



Combining functional modeling and reasoning with on-line event analytics

Kirchhübel, Denis

Publication date:
2020

Document Version
Publisher's PDF, also known as Version of record

[Link back to DTU Orbit](#)

Citation (APA):
Kirchhübel, D. (2020). *Combining functional modeling and reasoning with on-line event analytics*. Technical University of Denmark.

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.



Technical
University of
Denmark

Denis Kirchhübel

Combining Functional Modelling and Reasoning with on-line Event Analytics

PhD Thesis, December 2019

Automation and Control
Department of Electrical Engineering

Combining Functional Modelling and Reasoning with on-line Event Analytics

Denis Kirchhübel

Technical University of Denmark
Kgs. Lyngby, December 2019

**Technical University of Denmark
Department of Electrical Engineering
Automation and Control (AUT)**

Elektrovej
Building 326
2800 Kongens Lyngby, Denmark
Phone +45 4525 3576
info@elektro.dtu.dk
www.aut.elektro.dtu.dk

Summary

Modern industrial plants rely heavily on automation to increase energy and economic efficiency, however, human operators are tasked with the supervision of the plant and have to intervene when abnormal situations occur. To determine whether and how to mitigate an abnormal situation, control rooms provide a large number of sensor and status information to the operators and alarms are intended to draw operators attention to situations that require immediate action. In reality, however, the amount of information provided to operators can actually reduce operators' focus and distract them from maintaining awareness of the situation. The most severe situations that overload operators with information are alarm floods, in which a large number of events is presented to the operator requiring immediate attention. As coping strategy operators tend to resort to treating symptoms as quickly as possible to reduce the number of alarms, when rationally analysing the whole set of occurring events and treating underlying root causes would be more effective. Identifying these root causes from the large amount of data available and providing operators with the context of how occurring alarms can reduce the risk of overloading operators and ensure safe and efficient plant operation.

This work contributes to an operator support approach based on a functional representation of the process flow and operating goals in Multilevel Flow Modelling (MFM). MFM is an established modelling methodology for operations and process knowledge that has been applied for process design and diagnosis in a variety of industrial contexts. The qualitative diagnosis in MFM facilitates the identification of all possible fault scenarios. Two major aspects are considered in this work: maintaining correct causal analysis based on all occurring alarms from the plant and ensuring that the MFM model used for diagnosis fits the current plant situation.

Toward the correct on-line analysis an improved causal reasoning system leveraging the connection of occurring alarms has been developed and shown to increase efficiency and reasoning speed. Approaches to ranking the identified root cause candidates have been defined to provide meaningful information to operators. By providing a ranking of the root causes, operators' focus can be directed toward distinguishing the most likely root causes and determining an efficient mitigation strategy.

The configuration and operating goals of a plant are frequently changed in accordance with pre-defined operating procedures. To ensure that the correct model is used for the causal reasoning, methods to link these operating procedures to MFM have been investigated. In an industrial setting operators tend to adapt operating procedures, based on their experience and out of execution efficiency and convenience. To account for these adaptations, a structured representation of operating procedures and method for validating the representation against operations logs have been proposed.

All proposed methods have been demonstrated on case studies with industrial relevance for the chemical or petrochemical industry. Combining the qualitative modelling and reasoning in MFM with the analysis of alarms and events from the control system in real time facilitates the contextual situation assessment necessary to support operators. With ongoing research into machine-learning based alarm generation and MFM based counter action planning, the proposed methods provide the core functionality for establishing a comprehensive operator support system, which can relieve operators from the repetitive task of filtering out relevant events and provide assistance for efficient mitigation of abnormal situations.

Resumé

Moderne industrianlæg er meget afhængige af automatisering for at øge energimæssig og økonomisk effektivitet, men menneskelige operatører har til opgave at føre tilsyn med anlægget og gribe ind, når der opstår unormale situationer. For at bestemme, hvorvidt og hvordan man kan afbøde en unormal situation, giver kontrolrum et stort antal sensor- og statusoplysninger til operatørerne, og alarmer er beregnet til at henvende operatørernes opmærksomhed på situationer, der kræver øjeblikkelig handling. I virkeligheden kan mængden af information, der gives til operatører, reducere operatørernes fokus og forhindre dem i at bevare et overblik over situationen. De mest alvorlige situationer, der overbelaster operatører med information, kaldes for alarmbyger. En alarmbyge opstår når et stort antal meldinger præsenteres for operatøren, der kræver øjeblikkelig opmærksomhed. I de mest graverende situationer har operatører en tendens til at ty til behandling af symptomer så hurtigt som muligt for at reducere antallet af alarmer, selvom det ville være mere effektivt at analysere helheden af forekommende medlinger og behandler underliggende hovedårsager. At identificere disse hovedårsager fra en stor mængde tilgængelige data og at forsyne operatører med konteksten af, hvordan forekommende alarmer hænger sammen, kan reducere risikoen for overbelastning af operatører og sikre effektiv og sikker anlægsdrift.

Dette arbejde bidrager med en tilgang til operatørsupport baseret på en funktionel repræsentation af procesfunktioner og driftsmålene i Multilevel Flow Modelling (MFM). MFM er en etableret modelleringsmetodik til repræsentation af drifts- og procesviden, der er anvendt til procesdesign og diagnose i en række industrielle områder. Den kvalitative diagnose i MFM letter identificeringen af alle mulige fejlscenarier i en given situation. To vigtige aspekter overvejes i dette arbejde: opretholdelse af korrekt årsagsanalyse baseret på alle forekommende alarmer fra anlægget og at sikre, at MFM-modellen, der bruges til diagnose, passer til den aktuelle anlægssituation.

Med henblik på den rigtige onlineanalyse udvikles et forbedret system for årsagsidentificering med udnyttelse af sammenhængen mellem forekommende alarmer. Systemet øger effektiviteten og analysehastigheden. Der defineres metoder til rangordning af de identificerede hovedårsagskandidater for at give meningsfulde oplysninger til operatørerne. Ved at rangordne hovedårsagerne kan operatørernes fokus rettes mod at skelne mellem de mest sandsynlige grundårsager og fastlæggelse af effektive modforholdsregler.

Konfiguration og driftsmål for et anlæg ændres ofte på grund under afvikling af foruddefinerede driftsprocedurer. For at sikre, at den korrekte model bruges til årsagsanalyse, bliver metoder til at knytte disse driftsprocedurer til MFM undersøgt. I industrielle omgivelser har operatører en tendens til at tilpasse driftsprocedurer, baseret på deres erfaring for at øge effektivitet og komfort. For at redegøre for disse tilpasninger bliver der foreslået en struktureret repræsentation af driftsprocedurer og

en metode til validering af repræsentationen mod data af anlægsdrift.

