new functions or improving the existing functions. For variability reduction analysis, the existing LMIPA algorithm targets economic objective that depends on variability reduction of the output variables only. The inclusion of manipulated variables to the economic objective for variability reduction analysis turns out to be more complicated and needs complete dynamic models, and the work is in progress to simplify this requirement.

6. A NEW FRAMEWORK FOR PROBABILISTIC INFERENCING FOR DIAGNOSIS OF POOR CONTROL PERFORMANCE

6.1 Bayesian network for diagnosis

Monitoring of MPC performance has achieved considerable progress as discussed in the last section. Diagnosis of poor performance for a MPC system is, however, relatively behind despite there are a number of monitoring tools reported targeting individual problem sources. This is due to in part the complexity of MPC and lack of a systematic framework for the diagnosis.

A typical control system consists of at least four components, sensor, actuator, controller and process, each subject to possible performance degradation or failure. Any problem in one of these four components can affect control system performance. Each of them may have its own monitoring algorithms to detect the problems and these algorithms may all be affected by one or more of the four components. The relationship between the problem sources and the monitors may be drawn as a network.

Imagine a simplest network of eight nodes, representing four problem sources and four monitors, and they are related by probabilities. To completely determine the relation among all nodes, we need to know the joint probabilities of *eight* random variables. Even if each node has only two outcomes (e.g, fault vs non-fault), $2^8 - 1$ joint probability needs to be determined for the 8 node network. With increased values of each node to be considered and more nodes to be added, the complexity of the network can quickly go beyond computational possibility, a classical problem of "curse of dimension". A recently developed Bayesian network graphic model, which explores the sparse structure of the network, is the choice of the method that sheds a light on solution of the problem.

The building blocks of the graphic model are a network of nodes connected by conditional probabilities. These nodes are random variables, which can be continuous, discrete or simply binary. Consider the simplest binary random variables. If there are n binary random variables, the complete distribution is specified by $2^n - 1$ joint probabilities. In the illustrative figure, Fig. 4, there are 4 binary nodes, each node having two possible outcomes. For example, node A may take the value A or $\bar{A}$. To completely determine the distribution of the 4 binary variables, we need to determine joint probability $P(A, B, C, D)$ that has 16 outcomes. By taking account that sum of all probabilities must equal to 1, we need to calculate 15 probabilities. However, as illustrated in Fig. 4, by exploring the graphic relationship of each node, only 7 probabilities need to be determined, a considerable reduction from 15 calculations. The structure of the graphic relationship is the example of incorporating the *a priori* process knowledge and takes advantage of sparse structure of probabilistic relations among the nodes. With the increase of the nodes, the saving of computations is exponential, making it possible to apply statistical inference theory into practice.

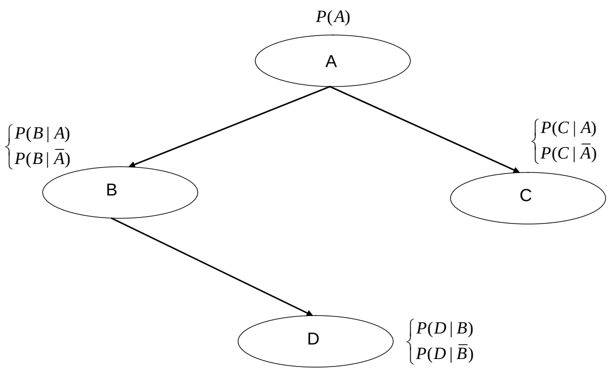

Fig. 4. An example of BN

If the network chart like Fig. 4 is available, one can make a variety of inferences. For example, if we have the observations of B, C, D, we would like to make a decision to determine whether A=A or A=$\bar{A}$. The decision process can be written under Bayesian framework as $P(A|BCD)$ that can be calculated, according to Bayes rule

$$P(A|BCD) = \frac{P(ABCD)}{P(BCD)} = \frac{P(ABCD)}{\sum_A P(ABCD)}$$

Using the rule of the Bayesian network according to the relationship of four nodes in Fig. 4, the joint probability can be calculated as

$$P(ABCD) = P(A)P(C|A)P(D|B)P(B|A)$$

The seven probabilities specified in Fig. 4 are sufficient to calculate $P(A|BCD)$ to make an optimal inference about the state of A. To show the flexibility of the graphic model approach, we consider the inference of C given observation D where in this case we assume C can not be observed. According to Bayes rule, it can easily be derived that

