Fault diagnosis of industrial systems with bayesian networks and mutual information

Sylvain Verron, Teodor Tiplica, Abdessamad Kobi

Fault diagnosis of industrial systems with bayesian networks and mutual information

Sylvain VERRON, Teodor TIPLICA, Abdessamad KOBI

Abstract— The purpose of this article is to present two new methods for industrial process diagnosis. These two methods are based on the use of a bayesian network. An identification of important variables is made by computing the mutual information between each variable of the system and the class variable. The performances of the two methods are evaluated on the data of a benchmark example: the Tennessee Eastman Process. Three kinds of fault are taken into account on this complex process. The challenging objective is to obtain the minimal recognition error rate for these three faults. Results are given and compared on the same data with those of other published methods.

I. INTRODUCTION

Nowadays, industrial processes are more and more complex. So, they include a lot of sensors. Consequently, an important amount of data can be obtained from a process. A process dealing with many variables can be named multivariate process. But, the monitoring of a multivariate process cannot be reduced to the monitoring of each process variable because the correlations between the variables have to be taken into account. The process monitoring is an essential task. The final goal of the process monitoring is to reduce the variability of the process, and so, to improve the quality of the product [1]. The process monitoring comprises four procedures: fault detection (decide if the process is under normal condition or out-of-control); fault identification (identify the variables implicated in an observed out-of-control status); fault diagnosis (find the root cause of the fault); process recovery (return the process to a normal status).

Three major kinds of approaches exist for the process monitoring [2]: data-driven, analytical and knowledge-based. Theoretically, the best method is the analytical one because this method constructs mathematic models of the process. But, for large systems (lots of inputs, outputs and states), obtaining detailed models is almost impossible. In the knowledge-based category are placed the methods that are based on qualitative models. Finally, the data-driven methods are techniques based on rigorous statistical developments of the process data. Our interest is to monitor large systems, and so, we are concerned with the data-driven methods.

In the literature, we can find many different data-driven techniques for the process control. For the fault detection of industrial processes many methods have been proposed: univariate statistical process control (Shewhart charts) [3], [4], multivariate statistical process control (T^2 and Q charts)

[5], [6], and some PCA (Principal Component Analysis) based techniques [7] like the Multiway PCA or the Moving PCA [8]. An interesting article is [9] in which authors make some comparisons between these different techniques. For the fault identification procedure, one of the better statistical techniques is the MYT decomposition of the T^2 statistic [10], [11]. Finally, for the fault diagnosis techniques we can cite the book of Chiang, Russell and Braatz [2] which presents a lot of them (PCA based techniques, Fisher Discriminant Analysis, PLS based techniques, etc).

The purpose of this article is to present two new procedures for the diagnosis of faults in large industrial systems. These new methods are based on bayesian networks and on the mutual information between the system variables and the fault category. We will use the mutual information in order to select the key variables of the process and we will use a bayesian network in a classification goal.

The article is structured in the following manner. In section II, we introduce the bayesian network (BN) classifiers and the specialized ones we use in this article, and we also introduce the notion of mutual information. After that, in section III, we present two new procedures for the diagnosis of the faults in an industrial system. Section IV presents an application of these procedures for the diagnosis of three types of fault on the benchmark Tennessee Eastman Problem. Finally, we conclude on the interests and the limitations of each procedure, and we present perspectives of the fault diagnosis with bayesian network classifiers.

II. TOOLS

A. Classification task

Once a problem (fault) has been detected in the evolution of the process by the mean of a detection method, we need to identify (diagnosis) the belonging class of this fault. Thereby, the diagnosis problem can be viewed as the task to correctly classify this fault in one of the predefined fault classes. The classification task needs the construction of a classifier (a function allocating a class to the observations described by the variables of the system). Two types of classification exist: unsupervised classification which objective is to identify the number and the composition of each class present in the data structure; supervised classification where the number of classes and the belonging class of each observation are known in a learning sample and whose objective is to class new observations to one of the existing classes.

In the case of industrial system diagnosis, we can frequently have a database containing recorded observations of previous faults of the system. So, we can easily use

supervised classification to classify a new faulty observation. In this article, we decided to use one of these supervised classification tools, precisely the bayesian network.

B. Bayesian network classifiers

A Bayesian Network (BN) [12], [13], [14] is an acyclic graph. In this graph, each variable is a node that can be continuous or discrete. Edges of the graph represent dependence between linked nodes. A formal definition is given here.