Alle de foreslåede metoder bliver demonstreret i casestudier af relevans for kemisk eller petrokemisk industri. Kombination af den kvalitative modellering og årsagsanalyse i MFM med analyse af alarmer og meldinger fra kontrolsystemet i realtid letter den kontekstuelle situationsvurdering, der er nødvendig for at støtte operatører. Sammen med igangværende forskning i maskinlæringsbaseret alarmgenerering og MFM-baseret planlægning af modforholdsregler udgør de foreslåede metoder kernefunktionaliteten til udvikling af et omfattende supportsystem for anlægsoperatørerne, der kan frigøre dem fra gentagne opgaver ved at filtrere relevante begivenheder og at yde hjælp til effektiv afhjælpning af unormale situationer.

Preface

This thesis was prepared at the department of Electrical Engineering at the Technical University of Denmark (DTU) in partial fulfilment of the requirements for acquiring a PhD degree. The PhD project was financed by the Danish Hydrocarbon Research and Technology Center (DHRTC) in the CTR-1 "Operations and Maintenance Technology" programme.

The supervisors were:

- Professor Ole Ravn (main supervisor), Department of Electrical Engineering, Automation and Control, DTU
- Professor Emeritus Morten Lind (co-supervisor), Department of Electrical Engineering, Automation and Control, DTU

The thesis consists of a summary report of the findings of the PhD project and the collection of articles submitted to peer reviewed scientific journals and published in conference proceedings during the project period 2016-2019.

Acknowledgements

*A human must turn information into intelligence
or knowledge.*

*We've tended to forget that no computer will
ever ask a new question.*

— Grace Hopper

I would like to thank my supervisors Professor Emeritus Morten Lind and Professor Ole Ravn for their support and guidance throughout the years leading up to this thesis. I am also deeply grateful to Assistant Professor Xinxin Zhang for the close collaboration. The "original" MFM discussion group with Professor Emeritus Sten Bay Jørgensen and Niels Jensen of Safepark provided a platform of inspiration and sparring on a broad range of perspectives related to functional modelling and process engineering.

The development focus in the project setting with the Danish Hydrocarbon Research and Technology Center, the collaboration and exchange with partners from Aalborg University Esbjerg, Eldor and Kairos Technology was a great stimulation. While it has not always been easy to reconcile the push for fast solutions with the academic ambition for thorough investigation it was a struggle worthwhile and a great source of motivation to see theoretical concepts come to live in a product. Within this setting a big thank you is due to Thomas Martini Jørgensen for his persistence and interest in cross-disciplinary collaboration.

Thanks to all colleagues at the group for Automation and Control (AUT) for a good work environment, entertaining and engaging discussions over lunch and coffee breaks, and their friendship. For the feedback, proofreading of papers and enduring my presentations, thank you – especially my fellow PhD students and colleagues on the MFM team, Emil Krabbe Nielsen, Adriana Zsurzsan, Robert Miklos, Christopher Reinartz, Dimitrios Papageorgiou, Ásgeir Hallgrímsson, and everyone else.

I want to express my gratitude to our supplementary family in DTU Volley for always being there. And not the least, I thank my parents and family for putting me on the path that got me this far, my wife for her unwavering support and being a great mother to our daughter.

Contents

Summary	i
Resumé	iii
Preface	v
Acknowledgements	vii
Contents	ix
1 Introduction	1
1.1 Background	1
1.2 Objectives	3
1.3 Structure of the Thesis	4
2 Summary of Main Contributions	5
3 State of the Art	9
3.1 Improved Operator Support	9
3.2 MFM based Operator Support	13
3.3 Summary	15
4 Reasoning System	17
4.1 Causal Inference	17
4.2 Inference Maintenance	18
4.3 Case Study	21
4.4 Summary	22
5 Analysing Models and Results	23
5.1 Multilevel Flow Models as causal graphs	23
5.2 Visualising inference results	25
5.3 Summary	28
6 Cause Ranking	29
6.1 Distance based Ranking	29
6.2 Bayesian Networks for Cause Ranking	31
6.3 Case study	34
6.4 Summary	36

7	Operating Modes and Procedures	37
7.1	Functional aspects of Operating Modes	38
7.2	Objective Related Modes in Multilevel Flow Modelling (MFM)	39
7.3	Modelling Redundant Components	42
7.4	Operating Procedures and Data	44
7.5	Industrial Case Study	48
7.6	Summary	52
8	Conclusion	53
8.1	Summary of the Project	53
8.2	Perspectives for Functional Modelling based Operators Support	54
	Publications	57
A	Dynamic Reasoning in Functional Models for Multiple Fault Di- agnosis submitted to Computers & Chemical Engineering	59
B	Combining Operations Documentation and Data to Diagnose Pro- cedure Execution conditionally accepted for Computers & Chemical En- gineering	75
C	Toward Comprehensive Decision Support Using Multilevel Flow Modeling presented at 5th IFAC ICONS, 2019	89
D	Generation of Signed Directed Graphs Using Functional Models presented at 5th IFAC ICONS, 2019	99
E	Identifying causality from alarm observations presented at ISOFIC, 2017	111
F	Generating Diagnostic Bayesian Networks from Qualitative Causal Models presented at 24th IEEE ETFA, 2019	119
G	Representing Operational Modes for Situation Awareness presented at 13th IFAC ACD, 2016	127
	Bibliography	141

CHAPTER 1

Introduction

Human operators are tasked with the supervisory control of most production plants in process industry. Control rooms are designed to provide operators with all essential information and remote control over the plant to ensure safe and efficient operations. However, established control room systems are prone to overload operators with unfiltered information, especially during severe abnormal situations. The work presented here contributes to a novel approach for operator support that aims at providing the operators with more context and less raw information. Within this framework methods for causal analysis, root cause ranking, considerations for operating procedures, and adaptation of the causal analysis are presented. In the following sections the background and motivation for the project are detailed, followed by the summary of research objectives of this project, and an outline of the thesis structure.

