Model– and Knowledge–based Fault Detection and Diagnosis of Gas Transmission Networks

Franz Perschl and Günther Schmidt
Division for Automatic Control Engineering (LSR)
Technische Universität München
D-80290 München, Germany

Abstract **This paper describes an expert system for online fault detection and diagnosis of gas transmission networks, combining model– and knowledge–based methods. It consists of a set of hierarchically structured components which include signal processing, Luenberger–type state observation, rule–based knowledge processing as well as an advanced user interface. The diagnosis system was tested with real measurement data from a medium sized gas distribution network. Its real–time capability and effectiveness for basic fault detection purposes was demonstrated by industrial applications.**

I. INTRODUCTION

Modern transmission networks for natural or product gases are highly automated systems. Sensors gather information on the network's states and computers control the network via actuators. The operator or dispatcher acts as a supervisor of the total system and takes control actions when faults occur [1]. With modern networks growing more and more complex, the operator's job becomes more and more difficult. Additional responsibilities are put on the operator by safety and economic requirements.

Introduction of a diagnosis system may support the operator in his task, particularly in abnormal situations. One goal of the diagnosis system is an early detection of fault symptoms and the search for possible technical causes behind them. Fault causes considered here are measurement failures, as for example caused by instrument failures and drifts, plant failures such as leaks and pipeline blockages as well as operator faults. By automatic supervision of the operator's control actions, the risk of causing new faults can be greatly reduced, particularly in stress situations. Furthermore, a diagnosis system may help to enhance network performance with respect to economy, reliability and availability.

The diagnosis system described in this article is designed for real–time operation. For this reason, special requirements exist particularly for its knowledge–based parts. In order to apply the diagnosis system to different gas transmission networks, the system is highly flexible, configurable and based on standard hardware and software components.

Similar investigations in real–time diagnosis systems are known from the field of electric power systems [2],[3]. Other papers concentrate mainly on certain aspects of a diagnosis system, such as state and parameter estimation [4]. In the field of gas transmission and distribution networks only a few papers and applications are reported [1] in the literature.

II. SYSTEM STRUCTURE

As shown in Fig. 1, the diagnosis system can be divided into four main components. All necessary input data is provided by the Supervisory Control And Data Acquisition (SCADA) system installed in the gas distribution network. These consist mainly of analog pressure and flow measurements $\underline{y}$, binary status signals $\underline{y}_z$ as well as analog and binary control signals $\underline{u}$. $\underline{u}$ includes signals automatically generated by the control system as well as by the operator.

The *signal-oriented component* processes the signals and detects abnormal deviations such as outliers, etc. Methods used are comparatively simple and provide fault symptoms of measurement failures. Symptoms $\underline{y}'_s$ and processed measurement data $\underline{y}'$, $\underline{u}$ are transfered to the *model-based component*.

This level incorporates two state observers. Observer 1 reconstructs not directly available network states, while observer 2 generates estimates for the measurements assuming that no fault has occured.

This component generates fault symptoms for slowly evolving measurement failures and plant failures such as leaks, pipeline or filter blockages and valve errors. Symptoms invoke the *knowledge-based diagnosis component*. This component transforms fault symptoms, which on their own have only small significance for the operator, into understandable messages. At the same time information about network state is compressed and redundancy is reduced.