A bayesian network is a triplet $\{\mathbf{G}, \mathbf{E}, \mathbf{D}\}$ where:

- $\{\mathbf{G}\}$ is a directed acyclic graph, $\mathbf{G} = (V, A)$, where V is the ensemble of nodes of $\mathbf{G}$, and A is the ensemble of edges of $\mathbf{G}$,
- $\{\mathbf{E}\}$ is a finite probabilistic space (Ω, Z, p), where Ω is a non-empty space, Z is a collection of subspace of Ω, and p is a probability measure on Z with $p(\Omega) = 1$,
- $\{\mathbf{D}\}$ is an ensemble of random variables associated to the nodes of $\mathbf{G}$ and defined on $\mathbf{E}$ such as:

$$p(V_1, V_2, \ldots, V_n) = \prod_{i=1}^{n} p(V_i | C(V_i)) \qquad (1)$$

where $C(V_i)$ is the ensemble of causes (parents) of V_i in the graph $\mathbf{G}$.

Bayesian network classifiers are particular bayesian networks [15], [16]. They always have a discrete node C coding the r different classes of the system. The other variables $(X_1, \ldots, X_p)$ represent the p descriptors (variables) of the system. A Naïve Bayesian Networks (NBN) is a particular type of BN classifiers [17], [16]. It is also known as the Bayes classifier. A NBN links the class node to all the other variables of the system (descriptors) as indicated on the figure 1.
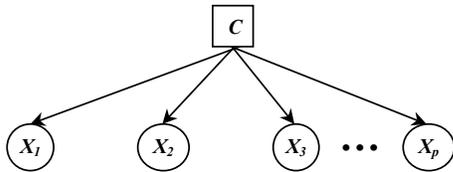

Fig. 1.   Naïve bayesian network

The NBN is called naïve because it makes the naïve (but strong) assumption that all the descriptors (variables of the system) are statistically independent (no correlation between each other). In [18] Inza made a comparative study of classification algorithms used in artificial intelligence. Performances of NBN were compared with other methods like hierarchic classification, k-nearest neighborhood, C4.5, decision tree and so on. In the case that the independence assumption of the descriptors is verified and that the probabilities are estimated with enough precision, Yang [19] demonstrated that the NBN is an optimal classifier in term of misclassification rate. This optimality is obtained when continuous variables are discretized in such a way that every distribution can be well approximated by discretization. Of course, the discretization of the variables is a loss

of information because it is a reduction of the variables space. But, assuming that continuous variables follow normal probability density function, we can deal with them directly in the network. And, if this assumption is verified, we keep this optimality. In our case (an industrial system), many variables are normally distributed. This assumption is verified in the majority of industrial systems. From the historical database of our system we will be able to estimate the parameters of the continuous (normal) variables. The Maximum Likelihood Estimation (MLE) and the bayesian approach are two different techniques of estimation of these parameters. In this paper, the parameters of the network are estimated with the MLE method. In the case that all the descriptors are continuous variables (with normal distribution assumption) and that the parameters are estimated with MLE, we obtain the known classifier: diagonal quadratic discriminant analysis.

But, this classifier is not well fitted for industrial systems. Indeed, in industrial systems, it is very frequent to have high correlations between the variables, and a NBN will not take into account these correlations. Some extensions of the NBN have been developed in order to solve this problem.

A first interesting extension is the TAN (Tree-Augmented bayesian Network) [16]. In a TAN, a maximum weighted spanning tree is constructed with the p descriptors following the algorithm of Chow et Liu [20]. So, each descriptor will have at most one other descriptor as parent. After that, the edges from the class node to each descriptor are added (like a NBN). An example of a TAN is given on the figure 2.
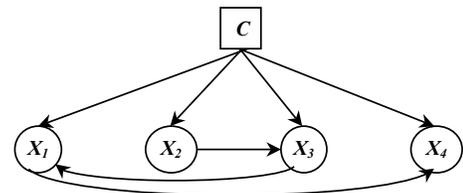

Fig. 2.   Tree-Augmented bayesian Network

Another extension is the k-dependence Bayesian classifier structure (kDB structure) [21] that extends the TAN structure in allowing a maximum of k predictor parents plus the class for each predictor variable (TAN structures are equivalent to kDB structures with k = 1). Finally, the bayesian multinets [22] introduces a different TAN structure for each values of the class variable.

Friedman shows that these classifiers give a lower misclassification rate than the NBN [16]. But, these classifiers do not take into account all the correlations between variables. An other extension to the NBN is the Condensed Semi Naïve Bayesian Network (CSNBN) [23]. The principle of this classifier is to represent some variables in a joint node. So, some normally distributed variables can be modeled with a node representing a multivariate normal distribution. In that way, all the correlations of the system will be taken into account. A CSNBN will be composed of two nodes: the class node and a multivariate node. An example of a CSNBN is

given on the figure 3.