1.1 Background

Control rooms provide the central interface for human operators to interact with a processing plant. They allow operators to keep an overview of the entire plant from a relatively safe location through remotely accessible sensor information and actuators. Early control rooms hosted a large number of analogue gauges and dials as well as arrays of alarm indicators, with the operators tasked with interpreting and tracing the states of the plant. With an increase in digital and distributed control systems, instrumentation and remote control of components have become cheaper and easier to implement. Digital control solutions and advanced control methodologies increase efficiency and reliability of systems, but for operators they also significantly increase the amount of information available and the need for contextual analysis. Modern control rooms typically provide the operators with a representation of the plant structure or process flow attributed with sensor values or trends accompanied by an overview of discrete events generated by the distributed control system, such as alarms, automated adaptations, and self-diagnosis (Koffsky et al., 2013).

”[Alarms are] audible and/or visible means of indicating to the operator an equipment malfunction, process deviation, or abnormal condition requiring a timely response” (ISA, 2009). While alarms are the most important notification to the human operators with respect to safe operation of the plant, plants are inherently connected and a severe deviation is rarely contained within one component of the plant. Severe upsets are instead prone to propagate through the system leading to more and more alarms as connected components and sub-systems become affected by the deviation. If more than 10 alarms occur in a 10 minute time frame, it is deemed to dramatically impact the operators capability to take informed decisions and the situation is char-

acterised as an alarm flood (EEMUA, 2013). Operators constantly monitor the plant for deviations and abnormal events, analyse their observation and evaluate the overall situation to identify the need for action. If an action is required the operators identify a plan of action, execute the mitigation plan, and continue monitoring the success of the mitigation strategy. The majority of the operators' tasks lies in the assessment of the current situation, which is made increasingly difficult by the vast amount of information available in the control room. With a large amount of unfiltered information, situations tend to evolve, where operators start losing the overview and the ability to thoroughly analyse the situation. In these situations implemented actions focus on the most notable or justifiable situation, treating the symptoms rather than the actual cause of an abnormal situation. (Hollnagel, 2002)

To reduce the cognitive load on operators caused by a large number of irrelevant alarms, regulators as well as industry associations have defined guidelines for alarm system improvement and management (EEMUA, 2013; IEC, 2014; ISA, 2009). Alarm management encompasses a wide range of methods and approaches to improve the performance of alarm systems. The major part of established alarm management is concerned with removing unnecessary alarms and thoroughly analyse the relevance of alarms. While this kind of analysis can be incorporated in the design process, it requires a lot of time and expert resources to reiterate this process on an established plant (Soares et al., 2016). In addition to the removal or redesign of alarms, the dynamics of the system can be incorporated in the alarm configuration to only activate alarms in a relevant context (Beebe et al., 2013). However, alarm flood situations are unlikely to be suppressed by any of these approaches. Even well configured alarms will be triggered in a large number due to the connections throughout the plant. Therefore more analysis incorporating past data, operator knowledge and process design knowledge is being investigated. Many approaches propose analysing recorded incidents and available data from a long period of time, assuming that important situations have emerged before and could be recognised in the future. (Wang et al., 2016a)

This work contributes to an operator support framework based on a functional model of the plant's operating goals and physical flows, specifically Multilevel Flow Modelling (MFM). MFM has the potential to not only represent known fault situations but also diagnose generally possible scenarios. In current control rooms, the majority of operators' attention is focused on filtering information about the system and identifying the causal context of all individual process alarms and indicators. With an increase in independently presented events, human operators tend to lose the overall perspective and start treating events in isolation, disregarding the connection between these events. In these situations, operators will struggle to keep up with the incoming events, instead of identifying and executing an effective mitigation strategy. Providing a support system that identifies the common cause of occurring events and presents them in a coherent context, frees up operators' capacity to make effective, informed decisions. This operator support system is not intended to replace human operators, but to relieve the repetitive task of information processing and provide operators with a short list of likely situation analyses. The levels of autonomy

(Parasuraman et al., 2000) for operator support are described in Figure 1.1(a) for the current state of the industry and the goal of the presented framework. The system can then provide a complete analysis of tentative consequence for a selected scenario, support the operator’s decision, and guide mitigation strategies. The building blocks of the proposed framework are shown in Figure 1.1(b): The knowledge acquisition relies on validated models (Nielsen et al., 2018a) and an approach of assembling the plant model from validated library elements (Lind, 2017), to get meaningful results from such a diagnostic system. The prognostic planning to offer guidance on mitigation strategies is being developed further with the approaches outlined by e.g. Song and Gofuku (2017). The focus of the present work is on the real-time inference and aspects of the situation evaluation, combining the functional modelling and causal reasoning in MFM to analyse occurring alarms and events in a processing plant.

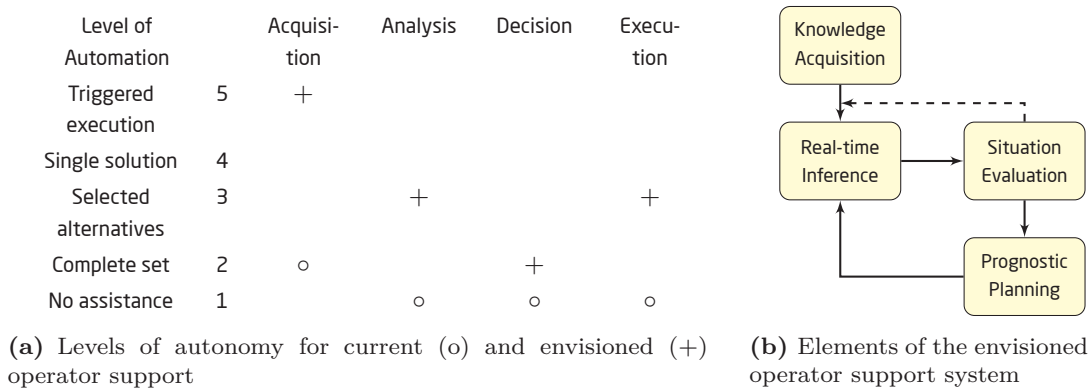


Figure 1.1: Levels and autonomy and elements of the envisioned operator support

1.2 Objectives

This thesis summarises the results of a research project funded by the Danish Hydrocarbon Research and Technology Center (DHRTC), which was conducted in a PhD program. The project setting provided a close collaboration with partners from petrochemical industry. The work was carried out at the Technical University of Denmark contributing expertise in utilising operations knowledge and MFM for operator support across industry domains.

Expanding on previous research into MFM based operator support, the goal is to adapt and apply the methodology to the scale of real industrial plants. The main objectives of this PhD project are to:

- compare the state of the art and best practise in decision support and alarm management with MFM based real-time applications, identifying the benefits

and open issues toward the application of MFM as foundation for a comprehensive alarm management and decision support tool.