Supervision of operator activities is carried out con-

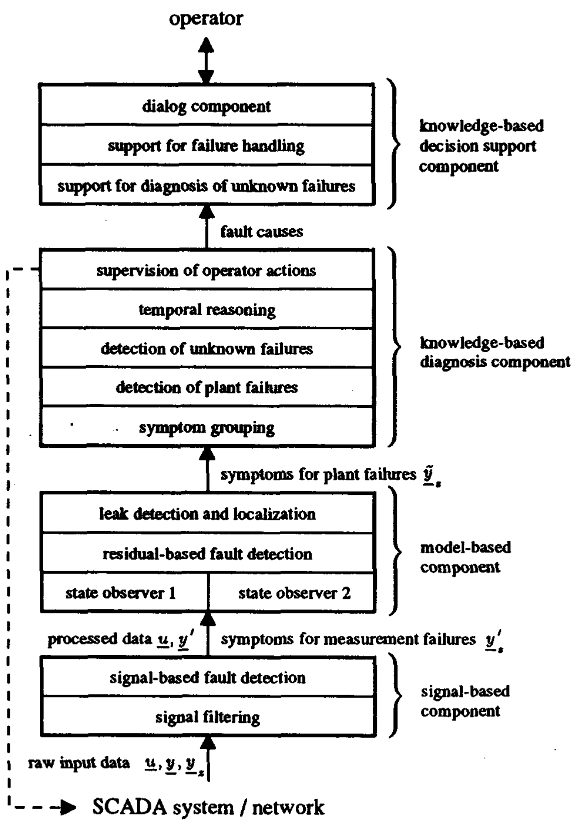

Figure 1: Basic structure of diagnosis system.

currently. Particularly, safety relevant activities receive special attention.

The diagnosis component provides as a result causes for various faults. In some cases a detailed diagnosis will need further information not available from the SCADA system. This information will be retrieved from the operator in the first layer of the ***knowledge-based decision support component.*** Based on this information, diagnosis results will be refined and the operator will be assisted with respect to the removal of the recognized faults and failures. This task is achieved through an advanced dialog component.

Details of the various components are discussed in the following sections.

III. SIGNAL-BASED COMPONENT

Signals, delivered by the SCADA system are filtered to remove noise which is caused by noisy measurements and quantization distortion. Next, certain well-known signal-oriented fault detection methods are applied to detect measurement faults.

A measurement fault in this context is defined as a fault, which causes loss or falsification of a signal on its way from the signal source (sensor) to its destination (SCADA system). In this definition the exact physical position or effect which causes the fault, is irrelevant. Usually there is not enough information available to determine the proper fault location.

Only smart or intelligent sensor transmitters, which incorporate certain fault detection methods, can generate some kind of status signal y_z. With this signal available, a more detailed diagnosis is possible on this level.

As Fig. 2 demonstrates, each signal y_i, associated by an optional status signal y_{zi} is processed separately. This approach has the advantage, that the user can choose individual filter and fault detection procedures for each signal.

Other advantages may be that this stage of the diagnosis system can be easily distributed on parallel processors for performance reasons. Time consuming computations can be implemented for example on digital signal processors and results can be read by a simple dummy procedure. Signal processing is often performed by the SCADA system itself, so that only a few signals need to be processed in the diagnosis system. The mechanisms developed in the system are flexible enough to support all the above mentioned cases.

Currently only digital low pass filters are used for noise filtering and smoothing of quantization errors. However, more sophisticated algorithms may be applied. It is even possible to transform a signal by using, for example, a Fast Fourier Transform, if required for diagnosis purposes.

For fault detection, limit and trend checking is used. If needed, statistical fault detection methods can be incorporated. In rare cases redundant measurements of the same signal are available, for example y_{21} and y_{22} in Fig. 2. They can be combined by plausibility checks into one comprehensive signal (y_2').

The output of the signal-based component are filtered

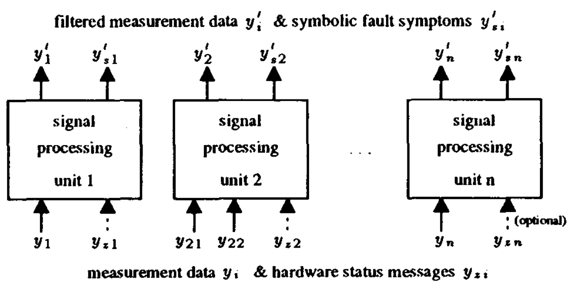

Figure 2: Signal processing activity.

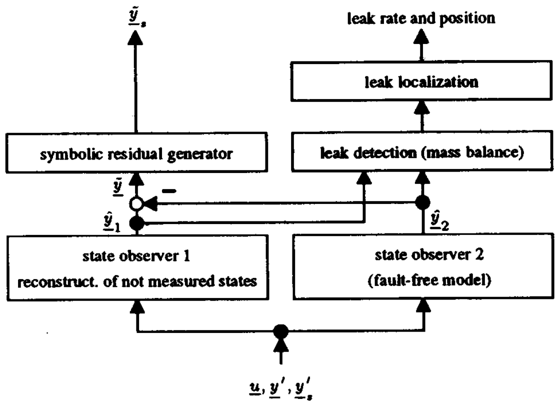

Figure 3: Structure of model-based component.

signals $\underline{y}'$ and symbolic fault symptoms $\underline{y}'_s$, like "signal OK", "signal MISSING", etc. For some minor faults, it is possible to substitute a corrupted signal by past or estimated values. Filtered signals and symptoms are transfered to the following component.