Fig. 3.   Condensed Semi Naïve Bayesian Network

As the NBN gives a discriminant rule equivalent to a diagonal quadratic discriminant analysis, the CSNBN is equivalent to the discriminant rule of the quadratic discriminant analysis. Although this classifier is well fitted for the classification task of faults in industrial systems, it remains a problem. If we have non-informative descriptors, the performances (in term of correct classification rate) are poor. So, if we want to diagnosis a system with many variables, even though only few are really important for classification, other (less important) must not be taken into account. We have to do a selection of the important variables for the classification. We will use mutual information for representing the importance of a variable in the classification (discrimination) task.

C. Mutual Information

In the information theory [24], the Mutual Information (I) of two random variables is a quantity measuring the mutual dependence of these two variables. In our case, the objective is to find important descriptors as well as to quantify this importance. For that, we must compute, for $i = 1, 2, \ldots, p$, $I(C; X_i)$ the mutual information between the class variable C and each descriptor X_i.

In [25], authors demonstrate that the mutual information between a gaussian (normaly distributed) variable and a multinomial (discrete) variable can be computed as indicated by equation 2. For this equation, it is assumed that: C is a multinomial random variable with r possible values and a probability distribution given by $P(C = c) = P(c)$; X is a random variable with a normal density function of parameters μ and σ^2; X conditioned to $C = c$ follows a normal density function with parameters μ_c and $\sigma_c{}^2$.

$$I(X; C) = \frac{1}{2} \left[\log(\sigma^2) - \sum_{c=1}^{r} P(c) \log(\sigma_c^2) \right] \qquad (2)$$

This equation can be reduced on the form of the equation 3.

$$I(X; C) = \log(\sigma) - \sum_{c=1}^{r} P(c) \log(\sigma_c) \qquad (3)$$

So, the mutual information (I) can be computed for all the variables (descriptors) of the industrial system. The most important variables for the classification task will be those having an important I value.

We have seen a tool able to classify (CSNBN) and a tool able to quantify the discriminant information of each descriptor (mutual information I). Now, we will see how to use these two tools for the diagnosis of a fault in an industrial system.

III. PROCEDURES FOR FAULT DIAGNOSIS

A. Approach A

In the first approach, that we name approach A, we use a threshold in order to select the key variables of the system. After that, we use a CSNBN to classify a new faulty observation. The principle of this approach is described on the figure 4.
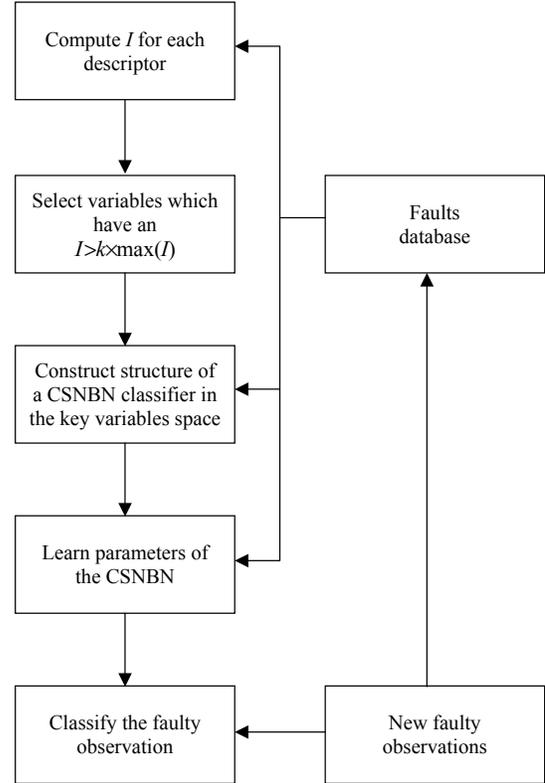

Fig. 4.   Procedure of the approach A

Of course, like each data-driven technique, this procedure requires a database, and more particularly a faults database. This database has to contain some examples of each fault. So, for each recorded fault, we must have the type of the fault and the value of the different variables at the instant of the fault. At this time, mutual information between the class node C (variable coding the type of fault) and each descriptor X_i (variable of the process) can be computed. So, we can sort the variables in decreasing order of mutual information. It is the step of the variables selection. For this step, a threshold value τ is used. This threshold value must be fixed at $k \times \max(I(X_i; C))$ with $0 < k < 1$. Each variable with mutual information less than τ is eliminated. So, we exclusively keep the variables giving important informations for the class discrimination. Then, we have to construct our classifier. We

could construct a naïve bayesian network (bayes classifier). But, as we explained in the previous section, this classifier makes the assumption that all descriptors are statistically independent, and we know that, in industrial processes, a lot of important variables can be highly correlated. So, on the faults database, we construct a CSNBN classifier using only the key variables. After the construction of the CSNBN, the parameters of this classifier are estimated from the faults database with the MLE method. Then, we can classify a new faulty observation (observation that the fault detection procedure gives us). At this moment, the diagnosis of the industrial system is made. If engineers give the confirmation of the diagnosis, the faulty observation and his class are added to the faults database.

