

Using the Process Schematic in Plant-wide Disturbance Analysis

S.Y. Yim^a, H. G. Ananthakumar^a, L. Benabbas^a, A. Horch^b, R. Drath^b, N.F. Thornhill^a

^aDepartment of E&E Engineering, University College London, London, UK, WC1E 7JE

^bABB Corporate Research Centre, Ladenburg, Germany, D-68526.

Abstract

This article describes how isolation and diagnosis of the root cause of a plant-wide disturbance is enhanced when process connectivity is considered alongside the results of data-driven analysis. A prototype software has been designed and implemented which, when given an electronic process schematic of a plant and results from a data-driven analysis of process measurements, allows the user to pose queries about the plant and to find root causes of plant-wide disturbances. The plant topology information is written in XML according to the Computer Aided Engineering Exchange (CAEX) schema.

Keywords: Fault diagnosis, plantwide oscillation; plant topology; root cause; XML.

1. Introduction

Methods for data-driven, signal-based analysis have been developed recently for finding root causes of plant-wide disturbances using measurements from routine process operations [Ruel and Gerry, 1998; Thornhill *et al.*, 2003a; Xia and Howell, 2003]. Several authors have observed, however, that data-driven analysis is enhanced if a qualitative model is used as well to capture the fundamental causal relationships of a process [Chiang and Braatz, 2003; Lee *et al.*, 2003]. The challenge is to represent causal information in electronic form and to manipulate it to draw conclusions.

Object-oriented representations of processes are becoming available using computer aided engineering tools. The plant topology can now be exported into a vendor independent and XML-based data format, giving a portable text file that describes all relevant equipments, their properties and directional connections between them [Fedai and Drath, 2005]. The Standard is IEC/PAS 62424, 2005, *Computer Aided Engineering Exchange* (CAEX). It specifies an XML schema. ISO-15926-7 is a similar standard.

A prototype tool called CAEX Plant Analyser that links a CAEX description with a report from a data-driven analysis is reported in this article. The features are (i) capture of process topology using CAEX, (ii) parsing and manipulation of the description, (iii) linkage of plant description and results from data-driven analysis, and (iv) logical tools to give root cause diagnosis and process insights.

Section 2 of the paper describes the background and places the work in context. Section 3 introduces the CAEX Plant Analyser and section 4 presents a case study.

2. Background and context

Reviews by Venkatasubramanian *et al.*, [2003a, 2003b] discussed detection, isolation and diagnosis of faults in chemical processes, classifying the available methods into quantitative and qualitative model-based methods and quantitative and qualitative process history based methods. This paper concerns a hybrid system using a qualitative models and quantitative process history.

The signed digraph (SDG) has been widely used as a causal qualitative model. Maurya *et. al.*, [2003, 2004] gave a comprehensive review of graph-based approaches for fault diagnosis of chemical process systems and showed how to develop SDGs from a system of differential-algebraic equations. Quantitative process history based methods involve signal analysis of process data to find the root cause of a disturbance. They include oscillation detection, linear and non-linear time series analysis [Ruel and Gerry, 1998; Xia and Howell, 2003; Thornhill *et. al.*, 2003b; Choudhury *et. al.*, 2004; Zang and Howell 2005]. These papers do not, however, exploit the physical connections between the measurements. A knowledge of process topology (connectivity) enhances the diagnosis, for example it shows which loops might disturb one another.

Work which has combined qualitative modeling and signal analysis includes Chiang and Braatz [2003] and Leung and Romagnoli [2002] who integrated multivariate statistical analysis with a causal map of a process to help in the diagnosis of faults, while Lee *et.al.*, [2003] also combined SDGs with multivariate statistical analysis. It is clear that signal analysis is enhanced by the capture and integration of cause and effect information from a process schematic, a step which now can be automated by CAEX.

3. The CAEX Plant Analyser

3.1. Overview

An overview of CAEX Plant Analyser is presented in Figure 1. One input is the CAEX file which describes the items of equipment in the plant such as tanks, pipes, valves and instruments and how they are linked together physically and/or through electronic control signals. A physical link (or path) is a pipe carrying a flow of mass or energy, while a control link (or path) is a cable connecting a valve to a controller carrying an electronic signal. The data input file contains information about plant-wide disturbances, for instance the period of oscillation, intensity and regularity, the measurement points where it was detected and any non-linearity detected in the time trends. These are given as a text-based report from a tool such as ABB's Plant Disturbance Analyser (PDA) [Horch *et.al.* 2005].