IV. MODEL-BASED FAULT DETECTION AND DIAGNOSIS COMPONENT

The purpose of the model-based fault detection component is the computation of residuals by use of dynamic process models. The residuals $\tilde{\underline{y}}$ can be considered as criteria for plant failures. Other model-based procedures make use of the estimated values for leak detection and leak localization. The basic structure of this component is shown in Fig 3.

Within the framework of two Luenberger-type state observers a high order network state model is incorporated. Because of economic reasons, only a small number of sensors is installed in a network. Therefore, state observer 1 reconstructs not directly measured signals $\hat{\underline{y}}_1$. This observer is tuned so that it can rapidly follow any changes in the input signals.

State observer 2 on the other hand is only weakly coupled to the input signals and follows any changes slowly. This is possible because of the high quality of the network model used which has been demonstrated in various industrial applications in the past [5]. The estimated measurements $\hat{\underline{y}}_2$, describe the dynamics of the fault free network.

An additional task of the model-based component is the transformation of the numerical residuals $\tilde{\underline{y}} = \hat{\underline{y}}_1 - \hat{\underline{y}}_2$ into symbolic fault symptoms $\tilde{\underline{y}}_s$, i.e. understandable terms for the knowledge-based diagnosis component. Symbolic fault symptoms in this context have values like "pressure HIGH", "pressure OK" or "pressure LOW". Generation of symbolic symptoms is performed by comparing the residuals with predefined limit values. If a residual is near the limit or oscillates around a limit, a symptom would be switched on and off. To avoid confusion of the subsequent knowledge-based components by switching a fuzzy evaluation of residuals, as proposed in [6] proves to be useful.

As an example of fault detection on this level, let us consider the pressure profile of a long pipeleg shown in Fig. 4. A to E represent measurement locations, in the following called nodes.

A dust filter is located between nodes B and C. The bold line in Fig. 4 represents the pressure profile under normal conditions. This profile is calculated by observer 2.

In case of *filter clogging* the dashed pressure profile would be measured. This profile is calculated by observer 1. The corresponding residuals $\tilde{\underline{y}}$ are translated into symbolic values HIGH and LOW (pressure) as shown in the figure.

Leak detection is another module of the model-based component. The algorithm works as follows: For every time step t_i both state observers compute current gas supply Q_s, offtake flow rates Q_d and pressures of the nodes in the network. The total gas flow into the network can be determined by $Q_{in}(t_i) = Q_s(t_i) - Q_d(t_i)$. For two consecutive time steps t_1 and t_2 the mass balance $\Delta m_1 = (Q_{in}(t_2) - Q_{in}(t_1)) * (t_2 - t_1)$ is computed. In addition, the total gas mass stored in the network $m_{st}(t_i)$ can be calculated from pressure values at time steps t_1 and t_2. Based on this data, a second mass balance $\Delta m_2 = m_{st}(t_2) - m_{st}(t_1)$ is determined.

The difference $\Delta m_{diff} = \Delta m_1 - \Delta m_2$ is called *short term mass balance*. It should be zero when no leak exists. In case of a leak however, the value of Δm_{diff} is a direct measure of the leak rate.

The sum of Δm_{diff} over time is the so called *long term*

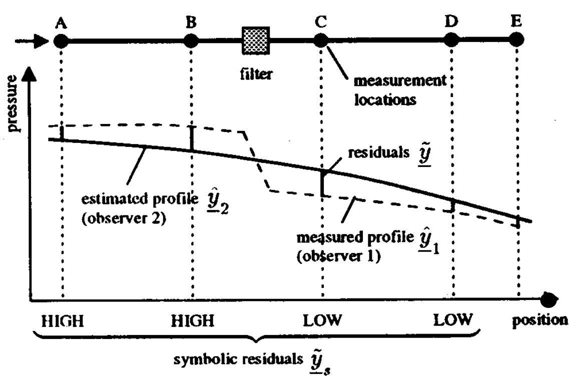

Figure 4: Fault symptom pattern for filter clogging.

mass balance, i.e. $\Delta m_{sum} = \Sigma\Delta m_{diff}$. Fig. 5 shows typical developments of both mass balances over time. Because of modeling and estimation errors, the current values of Δm_{diff} and Δm_{sum} cannot be applied directly as a leak criterion. Averages of two time windows are used as a substitute.