This computational method is extremely fast, and so, can be easily adapted for a real-time diagnosis. But, its major drawback is the choice of the k coefficient used for the computation of the threshold τ. Indeed, according to the value of τ we will take the decision to add a variable in the model or not. So, the fact that τ depends of an arbitrary coefficient k can result in quite sensible decision for the classification. In order to improve the diagnosis, we will see an other approach where no arbitrary coefficient is needed.

B. Approach B

This second approach, approach B, sorts the variables in decreasing order of $I(X_i; C)$. After that, we will iteratively add variables to the CSNBN and evaluate the misclassification rate. First we will observe that the misclassification rate decreases by adding some variables into the model. But once a certain number of variables is overpassed, the misclassification rate starts to increase. If the misclassification rate increases at step i, the best model will be the one with the $(i-1)$ first informative variables. So, we obtain the following algorithm:

1) Compute $I(X_i; C)$ for all X_i
2) Sort X_i in decreasing order of $I(X_i; C)$
3) Initialize structure S to a Condensed Semi Naïve Bayesian Network (CSNBN) structure with a unique joint node containing the first informative variable
4) Evaluate this classifier through the estimated classified percentage
5) do {
6) Add the next informative variable to the joint node
7) Evaluate this classifier through the estimated classified percentage
8) }until No option improves the inducted classifier.

Of course, the evaluation of classifiers is done only on the training dataset. For this evaluation procedure, we apply a well known technique: the m-fold cross validation [26]. In the m-fold cross validation, the training dataset is divided into m subsets. So, for each m, one of the m subsets is used as the testing set, and the m-1 other subsets are put together to form the training set. Then the average error across all m trials is computed [27]. The entirely procedure of the approach B is given on the figure 5.
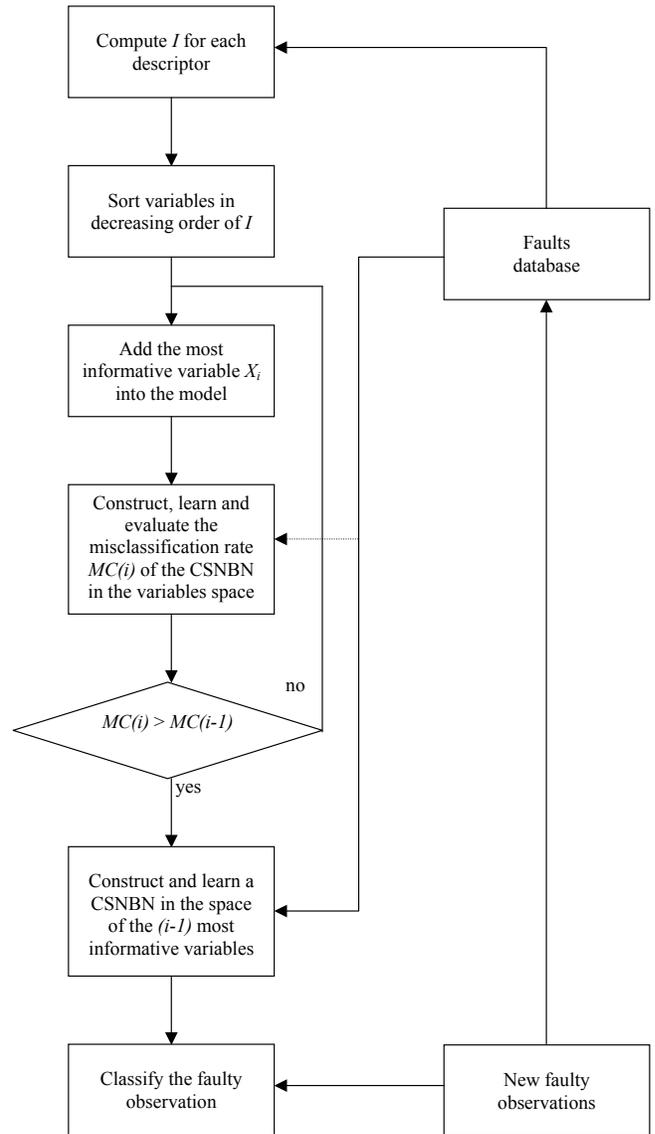

Fig. 5.   Procedure of the approach B

Now, we will see an application of these two approaches on a benchmark problem: the Tennessee Eastman Process (figure 6).