- develop a method deducing plausible failure paths by combining the MFM reasoning with all observed alarms.
- identify actual failure paths to reduce the amount of information needed to enable situation assessment by human operators.
- link operating procedures and MFM inference for on-line adaptation of the model used for analysis.
- demonstrate of the proposed methods in industrial settings.

1.3 Structure of the Thesis

This thesis summarises the different aspects of the contributions as a collection of scientific articles, which were submitted to peer-reviewed journals and conferences throughout the course of the PhD study. The articles are appended at the end of the thesis and form the main reference for Chapters 3 to 7 of the body of this work.

After this introductory chapter, Chapter 2 gives an overview of the main contributions, briefly summarising the submitted articles and detailing additional unpublished aspects disseminated in this thesis. A review of the state of the art in alarm management and decision support, with a focus on knowledge representation and acquisition is given in Chapter 3. Previous implementations and on-line applications of MFM are described and briefly discussed in the context of the present research project.

Chapter 4 describes the inference system implemented to determine possible scenarios. A new approach to leveraging the connection between events in alarm flood situations for efficient causal inference is presented and the improvements are demonstrated in a case study.

The representation and analysis of MFM models and the inferred propagation results is described in Chapter 5. The extraction of an equivalent Signed Directed Graph (SDG) from an MFM model as well as the merit of different visualisations of the inference results are discussed.

To evaluate the current situation, approaches for ranking the most likely root cause scenarios are discussed in Chapter 6. The direct interpretation of the qualitative inference results as a connected graph, as well as approaches to generate diagnostic Bayesian Belief Network (BBN)s are summarised and compared in a case study.

Additionally, the consideration of operating procedures and modes is examined in Chapter 7. The link of operating modes and MFM models is examined for the entire system as well as redundant subsystems to facilitate adaptation of the underlying model used for inference. Furthermore, the representation and tracking of operating procedure execution is discussed and validated on an industrial study.

Chapter 8 concludes the thesis with an outline of the perspective and possibilities for the overall framework and the evaluation of the findings of this work.

CHAPTER 2

Summary of Main Contributions

An overview of the envisioned MFM based operator support system is described in Paper C, together with a summary of previous work based on MFM and a positioning of current research within the operator support framework. Within this project, two main aspects are investigated to support the operator support system: First, an efficient causal inference to reason about causes and consequences of observed events or alarms is developed to provide information about likely root causes. Second, methods to include changes to the plant configuration based on operating procedures are proposed to adapt the model used for reasoning and diagnose the correct execution.

In Paper A the proposed efficient inference system is described and a first approach to ranking the resulting root causes is suggested. A method to compare the underlying MFM model to qualitative SDG is described in Paper D, which allows the combination and comparison of established methods for inference and analysis. The combination of inference results for causal analysis is discussed in Paper E. In Paper F, additional methods to identifying the most likely root cause from the model or inference results based on BBN are examined.

In Paper G, an approach to explicitly model the succession of steps in an operating procedure in MFM is investigated. In contrast, a more versatile methodology to represent an operating procedures as automaton and validate the nominal execution against plant logs is presented in Paper B. Additional considerations how to represent the configuration of redundant systems in MFM have not been published before and are described in Section 7.3 of this thesis for the first time.

Journal Articles

- (A) D. Kirchhübel, M. Lind, and O. Ravn (2019b). “Dynamic Reasoning in Functional Models for Multiple Fault Diagnosis”. *Computers and Chemical Engineering*. submitted in April 2019

Qualitative process models, like MFM, provide an efficient representation for causal analysis. By simulating the propagation of faults in the system root causes and eventual consequences can be identified. This article presents an efficient approach to infer causes and consequences for multiple alarms based on a qualitative model in an improved reasoning system for on-line analysis. Root causes are identified by the dynamic reasoning about observed faults and a ranking of most likely root causes is proposed. The efficiency of the inference and ranking methods is finally demonstrated on an industry process.

- (B) D. Kirchhübel, M. Lind, and O. Ravn (2019a). “Combining Operations Documentation and Data to Diagnose Procedure Execution”. *Computers and Chemical Engineering*. accepted in Nov 2019 pending revision

An industrial plant can be operated in a variety of configurations, due to changing production goals or operating procedures. To ensure a correct diagnosis from a qualitative process model, the model needs to follow configuration changes in the monitored plant. Standard Operating Procedures detail when and how the plant configuration is to be changed and are established during plant design to ensure consistent and safe operation. Tracking the correct execution of a procedure is relevant to both, detecting errors during procedure execution and adapting diagnostic models. In this contribution, methods are described to represent documented procedures as automata and to validate the procedures against log files of the control system. A fast approach of detecting procedure executions and action sets associated with procedure steps is proposed and demonstrated in an industrial case study.

Conference Papers

- (C) D. Kirchhübel, M. Lind, and O. Ravn (2019c). “Toward Comprehensive Decision Support Using Multilevel Flow Modeling”. In: *5th IFAC Conference on Intelligent Control and Automation Sciences*. Belfast, UK: IFAC-PapersOnLine

Automating control rooms by incorporating design and operation knowledge about the systems can significantly improve operator efficacy. Intelligent support systems should reduce the amount of information and provide more context to the operators. The operators’ focus should be shifted from information acquisition to taking informed decisions about mitigation steps. This paper describes and positions the ambition for an MFM based intelligent human machine interface to support operator performance. A brief review of the development of MFM and its application to provide operators with decision support and situation awareness is given, focusing on implementations directly utilising the knowledge represented in MFM. Finally, current research efforts are related to the envisioned comprehensive operator support system.

- (D) C. C. Reinartz, D. Kirchhübel, O. Ravn, and M. Lind (2019). “Generation of Signed Directed Graphs Using Functional Models”. In: *5th IFAC Conference on Intelligent Control and Automation Sciences*. Belfast, UK: IFAC-PapersOnLine

While MFM has been designed for operator support and applied to industrial processes in chemical, petroleum and nuclear industry, it has not been directly related to other frameworks for plant-wide diagnosis. SDGs have been shown to be a viable method for plant-wide diagnosis that can incorporate both quantitative information about the process condition as well as qualitative information about the system topology and the functions of its components. However, their

range of application in industrial settings has been limited due to difficulties regarding the interpretation of results and consistent graph generation. This contribution addresses these issues by proposing an automated generation of SDGs of industrial processes in the chemical, petroleum and nuclear industries using MFM; a functional modelling method designed for operator support. The approach is demonstrated through a case study conducted on the Tennessee Eastman Process, showing that MFM can facilitate a consistent modelling process for SDGs.