A leak is detected, when the difference between the averages *mk* and *ml* or the difference between the gradients of both regression lines *sk* and *sl* is greater than a predefined limit at current time t_{act}.

While the short term mass balance Δm_{diff} provides a reliable indication of bigger leak rates, the long term mass balance indicates smaller leak rates.

When applied to small sub–networks, the same procedure allows a rough approximation of *leak location*. Since division into sub–networks is restricted by the structure of the process model used in the state observers, exact localization is not possible.

Current research is concerned with the development of a new method for leak localization avoiding these restrictions. With this method more precise localization will be possible.

Symbolic fault symptoms $\underline{\tilde{y}}_s$, as well as leak rate and position are results of the model–based fault detection component.

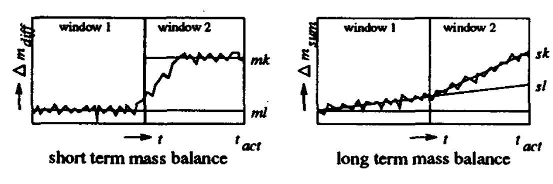

Figure 5: Example of mass balances considered for leak detection.

V. KNOWLEDGE-BASED DIAGNOSIS COMPONENT

This component makes use of the symbolic fault symptoms $\underline{\tilde{y}}_s$ to identify plant failures like pipeline and filter blockages, leaks and wrong valve positions. In order to save computing time, it is helpful to group fault symptoms in a network with respect to their local neighbourhood.

A. Symptom grouping

Fig. 6 shows an example. If two or more fault symptoms appear at neighbouring nodes in the network, it may be assumed that a direct physical relationship exists between them, and that the cause for all symptoms of one group is a single failure. On the other hand, if two or more fault symptoms in a network are separated by one or more measurement locations (nodes) without fault symptoms, it is assumed that no causal relationship exists between them.

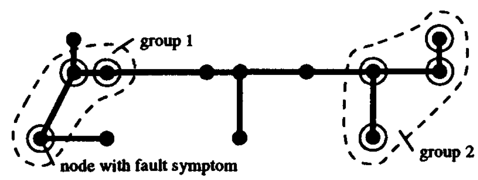

Figure 6: Symptom grouping in a network.

The problem of grouping can easily be solved by means of a graph theoretical approach. A recursive algorithm, based on a binary representation of faults in a network graph, was implemented. Its computing time is short, even for large gas transmission networks.

All subsequent diagnosis steps are based on symptom groups. Most of the diagnosis operations make use of search algorithms, which search for the existence of a given fault pattern in the network. These algorithms show their best performance, when working with symptom groups, representing small sub-networks,

B. Detection of plant failures

Before the search algorithm can be started, fault patterns must be defined: Considering again the example in Fig. 4, the following conclusion can be drawn: *If* a HIGH pressure symptom *and* a LOW symptom can be found at neighbouring nodes *and* a filter is located between both nodes, *then* this filter is probably clogged. In this case the algorithm searches for neighbouring HIGH and LOW symptoms in a symptom group.

Similar fault patterns can be derived for other failure types. Most fault situations are handled properly by this type of fault pattern search. Some special cases, particularly failures close to supply or offtake nodes, need the definition of more complex fault patterns in order to elaborate a proper diagnosis. In these cases, both pressure and flow profiles are used simultaneously to generate appropriate fault patterns.

Summing up, detection of plant failures is achieved by coding each fault pattern in a set of rules. The rules together with the mentioned search algorithm are applied to every symptom group detected. If a fault pattern matches the fault symptoms in the groups, then an appropriate message with the diagnosis result is generated.

C. Detection of unknown failures

Sometimes fault situations occur, where no predefined fault pattern matches. In this case limited support can be given by special rules, which may draw certain conclusions on possible fault causes based on the evolution of a fault situation over time and space. This method is often successful in localizing the fault origin.