$$P(C|D) = \frac{\sum_{AB} P(ABCD)}{\sum_{ABC} P(ABCD)}$$

Due to the unavoidable uncertainties in practice, none of the detection algorithms can give a definite conclusion. A decision is usually made according to certain probabilistic confidence or risk. Any detection algorithm can at most give the most likely problem source according to its probabilistic inferencing. In addition to the "most likely" problem inferred, there are a "second most likely" problem, third, and so on. According to the diagnostic results, engineers or instrument technicians have to service the specific controller or the instrument according to a troubleshooting sequence. There is an associated cost for each service, some more and some less. Interestingly, an optimal troubleshooting sequence does not necessarily follow the order of likelihood of the problems [Heckerman et al., 1995]. An optimal sequence of the service according to not only the occurrence probability of the problem but also the cost of the service has to be established. As an example, if there are two possible problem sources, A and B, and there are two observations C and D. The conditional probability has been calculated as $P(\bar{A}|CD) = 0.6, P(\bar{B}|CD) = 0.4$, while the service cost (confirming and/or fixing the problem) for A is \$1000 and for B is \$2000. Now the question is whether one should service the instruments according to the sequence AB or BA? An objective function considering the probabilities together with the service costs should be formulated and optimized to find the best service sequence.

While many monitoring algorithms have been developed, many are being or to be developed. The future research direction not only needs to consolidate and enhance monitoring algorithms that have been developed, but also needs to integrate them into the new probabilistic diagnosis and troubleshooting network. One of the objectives of this paper is to establish a new general framework for control system diagnosis and troubleshooting, particularly for the most common performance related problems and their diagnosis/troubleshooting, namely control tuning problem, process model mismatch, actuator problem, and sensor problem, and demonstrate the feasibility through several simplified inferencing examples.

6.2 Decision making for performance diagnosis

Let's assume, at the moment, that monitors for all four problem sources are available. Furthermore, for simplicity of illustration, we assume that sensor monitor, actuator monitor, and model validation (monitor) are designed in such way that they are only sensitive to their own problem sources. Performance monitor, on the other hand, is sensitive to all problem sources. For a batch analysis, we do not have to consider the evolution of problem sources with time; thus a static Bayesian network can be built shown in Fig.5, where shaded nodes are hidden and others are observed (evidence bode). Each evidence node is a child of the problem source node (parent node). Clearly the performance monitor is a child of all three problem source nodes. The structure of this network also implies that four evidence nodes are conditionally independent, conditioning on the three problem source nodes, an assumption that may not be rigorously true but nevertheless simplifies network inference considerably.

With availability of the three evidence nodes on the right hand side of three problem source nodes, it seems sufficient to solve the problem. Why one needs the performance monitor node?

Apart from the obvious fact that this additional evidence node increases credibility (certainty)

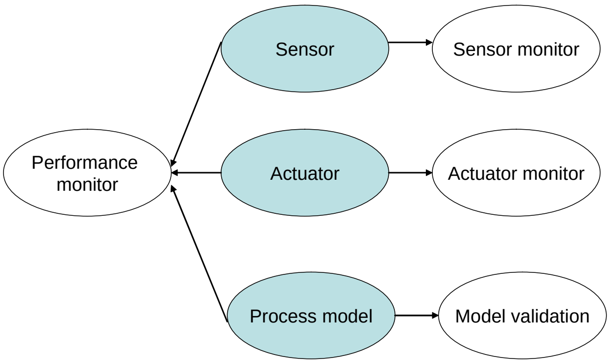

Fig. 5. An example of static Bayesian network for performance monitoring

when perform inference (diagnosis), it can play a substitution role if one of the other nodes become unavailable (e.g. due missing data). Furthermore, if other three nodes do not provide clear inference such as, the evidence is somewhere between yes and no, then this fourth node plays an important role in reducing the ambiguity.

The performance monitor node plays another interesting role in this network. Without evidence from performance monitor the three problem nodes are independent from the assumption. With the evidence the three problem nodes become dependent. For example, the actuator fault may be independent of sensor fault or process model parameter change. However, if performance monitor indicates a change of control performance, knowing actuator being the problem will reduce the chance for sensor or process model to have the problem.

Inference from the static Bayesian network is relatively simple. Many standard Bayesian network inferencing software can readily provide solutions.

For on-line application, the static Bayesian network is not sufficient. In this case, the problem sources may be temporally dependent. For example, if a sensor has mean time between failure (MTBF) of 1000 hrs. The sensor monitor is evaluated every 1 hr. If sensor has no fault at current time t, then the probability of no fault at next evaluation time $t+1$ would be $1-1/1000 = 0.999$. On the other hand, for an old sensor that has MTBF of 10 hrs, the probability becomes 0.9. Knowing this temporal probability dependence can significantly reduce false alarms as shown by Smyth (1994). Therefore, a dynamic Bayesian network should be created for on-line monitoring and diagnosis. An example is shown in Fig.6, where only two time slices are shown at time t and $t+1$, respectively. Repeating these two slices provides entire network. This network is also known Dynamic Bayesian Network, and once again, may be solved by dynamic Bayesian inference algorithms [Murphy, 2002; Smyth, 1994].