IV. APPLICATION TO THE TEP

A. Presentation of the TEP

The Tennessee Eastman Process (TEP) is a chemical process. It is not a real process but a simulation of a process that was created by the Eastman Chemical Company to provide a realistic industrial process in order to evaluate process control and monitoring methods. Article of Downs and Vogel [28] entirely describes this process. Authors also give the Fortran code of the simulation of the process. Ricker [29] has implemented the simulation on Matlab. The TEP is composed of five major operation units: a reactor, a condenser, a compressor, a stripper and a separator. Four gaseous reactant A, C, D, E and an inert B are fed to the reactor where
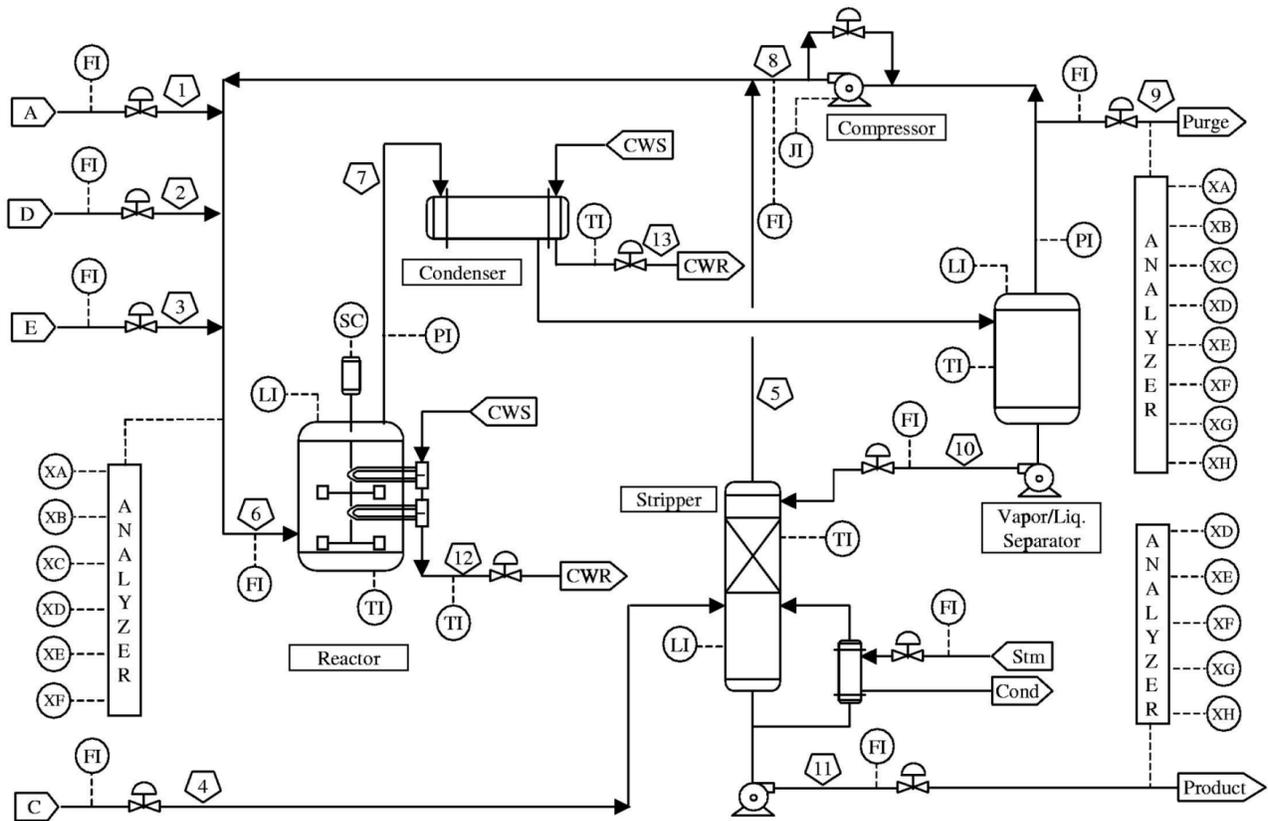
Fig. 6.   Process flowsheet of the TEP

the liquid products F, G and H are formed. This process has 12 input variables and 41 output variables. The TEP has 20 types of identified faults. So, this process is ideal to test monitoring methods. But, it is also a benchmark problem for control techniques because it is open-loop unstable. Many articles (authors) present the TEP and test their approaches on it. For example, in fault detection, we can cite [9], [30], [31]. Some fault diagnosis techniques have also been tested on the TEP [2], [32], [33], [34] with the plant-wide control structure recommended in Lyman and Georgakis [35]. In [32], [33], [34], authors focus on only 3 types of fault and give the datasets they used. For this reason, we will take the same data that in these articles and compare our approaches to those of the others.

As we said, we have taken into account 3 types of faults: fault 4, 9 and 11 (see table I). These three types of fault are good representations of overlapping data and so, are not easy to classify.

As indicated on the table I, each type of fault have 2 datasets: a training sample and a testing sample, containing respectively 480 and 800 observations.