To date, the diagnosis system is capable of detecting abruptly arising failures. However, there are certain failures, with slowly developing effects, for example, drifting sensor signals or spreading of fault situations around the network because of transient gas flow.

D. Temporal reasoning

To handle this kind of problem, all fault situations and the detected fault causes are stored for every time step. When new fault situations occur, the stored situations are retrieved and compared to the current situation. If the faults are identical or at least similar, then a causal relationship between old and new fault situation may exist with some probability . With this method, slowly evolving fault situations can be handled properly. Stored fault situations are also used for analysis of unknown failures mentioned earlier.

A history factor between 0 and 1 is assigned to each stored situation. Its value is increased when the associated fault situation occurs repeatedly and it is decreased when the fault situation disappears. A stored situation is deleted when the history factor is below a certain limit.

E. Supervision of operator actions

The knowledge-based diagnosis component deals also with supervision of operator activities. There are certain control actions which are not allowed in certain network states. For example, it may be extremely dangerous to open a closed valve in a high pressure gas transmission network if the pressure difference between inlet and outlet is higher than a certain limit. Opening of the valve would cause a high velocity of gas resulting in high frictional energy causing severe damage or even destruction of the valve.

To avoid this type of situation, the diagnosis system includes special rules which take care of the control actions. If an operator action is considered to be unadmissable, a message is generated and the action will not be carried out until the operator confirms his decision and overrides the warning of the diagnosis system.

VI. KNOWLEDGE-BASED DECISION SUPPORT COMPONENT

As mentioned in the last section unknown failures may appear from time to time. In this case the diagnosis system needs more information than available from the SCADA system to come up with a correct diagnosis. Some of the missing information can be obtained through a dialog with the operator.

A similar problem arises when more than one fault cause for the same situation is found. This problem may also be due to missing information.

When all ambiguities are resolved through a dialog with the operator, the next level will invoke a module which supports the operator in removing detected faults. This can be achieved by offering a dialog with a predefined sequence of required control actions to remove the fault. At the same time the actions are supervised by the diagnosis component so that new faults can be avoided.

For this and other reasons, the development of a user-friendly and ergonomic dialog component proves to be very important. It must provide an easy to understand presentation of diagnosis results combined with an easy to use data input facility.

The decision support component is still in a preliminary state. Knowledge acquisition causes a major problem since only the operator has the knowledge on how to handle certain fault situations.

Some details of the user interface are reported in the next section.

VII. IMPLEMENTATION ISSUES

A major part of the development and experimental evaluation of the fault detection and diagnosis system was accomplished using data collected in real gas transmission networks. Parts of the described diagnosis system such as leak detection and localization were tested in an industrial gas distribution network.

Fig. 7 shows the integration of the diagnosis system into a modern SCADA system. The diagnosis system can be connected to the system bus and uses process data available in the SCADA system's database.

The diagnosis system is currently implemented on a

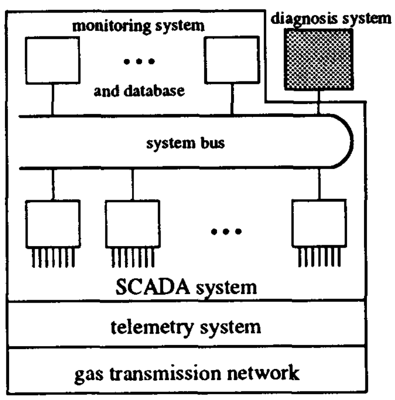

Figure 7: Implementation of the diagnosis system as part of a SCADA system.