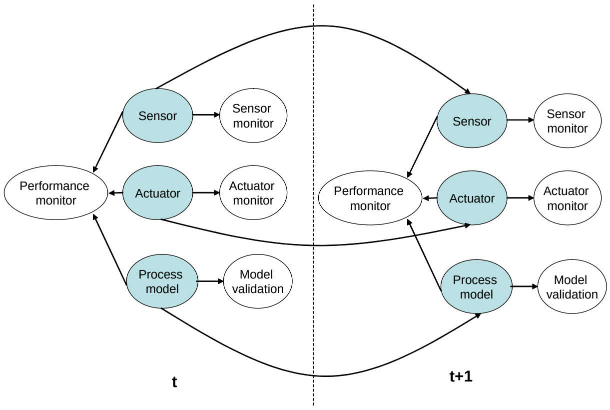

Fig. 6. An example of dynamic Bayesian network for performance monitoring

7. CONCLUSIONS

In this paper, a newly developed toolbox for multivariate control performance monitoring and diagnosis is presented. The toolbox contains several main functions including, general interactor matrix calculation, model-based and model-free MVPA, MPC economic performance assessment and tuning guideline, robust system identification. The model-free MVPA, a subspace method without a prior knowledge of the interactor matrix, simplifies the calculation of performance index and gives an explicit "one-shot" solution. The performance can be assessed from a set of open loop experimental data and a set of closed-loop routine operating data. These functions have been tested on simulation examples and integrated into the software package PATS. The MPC economic performance assessment and tuning guideline is highly relevant to model predictive control, and its integration with MVPA provides a powerful tool for guiding variance reduction, performance diagnosis, economic performance monitoring and tuning for model predictive control. A limited subset of the Matlab functions for MVPA will be available upon request.

In view of a lack of systematic performance diagnosis means in current literature, a new framework for systematic control performance diagnosis and troubleshooting is proposed. The difficulty of developing systematic diagnosis tools is addressed. The solution strategy using Bayesian graphic network is then elaborated. The proposed performance diagnosis framework is illustrated through some graphic examples.

REFERENCES

M. Choudhury, S.L. Shah, and N.F. Thornhill. Diagnosis of poor control-loop performance using higher-order statistics. *Automatica*, 40(10): 1719–1728, 2004.

J. Gao, R. Patwardhan, K. Akamatsu, Y. Hashimoto, G. Emonto, S.L. Shah, and B. Huang. Performance evaluation of two industrial mpc controllers. *Control Engineering Practice*, 11:1371–1387, 2003.

T. Harris. Assessment of closed loop performance. *Can. J. Chem. Eng.*, 67:856–861, 1989.

D. Heckerman, J.S. Breese, and K. Rommeise. Decison-theoretic troubleshooting. *Communications of the ACM*, 38(3):49–57, 1995.

B. Huang and S.L. Shah. *Control Loop Performance Assessment: Theory and Applications.* Springer Verlag, London, 1999.

B. Huang, A. Malhotra, and E.C. Tamayo. Model predictive control relevant identification and validation. *Chemical Engineering Sciences*, 58: 2389–2401, 2003.

B. Huang, R. Kadali, and S.X. Ding. A direct input/output data-driven approach to multivariable feedback control performance assessment without interactor matrices. *Submitted to Control Engineering Practice (Special Issue)*, 2004.

P. Kesavan and J.H. Lee. Diagnostic tools for multivariable model-based control systems. *Ind. Eng. Chem. Res.*, 36:2725–2738, 1997.

B.S. Ko and T.F. Edgar. Performance assessment of multivariate feedback control systems. *Automatica*, 37(5):899–905, 2001.

S. Lee, Yeom S, and Lee K.S. Methods for performance monitoring and diagnosis of multivariable model-based control systems. *KOREAN JOURNAL OF CHEMICAL ENGINEERING*, 21(3):575–581, 2004.

C.A. McNabb and S.J. Qin. Projection based mimo control performance monitoring: Ii - measured disturbances and setpoint changes. *Journal of Process Control*, 15(1):89–102, 2005.

K.P. Murphy. *Dynamic Bayesian Networks: Representation, Inference and Learning.* PhD thesis, University of California, Berkeley, 2002.

C.V. Rao and J.B. Rawlings. Constrained process monitoring: A moving horizon approach. *AIChE Journal*, pages 97–109, 2002.

P. Smyth. Hidden markov models for fault detection in dynamic systems. *Pattern Recognition*, 279:149–164, 1994.

F. Xu, B. Huang, and E.C. Tamayo. Assessment of economic performance of model predictive control through variance/constraint tuning. In *ADCHEM*, Gramado, Brazil, April 2006.