We have tested our 2 approaches (approach A and B) on the Tennessee Eastman Process. We precise that all computations have been made on Matlab with the BNT (BayesNet Toolbox) developed by Murphy [36]. We also precise that the two approaches presented in the third section save a new faulty observation in the database. But, for correct

Class	Fault type	Train data	Test data
1	Fault 4: step change in the reactor cooling water inlet temperature	480	800
2	Fault 9: random variation in D feed temperature	480	800
3	Fault 11: random variation in the reactor cooling water inlet temperature	480	800

TABLE I

DESCRIPTION OF FAULT DATASETS

comparison with the results of other authors, we keep in the database just the 480 observations by fault.

B. Approach A on the TEP

Firstly, the mutual information is computed for all variables. Results of this computation are given in the figure 7.

We can visually see that the variables 9 and 51 are important variables to discriminate the different types of fault. If we fix the k value to 0.25, τ is equal to 0.27. For a τ value of 0.27, only variables 9 and 51 are selected. Chiang in [33] has obtained the same conclusion with different methods (contribution charts, discriminant partial least square, genetic algorithms combined with Fisher discriminant analysis). So, as explained in the section III-A, we will use the CSNBN on
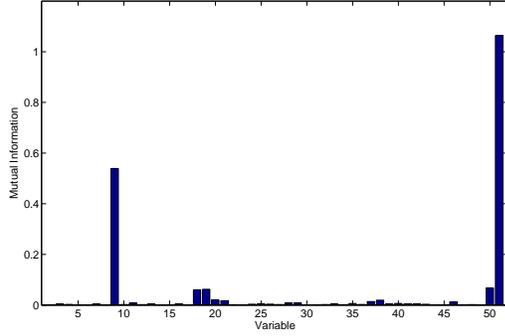
Fig. 7.   Mutual Information for the TEP application

the space of the key variables (9 and 51). Then, we can learn parameters of the CSNBN from the training dataset. Finally, we can classify the test sample with this classifier (2400 new observations to classify). The results of this classification are given by the confusion matrix of the table II.

Class	Fault 4	Fault 9	Fault 11	Total
Fault 4	796	0	37	833
Fault 9	0	775	75	850
Fault 11	4	25	688	717
Total	800	800	800	2400

TABLE II

CONFUSION MATRIX OF APPROACH A ON THE TEP DATA

Then, we compute the misclassification rate (percentage of observations that are not well diagnosis): 5.87%.

C. Approach B on the TEP

Now, we will apply approach B on the TEP application. As in the approach A, we first compute I for all variables and sort variables in decreasing order of I. After that, we iteratively add variables to the model and compute misclassification rate of a 10-fold cross validation on the training dataset. Result of these misclassification rates are given in table III.

Variable(s) in the model	Misclassification rate
51	12.5
51, 9	6.46
51, 9, 50	6.04
51, 9, 50, 19	5.90
51, 9, 50, 19, 18	**5.62**
51, 9, 50, 19, 18, 20	5.70

TABLE III

MISCLASSIFICATIONS OF STEPS OF APPROACH B

On this table III, we can see that the best result (on training dataset) is obtained for the selection of the 5 first informative variables (variables 51, 9, 50, 19 and 18). So, we construct a CSNBN in the space of these determined key variables and learn the parameters of the CSNBN with MLE. Finally, we can classify the 2400 observations of the testing dataset.

The results of this classification can be seen on the confusion matrix of the table IV.

Class	Fault 4	Fault 9	Fault 11	Total
Fault 4	788	0	31	819
Fault 9	0	778	74	852
Fault 11	12	22	695	729
Total	800	800	800	2400

TABLE IV

CONFUSION MATRIX OF APPROACH B ON THE TEP DATA

Based on this table, we can compute the misclassification rate. Its value is 5.79%. So, this approach seems to give better result than the approach A. We can now analyze our results and compare them to those of other authors.

D. Analysis and comparison of results

We will compare the results of approaches A and B to those of other methods on the same data (see table V). We notify that the results for the methods FDA (Fisher Discriminant Analysis), SVM (Support Vector Machines) and PSVM (Proximal Support Vector Machines) are extracted from [33], and the results for the ISVM (Independent Support Vector Machines) from [34]. In these articles, the classification was made in the space of the variables 9 and 51. In the table V, we have added the misclassification rate of the CSNBN in the space of all the variables (53 variables) $CSNBN_{all}$. We have also computed the misclassification rate that would be obtained by the use of a Naïve Bayesian Network, firstly in the space of all the variables (NBN_{all}), secondly in the space described by the variables 9 and 51 ($NBN_{9,51}$) and finally in the same space that the result of approach B ($NBN_{9,18,19,50,51}$).