- (E) D. Kirchhübel, X. Zhang, M. Lind, and O. Ravn (2017b). “Identifying causality from alarm observations”. In: *International Symposium on Future I&C for Nuclear Power Plants (ISOFIC) 2017*. Gyeongju, Korea, pp. 1–6

The application of MFM for root cause analysis based alarm grouping has been demonstrated and can be extended to reason about the direction of causality considering the entirety of the alarms present in the system for more comprehensive decision support. Combining the causal analysis from multiple alarms increases the efficiency of analysing connected alarms and provides a more complete diagnosis. This contribution presents the foundation for combining the cause and consequence propagation of multiple observations from the system based on an MFM model. The proposed logical reasoning matches actually observed alarms to the propagation analysis in MFM to distinguish plausible causes and consequences. This extended analysis results in causal paths from likely root causes to tentative consequences, providing a tool to not only identify but also rank the criticality of a large number of concurrent alarms in the system.

- (F) D. Kirchhübel and T. M. Jørgensen (2019). “Generating Diagnostic Bayesian Networks from Qualitative Causal Models”. In: *24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*. IEEE, pp. 1239–1242

Causal analysis based on qualitative models provides a comprehensive set of potential root causes. To properly diagnose a situation and deduce corrective actions, however, it is important to identify the most likely root cause. This paper investigates the translation of qualitative causal models and the root cause analysis into BBNs to utilise efficient tools for probability inference. The diagnosis results of a fault scenario of the Tennessee Eastman Process highlight the feasibility of the principle approach in order to leverage the potential of BBN.

- (G) D. Kirchhübel, M. Lind, and O. Ravn (Jan. 2017a). “Representing Operational Modes for Situation Awareness”. In: *13th European Workshop on Advanced Control and Diagnosis (ACD 2016)*. Vol. 783. 012055. Journal of Physics: Conference Series. DOI: 10.1088/1742-6596/783/1/012055

Intelligent operator support tools for complex industrial systems require knowledge about the overall system structure and behaviour. The desired behaviour of a complex system, however, changes due to operating procedures that require more than one physical and functional configuration. While functional modelling, like MFM, can reduce the complexity of plant-wide models for diagnosis, these changing configurations need to be considered in the model for a correct diagnosis and situation assessment. In this paper a consistent representation of possible configurations is deduced from the analysis of an exemplary start-up procedure by functional models. The proposed interpretation of the modelling concepts simplifies the functional modelling of distinct modes.

Unpublished Work

- Industrial systems frequently incorporate redundancy, typically due to economic or reliability concerns. Section 7.3 discusses the relation of MFM and redundant components. An extension to the means-end description of process objectives in MFM is suggested to incorporate the constraints of redundancy, similar to the resilience represented by voting OR gates in a fault tree.

CHAPTER 3

State of the Art

As described in Chapter 1, operators' tasks comprise data acquisition, situation analysis, decision making and counter-action planning. Alarm systems currently provide the main interface for operators to acquire information about the system, so a well maintained alarm system is essential for effective plant operation (Rothenberg, 2009). Additional systems to support operators have been proposed to aid the decision process by supporting the situation assessment, decision alternatives or mitigation strategies for a diagnosed situation. In this chapter an overview of research for improved operator support in general and more specifically different approaches based on the functional modelling framework of MFM is provided.

3.1 Improved Operator Support

Alarm management has been the subject of many improvement efforts especially in chemical and petrochemical industry. This reflects in the guidelines and standards defined for process industries, such as EEMUA (2013), IEC (2014), and ISA (2009). Hollifield and Habibi (2006) and Rothenberg (2009) have compiled overviews of the current best-practice in industry. However, the overview of scientific endeavours in the field of alarm systems published by Wang et al. (2016a) reveals that the issues related to alarm management are mostly researched independent of each other and rarely as a whole. In this section a categorisation of scientific work from the last years in topics of alarm management and decision support is given. Subsequently the most relevant methods for representing the required process knowledge are compared with respect to their applicability and the ways of generating the knowledge.

3.1.1 Alarm Management

As described by Rothenberg (2009) most aspects of alarm management are related to a thorough revision and scrutiny of the existing alarm system to ensure that all alarms actually entail some operator reaction and yield meaningful information. Different approaches to this problem are categorised in Figure 3.1 with considerations for offline analysis of the configured system, on-line grouping of alarms, and establishing logic constraints for filtering irrelevant alarms. The established approach of alarm rationalisation is mostly an offline process that is meant to reduce the amount of nuisance or irrelevant alarms by redesigning the alarms. One approach to this problem is reflected in the work of Charbonnier et al. (2014) and Soares et al. (2016), who present statistical analysis methods to identify groups or clusters of alarms that move together as a means of identifying redundant and possibly superfluous alarms since

they inform about the same failure state. Hu et al. (2017b) present a comprehensive framework that allows the analysis of overall alarm system performance. The framework provides intuitive visualisation and analysis of "worst-actor alarms" occurring most frequently and typically without direct implications for operations. To facilitate alarm rationalisation Hu et al. (2017b) implemented means to reduce chattering alarms by redesigning alarm limits and delays. Another branch of research concerned with nuisance or chattering alarms is the analysis of oscillations in the system. Established alarm redesign would introduce pre-processing of the process variables by dead-bands or counters to reduce this kind of alarms (Rothenberg, 2009). The work of Duan et al. (2014), Landman et al. (2014), and Yang et al. (2014) aim to identify badly tuned controllers or faulty actuators that lead to propagating oscillations in the system. These efforts can help improve the performance of the alarm system by reducing chattering alarms and are thus considered as offline redesign approaches in the context of alarm management.