3.2. The system components, programming and integration

The system architecture consists of five components. The purpose of the parsers (blocks 1 and 2 in Figure 1) is to read and deconstruct the XML file containing the CAEX description and the PDA results text file. The XML parsing leads to lists of items of equipment and their connections from which the algorithms of the reasoning engine (3) find physical and control paths in the plant. With additional results from signal-based analysis it also determines root causes for plant-wide disturbances. It can also check that there is a feasible propagation path between a candidate root cause and all the other locations in the plant where secondary disturbances have been detected. The user interface (5) makes it easy to present such queries and to read the answers.

Prolog was used to implement the functions of the reasoning engine and C# was used for the remaining parts of the system. Prolog exploits the rule based nature of the connectivity information and the procedural and object-oriented features of C# allow an

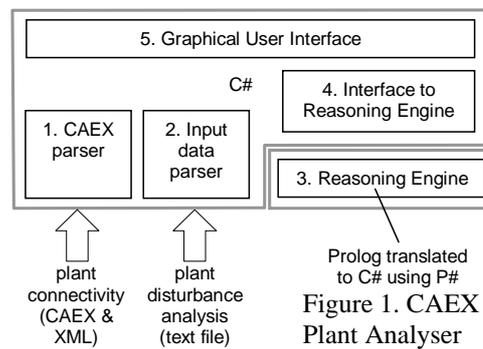


Figure 1. CAEX Plant Analyser

efficient parser and graphical application. The use of P# [Cook, 2004] to translate Prolog code to C# leads to an integrated system as a standard Windows application. Information is represented in different ways due to differing data types which exist in the two languages. The interface to the reasoning engine (block 4 in Figure 1) converts data that must flow between the components into the correct format.

3.3. Core technologies

Extensible Markup Language and CAEX: XML is an open standard developed by the XML Core Working Group, part of the World Wide Web consortium (Quin, 2005). XML uses plain text to represent structured data and uses tags to mark up the information. The structure of an XML document is defined in an XML schema which specifies which tags are allowed and what attributes they can have. A valid XML document must conform to the schema and the schema for representing an industrial plant is defined in the Computer Aided Engineering Exchange (CAEX) standard.

Microsoft .Net: Microsoft .Net allows applications to integrate and communicate easily. It consists of the Microsoft .Net Framework, development tools to create software for .Net, and client and server software. The .Net Framework is a run-time environment which allows programs built using different programming languages and running on different supported platforms to exchange data and work together.

The C# Programming Language and Prolog: The whole of the CAEX Plant Analyser Application except for the reasoning engine is written in the C# programming language. C# was developed by Microsoft and uses object-oriented features. It is supported by .Net and includes a number of classes that aid working with XML and the creation of Graphical User Interfaces. Prolog (PROgramming LOGic) is a declarative programming language which uses rules called predicates and facts to determine whether a query is true or false. For example, given that pipe1 is connected to valve1, and valve1 is connected to pipe2, and that a path is a list of connections, it is clear that there exists a path between pipe1 and pipe2. Prolog is able to determine that this is logically the case by applying its rules. The Prolog environment was P# which includes a subset of the Prolog programming language as a native implementation language for the .NET platform and interoperation is achieved by means of C# objects created from Prolog.

3.4. Distinctiveness of the approach

The distinctiveness of the CAEX approach should be considered relative to successful developments in SDGs e.g. Maurya *et. al.*, [2004]. A CAEX process representation has less predictive capability than an SDG because it is derived from a process schematic and not from a mathematical model. It gives binary (yes/no) answers to queries about presence or not of physical links and control paths, but not the signs of deviations from a nominal operating point. On the other hand, it is easy to generate an XML file describing the plant topology and the binary nature works well to verify or falsify hypotheses about disturbances and to infer root causes and propagation paths.