Method	Misclassification rate
NBN_{all}	20.83%
$CSNBN_{all}$	18.83%
$FDA_{9,51}$	16%
$PSVM_{9,51}$	6.0%
$SVM_{9,51}$	6.5%
$ISVM_{9,51}$	6.0%
$NBN_{9,51}$	7.79%
$CSNBN_{9,51}$ (App. A)	5.87%
$NBN_{9,18,19,50,51}$	13.12%
$CSNBN_{9,18,19,50,51}$ (App. B)	**5.79%**

TABLE V

RESULTS OF THE APPLICATION

We can see that the classification methods in the selected variables space give better results than in the space of all the variables. Second interesting remark is the fact that on

the same space (variables 9 and 51), the best classifier is the CSNBN. Indeed, CSNBN of approach A outperforms all other classifiers (FDA, NBN and SVM based classifiers). The fact that CSNBN outperforms FDA is not surprising because FDA is a linear technique while CSNBN is a quadratic one. More, it is also logic than CSNBN outperforms NBN because NBN do not take into account the correlation between variable 9 and variable 51 (correlation of 0.42). Although, CSNBN requires quite simple computation, its results are quite similar to SVM based techniques (SVM, PSVM, ISVM), which are techniques requiring more computational potential. Finally, we can see that the best method is the approach B. Indeed, this method gives the best result of misclassification rate for this example of the TEP. But, it will be interesting to have misclassification rates of others techniques in the same space of variables (9, 18, 19, 50, 51).

Approaches A and B give good results on this example because fault classes are normally distributed (ellipsoidal shapes). Results of the CSNBN would not be so efficient if the data were not normal. But, an interesting way for taking into account not normally distributed variables is the gaussian mixture model which, on the based of few gaussian distributions, can represent any type of distribution. It will be a good answer to the problem of normally distributed data assumption. More, it is easy to represent a gaussian mixture model with a bayesian network.

Another way of interest is the diagnosis of a not identified fault type (no historical data on this fault type). Indeed, the major drawback of our procedure is the selection of the variables able to discriminate identified faults. But, if a new type of fault appeared, not discriminable on the selected variables, the procedure randomly attributes to this fault an identified type.

V. CONCLUSION AND OUTLOOKS

The main interest of this article is the presentation of two new procedures for fault diagnosis of industrial processes. These two procedures use a faults database to compute the mutual information between each descriptor and the fault type variable. In the first one (approach A), a threshold on the value of this mutual information permits to select important variables for discrimination of the different types of fault. The classification is then made with CSNBN in the space described by the selected variables. In the second approach (approach B), the informative variables are iteratively added to the model and the misclassification rates are estimated with a 10-fold cross validation. The iteration stops as soon as the estimated misclassification rate increases. the performances of these approaches were tested on a concrete example: the Tennessee Eastman Process. Results of these methods are good and outperform some previous results of other published methods.

Some outlooks of these procedures will be application of similar techniques, based on bayesian networks, in order to improve the fault diagnosis (cases of a non identified fault, case of non normally distributed data), but also in order to study in which way the fault detection step can be made with a bayesian network. The final goal is to have on the same procedure the fault detection and the fault diagnosis.