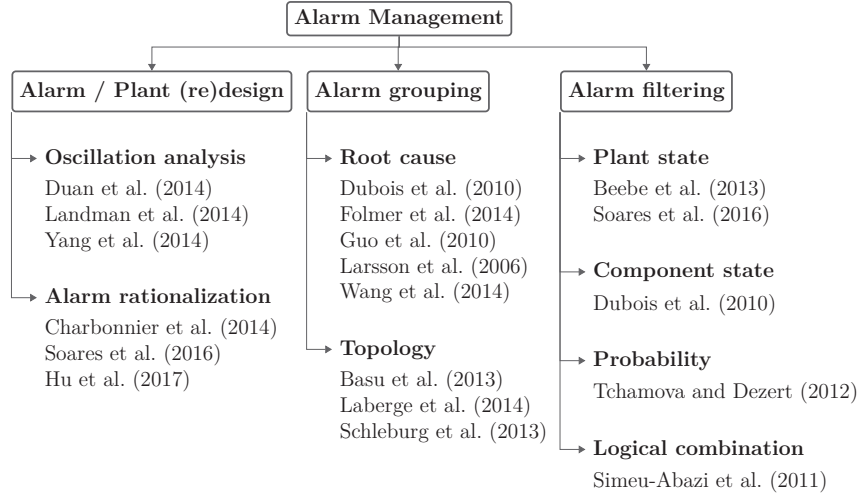


Figure 3.1: Research approaches to Alarm Management

As pointed out by Beebe et al. (2013) rationalisation and improvement of alarms does not alleviate the load on operators sufficiently in critical situations, e.g. when many correct alarms are triggered due to the propagation of abnormal states through the system. Enhanced alarm methods such as alarm suppression, i.e. filtering, and eclipsing or grouping are proposed to reduce the effective number of alarms the operator is confronted with in such situations of alarm floods (Rothenberg, 2009). For these methods it is relevant to consider the current configuration, i.e. the plant state or operation mode, to exclude floods of false alarms caused e.g. by equipment under maintenance (Beebe et al., 2013). Soares et al. (2016) outline the application of their clustering method to identify such operation modes, identifying changing clusters caused by changes to the plant configuration. In contrast, Dubois et al. (2010) propose to use an explicit state machine for each component to determine whether its

alarms are meaningful in a given situation. And Tchamova and Dezert (2012) propose a fusion of probabilities of different alarm states for a variable, while Simeu-Abazi et al. (2011) suggest defining fault trees with logical and temporal combinations of alarms that correspond to actual failures to only report those to the operator.

By filtering out false or irrelevant alarms, a part of the occurring alarm floods in a processing plant is eliminated. In situations of incidents, however, many true alarms can be triggered as symptoms of a fault propagating through interconnected parts of the system. Dubois et al. (2010), Folmer et al. (2014), Guo et al. (2010), Larsson et al. (2006), and Wang et al. (2014) apply root-cause analysis to hide a number of alarms related to a common root-cause and only present the operator with that initial failure. The root-cause analysis is based on a representation of process knowledge, such as fault trees (Dubois et al., 2010), temporal constraints (Folmer et al., 2014; Guo et al., 2010), multilevel flow models (Larsson et al., 2006) and Bayesian networks (Wang et al., 2014). Alternatively, alarm grouping by topological neighbourhood or processing unit has been proposed by Laberge et al. (2014) and Schlegel et al. (2013) respectively. In a similar fashion Basu et al. (2013) suggest a severity measure to highlight alarms with a large temporal or spatial extent as requiring the most urgent response from the operator.

3.1.2 Decision Support

Apart from improving conventional alarm systems, research into providing better support to operators has been published in the recent years. The research summarised in Figure 3.2 suggests incorporating automated methods to ease the operators' task of diagnosing the current plant state. Ideally, these systems are integrated with the operator interface for on-line support. However, many of the methods described here have not yet been demonstrated in real-time environments. Azam et al. (2014) propose to track the development of monitored process variables and determine when they will tentatively raise an alarm. Using connectivity information, the propagation of component faults to those process variables is interpreted to pinpoint components that are likely to fail soon (Azam et al., 2014). Zhu et al. (2016) propose the use of data based probability of critical alarms occurring in a sequence of alarms to predict their likelihood. The MFM based propagation analysis presented by Zhang (2015) is applicable to cause analysis as well as enabling the prediction of consequences based on the causal relations between currently triggered alarms and other parts of the plant.

While root-cause analysis can be used to hide all connected alarms, independent of the alarm system root cause analysis can also provide an additional diagnostic tool for the operator. Dubois et al. (2010) propose the use of fault-trees, and Natarajan and Srinivasan (2014) propose local fault-detection agents per component that are coordinated by a global analysis agent to generate a root-cause presentation for the operator. Bayesian networks have been widely proposed to determine the likelihood of different root cause scenarios (Cai et al., 2016; Hu et al., 2015; Nguyen et al., 2016).

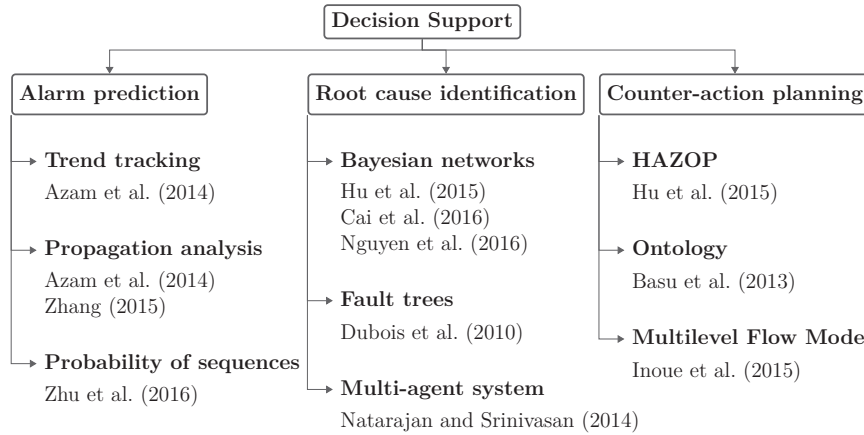


Figure 3.2: Research approaches to decision support in control rooms.

Based on the identified faults, guidance on mitigation or counter-action strategies can be provided to operators. Hu et al. (2015) base their approach on a Hazard and Operability (HAZOP) study of the system which are part of the design and verification of a plant and link failure symptoms to causes and actions for remedy. The diagnosed root cause can then be associated to the identified counter-actions from the HAZOP study. Basu et al. (2013) derive linked ontologies of alarm types and control actions on the assumption that alarms representing a similar fault require a similar remedy. Inoue et al. (2015) propose a method based on the adaptation of generic functional models to compensate for a given fault scenario providing operators with a step-by-step operating procedure.

3.1.3 Knowledge Representation

Performing advanced analysis of alarms and improving the performance of the operator interface require knowledge of the process. Figure 3.3 categorises the methods proposed for alarm management and operator support with regards to the knowledge representation. Process connectivity and causality between process variables and events are most commonly used in those methods, but a variety of approaches have been outlined to obtain the respective representation. Ontologies present an alternative representation. Ontologies are an established concept of artificial intelligence, where groups of similar properties are established to describe the relations between elements. For the analysis, ontologically defined agents (Natarajan and Srinivasan, 2014), as well as linked alarm and control action ontologies to characterise alarms (Basu et al., 2013) are suggested.