4. Case study

4.1. The application and plant object model

The application is from Thornhill *et. al.*, [2003a]. Figure 2 shows the process schematic in which the spots (placed by hand) indicate the locations where a plant-wide disturbance was detected. The larger spots show the locations of measurements whose time series were non-linear and therefore are candidates for the root cause. As described in Thornhill *et. al.*, [2003a], the key logical reasoning step is that the non-linearity reduces as the disturbance propagates away from its source. The task for CAEX Plant

Analyser is to locate the area of the process where the signals are most non-linear, and to use the connectivity information to identify the location furthest upstream as the most likely candidate for the root cause.

Loading the CAEX file from the *File* menu creates a plant object model and populates the *Elements* and *Internal Links* tabs of the GUI. The *Elements* tab shows the elements found in the topology and whether they are controllers, indicators or neither. The *Internal Links* tab shows the internal links and the elements connected by the link. Opening of a data file containing the PDA report causes the *Input Data* tab of the application to be populated with the table of data read from the file.

4.2. Queries

Analyses done in the *Perform Queries* tab (Figure 3) relate to the CAEX plant description. The following functions are available:

- Find a physical path between two elements in the topology;
- Find a control path between two elements in the topology;
- Find out what other elements are directly connected to another element;
- Find out whether an element is in a control loop.

For the purposes of visualization, the two physical paths found in Figure 3 are shown as heavy lines in Figure 2. The lower half of the window allows queries on individual elements in the topology. The *Root Cause* tab (Figure 4) is the part of CAEX Analyser where the process connectivity information is linked with the data-driven analysis. The panels in the display show the results of queries about:

- Working controllers (i.e. those achieving disturbance rejection);
- Non-linear controllers (those with non-linearity in PV or OP or SP);
- Non-linear indicators (those with non-linearity in the PV);
- Possible root cause controllers (those with nonlinearity in PV and OP, and which are upstream of other non-linear controllers and indicators).

The results show that Valve-003 (the level control of the decanter marked A) is the root cause of disturbance 1, which is the correct result. It has also identified the causes of two more oscillating disturbances which were discussed in Thornhill *et al.* (2003a).

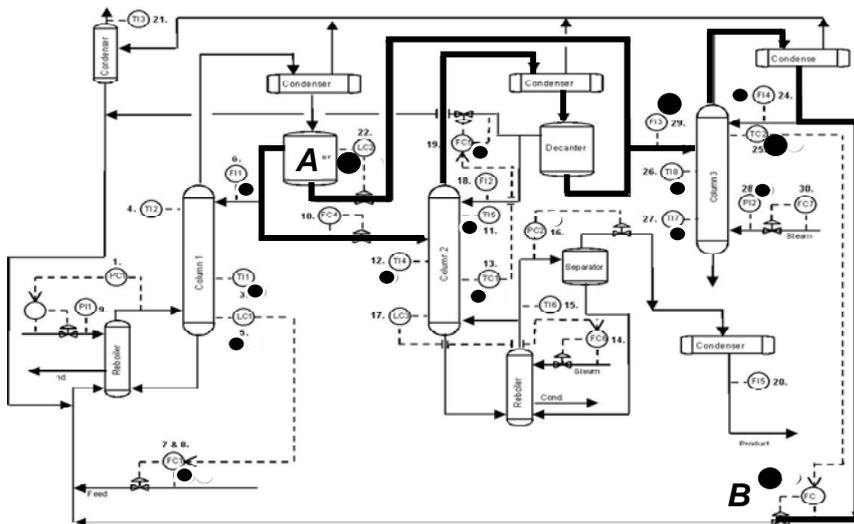


Figure 2 Process schematic (courtesy of J.W. Cox, Eastman Chemical Company). See sections 4.1 and 4.2 for explanations of the spots and heavy lines.

5. Critical evaluation and conclusion

Requirements for a plant-wide control loop performance analysis system have been given by Paulonis and Cox (2003) and Desborough and Miller (2002), including:

1. Facility-wide approaches including behaviour clustering;
2. Detection of the presence of one or more periodic oscillations;
3. Detection of non-periodic disturbances and plant upsets;
4. Determination of the locations of the various oscillations/disturbances;
5. Incorporation of process knowledge such as the role of each controller;
6. Automated model-free causal analysis to find the most likely root causes.