REFERENCES

[1] D. C. Montgomery, *Introduction to Statistical Quality Control, Third Edition*. John Wiley and Sons, 1997.
[2] L. H. Chiang, E. L. Russell, and R. D. Braatz, *Fault detection and diagnosis in industrial systems*. New York: Springer-Verlag, 2001.
[3] W. A. Shewhart, *Economic control of quality of manufactured product*. New York : D. Van Nostrand Co., 1931.
[4] S. W. Roberts, "Control chart tests based on geometric moving averages," *Technometrics*, vol. 1, pp. 239–250, Aot 1959.
[5] H. Hotelling, "Multivariate quality control," *Techniques of Statistical Analysis*, vol. , pp. 111–184, 1947.
[6] J. Westerhuis, S. Gurden, and A. Smilde, "Standardized q-statistic for improved sensitivity in the monitoring of residuals in mspc," *Journal of Chemometrics*, vol. 14, no. 4, pp. 335–349, 2000.
[7] E. J. Jackson, "Multivariate quality control," *Communication Statistics - Theory and Methods*, vol. 14, pp. 2657 – 2688, 1985.
[8] B. R. Bakshi, "Multiscale PCA with application to multivariate statistical process monitoring," *AIChE Journal*, vol. 44, pp. 1596–1610, 1998.
[9] M. Kano, K. Nagao, S. Hasebe, I. Hashimoto, H. Ohno, R. Strauss, and B. Bakshi, "Comparison of multivariate statistical process monitoring methods with applications to the eastman challenge problem," *Computers and Chemical Engineering*, vol. 26, no. 2, pp. 161–174, 2002.
[10] R. L. Mason, N. D. Tracy, and J. C. Young, "Decomposition of T^2 for multivariate control chart interpretation," *Journal of Quality Technology*, vol. 27, no. 2, pp. 99–108, 1995.
[11] R. Mason, N. Tracy, and J. Young, "A practical approach for interpreting multivariate t^2 control chart signals," *Journal of Quality Technology*, vol. 29, no. 4, pp. 396–406, 1997.
[12] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann Publishers, 1988.
[13] E. Charniak, "Bayesian networks without tears," *AI Magazine*, vol. 12, no. 4, pp. 50–63, 1991.
[14] F. V. Jensen, *An introduction to Bayesian Networks*. Taylor and Francis, London, United Kingdom, 1996.
[15] P. Langley, W. Iba, and K. Thompson, "An analysis of bayesian classifiers," in *National Conference on Artificial Intelligence*, 1992.
[16] N. Friedman, D. Geiger, and M. Goldszmidt, "Bayesian network classifiers," *Machine Learning*, vol. 29, no. 2-3, pp. 131–163, 1997.
[17] P. Domingos and M. J. Pazzani, "Beyond independence: Conditions for the optimality of the simple bayesian classifier," in *International Conference on Machine Learning*, 1996.
[18] I. Inza, P. Larranaga, B. Sierra, R. Etxeberria, J. Lozano, and J. Pena, "Representing the behaviour of supervised classification learning algorithms by bayesian networks," *Pattern Recognition Letters*, vol. 20, no. 11-13, pp. 1201–1209, 1999.
[19] Y. Yang and G. I. Webb, "Discretization for naive-bayes learning: Managing discretization bias and variance," Tech. Rep. 131, School of Computer Science and Software Engineering, Monash University,, 2003.
[20] C. Chow and C. Liu, "Approximating discrete probability distributions with dependence trees," *Information Theory, IEEE Transactions on*, vol. 14, no. 3, pp. 462–467, 1968.
[21] M. Sahami, "Learning limited dependence bayesian classifiers," in *Second International Conference on Knowledge Discovery in Databases*, 1996.
[22] D. Geiger and D. Heckerman, "Knowledge representation and inference in similarity networks and bayesian multinets," *Artificial Intelligence*, vol. 82, no. 1-2, pp. 45–74, 1996.
[23] I. Kononenko, "Semi-naive bayesian classifier," in *EWSL-91: Proceedings of the European working session on learning on Machine learning*, pp. 206–219, 1991.
[24] C. E. Shannon, "A mathematical theory of communication," *Bell Sys. Tech. J.*, vol. 27, pp. 379–423, 623–656, 1948.
[25] A. Perez, P. Larranaga, and I. Inza, "Supervised classification with conditional gaussian networks: Increasing the structure complexity from naive bayes," *International Journal of Approximate Reasoning*, vol. 43, pp. 1–25, 2006.
[26] T. M. Cover, *Learning in pattern recognition*. NY: Methodologies of Pattern Recognition, s. watanabe (ed.) ed., 1969.

[27] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification 2nd edition*. Wiley, 2001.

[28] J. Downs and E. Vogel, "Plant-wide industrial process control problem," *Computers and Chemical Engineering*, vol. 17, no. 3, pp. 245–255, 1993.

[29] N. Ricker, "Decentralized control of the tennessee eastman challenge process," *Journal of Process Control*, vol. 6, no. 4, pp. 205–221, 1996.

[30] J.-M. Lee, C. Yoo, and I.-B. Lee, "Statistical monitoring of dynamic processes based on dynamic independent component analysis," *Chemical Engineering Science*, vol. 59, no. 14, pp. 2995–3006, 2004.

[31] U. Kruger, Y. Zhou, and G. Irwin, "Improved principal component monitoring of large-scale processes," *Journal of Process Control*, vol. 14, no. 8, pp. 879–888, 2004.

[32] L. Chiang, E. Russell, and R. Braatz, "Fault diagnosis in chemical processes using fisher discriminant analysis, discriminant partial least squares, and principal component analysis," *Chemometrics and Intelligent Laboratory Systems*, vol. 50, no. 2, pp. 243–252, 2000.

[33] L. Chiang, M. Kotanchek, and A. Kordon, "Fault diagnosis based on fisher discriminant analysis and support vector machines," *Computers and Chemical Engineering*, vol. 28, no. 8, pp. 1389–1401, 2004.

[34] A. Kulkarni, V. Jayaraman, and B. Kulkarni, "Knowledge incorporated support vector machines to detect faults in tennessee eastman process," *Computers and Chemical Engineering*, vol. 29, no. 10, pp. 2128–2133, 2005.

[35] P. Lyman and C. Georgakis, "Plant-wide control of the tennessee eastman problem," *Computers and Chemical Engineering*, vol. 19, no. 3, pp. 321–331, 1995.

[36] K. P. Murphy, "The bayes net toolbox for matlab," in *In Computing Science and Statistics : Proceedings of Interface*, 2001.