Considering fault states directly, fault trees establish a hierarchy of component or subsystem faults contributing to a fault in the system. These can be based on the combinations of possibly critical component states (Dubois et al., 2010) or integrate additional logic and timing constraints of the fault occurrences (Simeu-Abazi et al.,

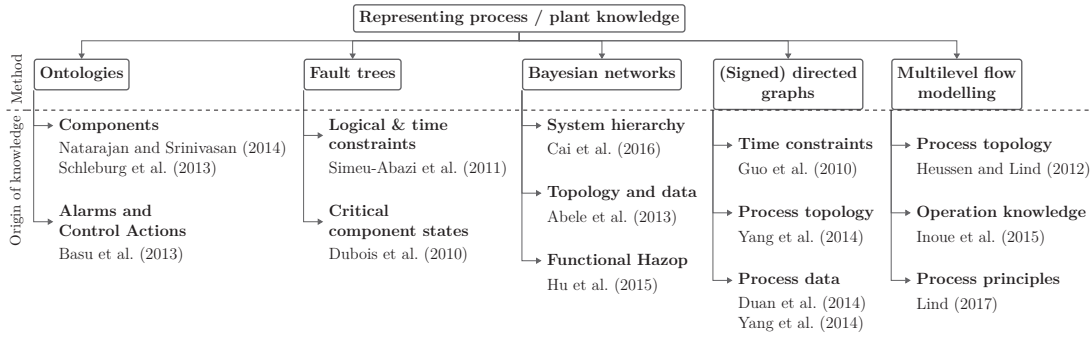


Figure 3.3: Summary of knowledge representation methods by source of knowledge.

2011). Bayesian Networks, or BBNs are widely used to determine the likelihood of a number of causally connected events. The structure of a BBN can for instance be derived from a hierarchy of component-wise BBNs (Cai et al., 2016), a combination of process topology and probabilities derived from data (Abele et al., 2013), or from the causal paths identified in a structured HAZOP study (Hu et al., 2015).

While SDG and MFM do not represent probability, they facilitate the representation of complex causality throughout the process. The causality in a SDG can be derived from different correlation analyses in data (Duan et al., 2014; Yang et al., 2014), temporal constraints based on the sequence of alarms (Guo et al., 2010), or process topology (Yang et al., 2014). With application in both alarm management (Larsson et al., 2006) as well as for decision support (Hu et al., 2015; Inoue et al., 2015; Zhang, 2015) MFM provides a versatile knowledge representation framework. Similar to SDG, MFM represents causality between system functions but provides an abstract formalisation connecting different process perspectives. A complete MFM model incorporates process topology and operation knowledge (Inoue et al., 2015). However, models based on process topology (Heussen and Lind, 2012), physical process principles, and the hierarchical combination of validated sub-models (Lind, 2017) have been shown to be sufficient for diagnosis.

3.2 MFM based Operator Support

As outlined in the previous section there is a large variety of knowledge representations applied to both alarm management and decision support. Out of the described representations MFM has been applied to the majority of aspects of operator support for on-line alarm management (Larsson et al., 2006) as well as for decision support such as fault scenario identification (Zhang, 2015) and counter-action planning (Inoue et al., 2015). While multiple approaches, like the BBN based diagnosis presented by Hu et al. (2015), are in fact based on MFM, this section focuses on approaches directly applying MFM for different aspects of operator support. The chronology of the respective publications is illustrated in Figure 3.4 and is also described in paper C.

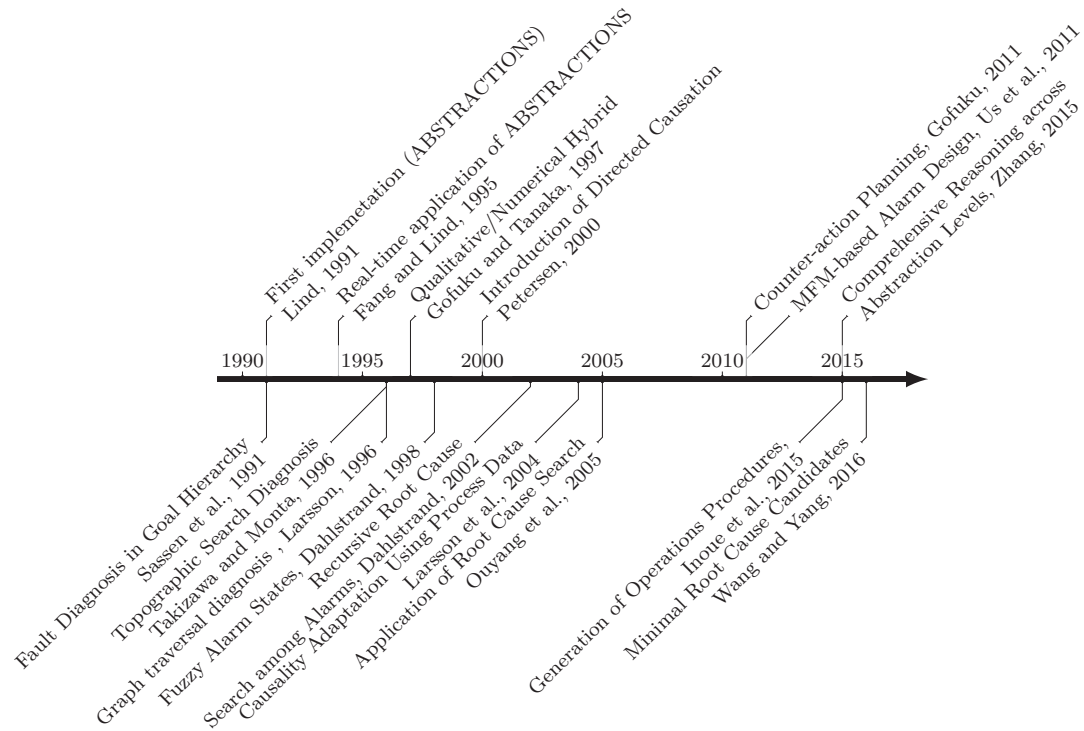


Figure 3.4: Chronology of MFM implementations for operator support and situation awareness.

Lind (1991) presented the fundamentals for using MFM in automatic operator support with an object oriented framework and generic cause and consequence reasoning called ABSTRACTIONS. Fang and Lind (1995) applied ABSTRACTIONS for a real-time application with an interface to a pilot process for causal diagnosis based on an MFM model. An alternative approach based on MFM was proposed by Sassen et al. (1991), who implemented a hierarchical search through goals and sub-goals in the means-end hierarchy of MFM. The same concept was adapted by Takizawa and Monta (1996) in tracing faults to a specific flow structure in the hierarchy and identifying the fault from inconsistencies between the fault propagation and sensor observations. With the focus on decision support Gofuku and Tanaka (1997) proposed to extend the functional model with operational knowledge, including alternative behaviours of sub-systems. By incorporating this operational knowledge into the abstract representation of MFM they established a quantitative simulation based on Hybrid Phenomena Theory to facilitate quantitative prognosis.