Requirements 1 to 4 have been substantially addressed by the quantitative process history based methods reviewed earlier. While it is possible to interpret the results from topics 1-4 by hand, the benefit of automation is that it speeds up the analysis by helping process control engineers to make queries and document the results. The CAEX Plant Analyser now demonstrates the feasibility of requirements 5 and 6. In summary, the CAEX Plant Analyser has given a new way forward to allow a user to pose queries about the plant and to gain insights into the root causes of plant-wide disturbances.

References

- Chiang, L.H., and Braatz, R.D., 2003, Process monitoring using causal map and multivariate statistics: fault detection and identification. *Chemometrics and Intelligent Laboratory Systems*, 65, 159-178.
- Choudhury, M.A.A.S., Shah, S.L., and Thornhill, N.F., 2004, Diagnosis of poor control loop performance using higher order statistics. *Automatica*, 40, 1719-1728.
- Cook, J.J., 2004, P#: A concurrent Prolog for the .NET Framework, *Software: Practice and Experience*, 34(9):815-845.
- Desborough, L., and Miller, R., 2002, Increasing customer value of industrial control performance monitoring – Honeywell's experience, *AIChE Symposium Series* No 326, 98, 153-186.
- Fedai, M., and Drath, R., 2005, CAEX – A neutral data exchange format for engineering data, *ATP International Automation Technology* 01/2005, 3, 43-51.
- Horch, A., Hegre, V., Hilmen K., Melbø, H., Benabbas, L., Pistikopoulos, E.N., Thornhill, N.F. and Bonavita, N., 2005, Root Cause - Computer-aided plant auditing made possible by successful university cooperation, *ABB Review* 2/2005, 44-48.
- Lee, G.B, Song, S.O., and Yoon, E.S., 2003, Multiple-fault diagnosis based on system decomposition and dynamic PLS, *Industrial & Engineering Chemistry Research*, 42, 6145-6154.
- Leung, D., and Romagnoli, J., 2002, An integrated mechanism for multivariate knowledge-based fault diagnosis, *Journal of Process Control*, 12, 15-26.
- Maurya, M.R., Rengaswamy, R., and Venkatasubramanian, V., 2003, A systematic framework for the development and analysis of signed digraphs for chemical processes. 1. Algorithms and analysis, *Industrial and Engineering Chemistry Research*, 42, 4789-4810.
- Maurya, M.R., Rengaswamy, R., and Venkatasubramanian, V., 2004, Application of signed digraphs-based analysis for fault diagnosis of chemical process flowsheets, *Engineering Applications of Artificial Intelligence*, 17, 501-518.
- Paulonis, M.A., and Cox, J.W., 2003, A practical approach for large-scale controller performance assessment, diagnosis, and improvement, *Journal of Process Control*, 13, 155-168.
- Quin, L., 2005, Extensible Markup Language (XML), On-line: <http://www.w3.org/XML/> Accessed: 7th Sept 2005.
- Ruel, M., and Gerry, J., 1998, Quebec quandary solved by Fourier transform, *Intech* (August), 53-55.
- Thornhill, N.F., Cox, J.W., and Paulonis, M., 2003a, Diagnosis of plant-wide oscillation through data-driven analysis and process understanding, *Control Engineering Practice*, 11, 1481-1490.
- Thornhill, N.F., Huang, B., and Zhang, H., 2003b, Detection of multiple oscillations in control loops. *Journal of Process Control*, 13, 91-100.
- Venkatasubramanian, V., Rengaswamy, R., and Kavuri, S.N., 2003a, A review of process fault detection and diagnosis Part II: Qualitative model and search strategies, *Computers and Chemical Engineering*, 27, 313-326.
- Venkatasubramanian, V., Rengaswamy, R., Kavuri, S.N., and Yin, K., 2003b, A review of process fault detection and diagnosis Part III: Process history based methods, *Computers and Chemical Engineering*, 27, 327-34.

Xia, C. and Howell, J., 2003, Loop status monitoring and fault localisation. *Journal of Process Control*. 13, 679-691.

Zang, X.Y., and Howell, J., 2005, Isolating the root cause of propagated oscillations in process plants. *International Journal of Adaptive Control Signal Processing*, 19, 247-265.