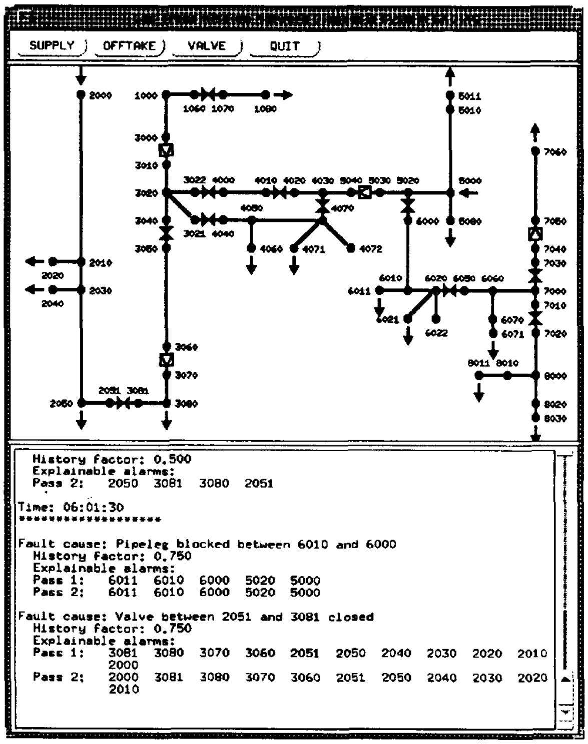

Figure 8: Hardcopy of operator's interface.

UNIX work station. It consists of two processes: the diagnosis system kernel and the user interface. The diagnosis system kernel receives its inputs either from a network simulator or from a file which contains real measurement data. Diagnosis results are visualized through the user interface.

The user interface is implemented in C and uses X-Windows (Open Look). The diagnosis system kernel is also programmed in C and the rule-based part in Pamela-C, a fast forward chaining expert system tool. Both observers, incorporated in the kernel as well as the simulator are written in FORTRAN. They are based on the well-known programs GANBEO and GANESI [5].

For communication between the processes, standard UNIX interprocess communication techniques based on TCP/IP are used. One complete cycle of simulation, diagnosis and visualization for the gas transmission network, shown in the upper part of a hardcopy of the operator's interface in Fig. 8 takes less than 1 sec on a SUN Sparc 1+ work station. For comparison: in real-time, one measurement updating cycle is 15 sec, as forced by the data aquisition system.

The examined network consists of 61 nodes with 11 valve stations, 2 main supply stations and 15 offtake nodes. The total pipeline length is about 170 km.

VIII. CONCLUSIONS

The gas network diagnosis system described in this paper is capable of detecting most basic measurement failures, plant failures and operator faults. This is accomplished by use of a combination of signal-based, model-based and knowledge-based techniques.

The implemented prototype demonstrated that the diagnosis system can be applied online, as part of a standard SCADA system, which monitors and controls a gas transmission network.

Improvements, particularly with respect to model-based leak detection and localization, are planned for the future. Further investigations will concentrate on the decision support component and the interaction with the dialog component as part of the user interface.

Because of the encouraging experiences with the diagnosis of gas transmission networks, the developed methods are currently adapted and generalized for use in chemical plants, with major emphasis on supervision and diagnosis of the material flow subprocesses.

ACKNOWLEDGMENTS

This work is partially supported by the "Bayerische Forschungsverbund Systemtechnik, BayFORSYS".

References

[1] G. Lappus, G. Schmidt, "Process monitoring and control of gas pipeline networks," *in S. G. Tzafestas, J. K. Pal (eds.), Real Time Microcomputer Control of Industrial Processes*, Kluwer Academic Publishers, 1990, pp. 397 - 433.

[2] K. Okuda, T. Fujii, N. Miyasaka, "Model based approach to the real-time fault diagnosis," *Proc. of the IMACS/SICE Int. Symp. on Robotics, Mechatronics and Manufacturing Systems*, Sep. 1992, Kobe, Japan, pp. 817 - 822.

[3] Y. Sekine, Y. Akimoto, M. Kunugi, C. Fukui, S. Fukui, "Fault diagnosis of power systems," *Proc. of the IEEE*, Vol. 80, No. 5, May 1992, pp. 673 - 683.

[4] P. M. Frank, "Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy - A survey and some new results," *Automatica*, Vol. 26, No. 3, 1990, pp. 459 - 474.

[5] G. Lappus, G. Schmidt, "Design and industrial application of a large scale observer system," *IFAC 10th World Congress on Automatic Control*, preprints Vol. 3, July 1987, Munich, Germany, pp. 117 - 122.

[6] J. Prock, "Computer-based sensor and process fault diagnosis in real time," *IFAC-Symp. on Control of Power Plants and Power Systems*, preprints, Vol. 1, March 1992, Munich, pp. 189 - 194.

[7] F. Barachini, "The evolution of PAMELA," *Expert Systems*, Vol. 8, No. 2, May 1991, pp. 87 - 98.