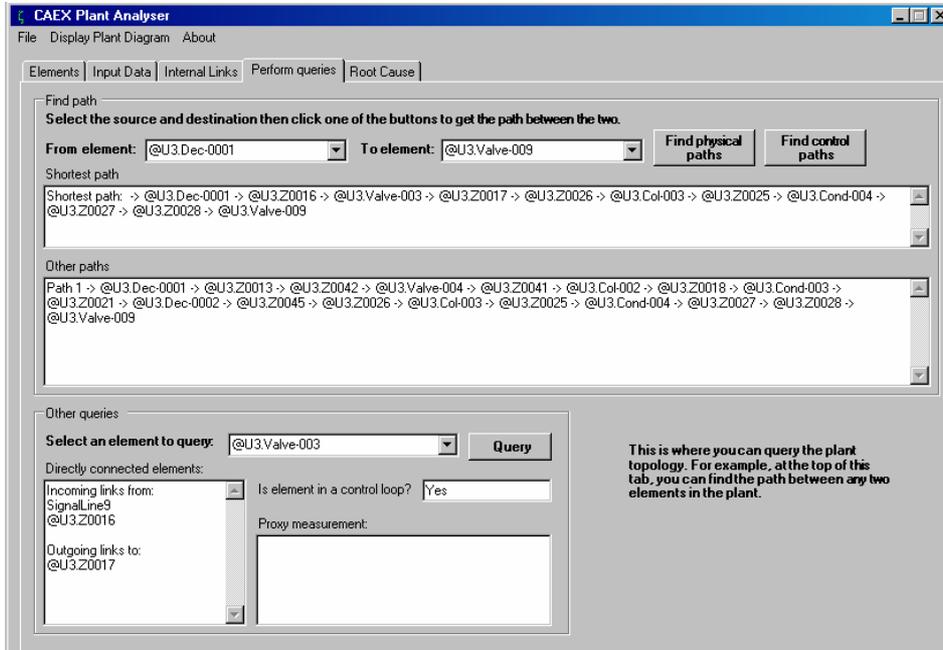


Figure 3. The *Perform Queries* tab. See section 4.2 for commentary.

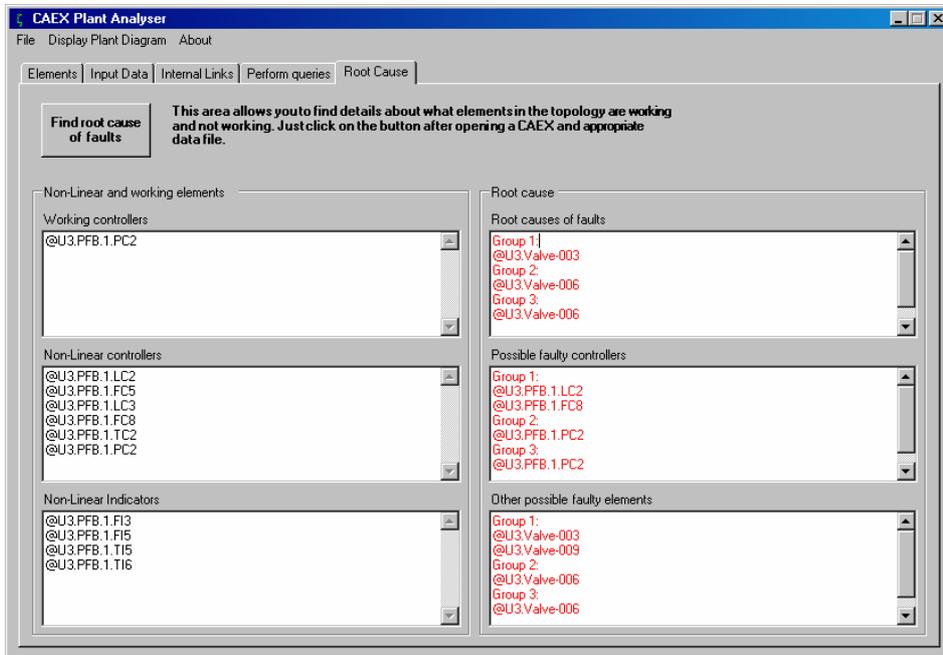


Figure 4. The *Root Cause* tab. See section 4.2 for commentary.