Larsson (1996) demonstrated the application of an MFM based expert system implementation for alarm analysis and sensor validation. The system employed cause analysis to distinguish primary alarms, close to the root cause, and consequential alarms associated with the fault propagating from the root cause. This system was

designed as an interactive console querying responses from the operator to evaluate states of MFM functions. In extension of this work Dahlstrand (1998) considered a fuzzy assignment of alarm states and propagated states to account for inconsistent alarm configurations, making the system more robust against chattering alarms. Based on the causal analysis of alarms Dahlstrand (2002) proposed to identify the minimal set of root causes consistent with all observations in the system, which could be determined by removing inconsistent edges in a dependency graph of all possible states derived from the corresponding MFM model, similar to the on-line propagation established by Lind (1991). Even though the MFM modelling of functions and goals had been well established, Petersen (2000) identified the need for a distinction of different causal relations, defining direct and indirect influence as well as defining a comprehensive set of propagation rules based on the patterns in MFM syntax. Larsson et al. (2004) expanded on the refinement of causal relations by introducing a dynamic adjustment of causality based on a pairwise correlation between the variables corresponding to adjacent functions. As only the causal relation is adapted the fundamental model is maintained and adapted to the current operating mode (Larsson et al., 2004). Ouyang et al. (2005) used MFM to successfully diagnose different design accident scenarios of a nuclear reactor, validating the process representation and propagation method. Most recently Wang and Yang (2016) proposed an expert system implementation derived from an MFM model to determine minimal root cause sets, similar to Dahlstrand (2002) with the addition of information about common operator mistakes to verbalise the diagnosis. Zhang (2015) presented the most recent set of propagation patterns for cause and consequence analysis to support operators situation awareness in nuclear power plants.

Leveraging additional operational knowledge to extend MFM models (Gofuku and Tanaka, 1997), Gofuku (2011) proposed a purely qualitative approach to generate natural language explanations for the diagnosis. Inoue et al. (2015) carried the concept of MFM and operational knowledge further to automatically generate counter-action procedures.

3.3 Summary

Both alarm management and decision support are active research fields with a wide variety of methods investigated to support the respective tasks. Among the discussed methods, MFM stands out as having found the most versatile application across the aspects of operator support to various degrees of maturity. With methods for alarm design, alarm analysis, situation assessment and counter-action planning described in literature, MFM is well suited for a comprehensive support system bridging all aspects of operators' tasks. While many contributions outline the principles toward this kind of comprehensive system, independent of the methodology the challenge described by Wang et al. (2016a) remains to create a framework that integrates different facets of alarm management, data analysis and decision support.

CHAPTER 4

Reasoning System

The advantage of using qualitative rather than quantitative models for diagnosis is the application of discrete states which significantly reduce the simulation space to run a variety of scenarios. In a qualitative causal model the effects and causes of a deviation can be identified by propagating it within the model. For operator support, causes of observations available from the system can be analysed to determine common root causes and present only this as the most relevant. Additionally, the model can be applied to infer consequences of an identified cause or an intended action on the system to evaluate the severity of a failure or the viability of a mitigation strategy, respectively. A reasoning system is realised to provide the inference as well as ensuring that the suggested scenarios are updated when observations change. An updated reasoning approach for Multilevel Flow Model based operator support in a real-time setting is described in this chapter. The outlined reasoning method and case study are discussed in detail in Paper A.

Previous implementations of the MFM reasoning considered one evidence as the trigger for analysis and used other evidences to prune the resulting cause or consequence tree based on other evidence Zhang (2015) or compile a fault tree analysis for a given evidence (Wang et al., 2016b). A new approach to propagation in causal models is proposed that leverages the connection of the majority of occurring events during alarm floods, with the intention of being applicable to on-line diagnosis. For on-line application the method improves execution speed while maintaining the inference up to date with the observations from the system. The reasoning system is realized for dynamic inference to accommodate the model changing according to the configuration of the system or to simulate the effects of deliberate changes for counter-action planning.

4.1 Causal Inference

Multilevel flow models utilize the functional syntax to describe the system while providing a causal representation of the system. Without the context provided by MFM, causal relations are explicitly shown in a SDG. Each edge in the causal SDG points from the cause to effect and is denoted with a positive or negative sign. A positive sign indicates that an increase in the cause leads to a increase in the effect while a negative edge indicates the inverse behaviour, i.e. an increase leads to a decrease in the effect. The mapping between function-relation pairings in MFM to Signed Directed Graphs is illustrated in Figure 4.1. The comprehensive definition of cause-effect relations in MFM is found in (Zhang, 2015).

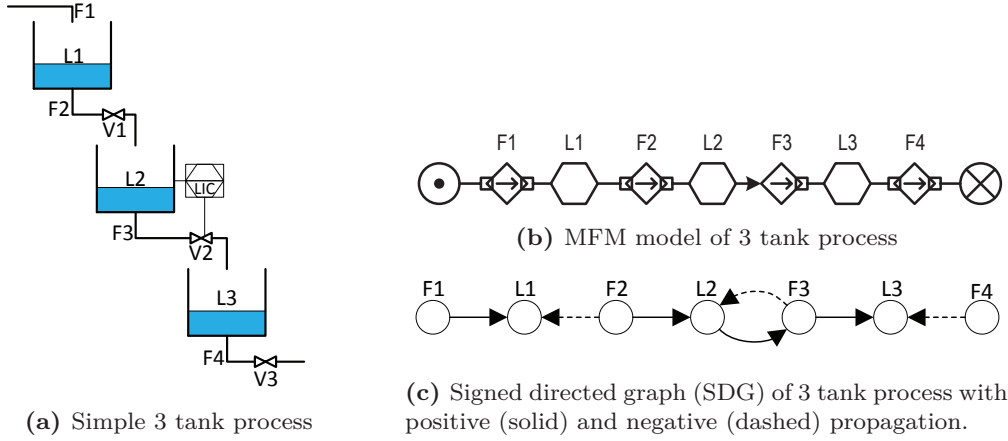


Figure 4.1: MFM and SDG representation of simple 3 tank process

To facilitate dynamic adaptation of the model, the reasoning system is established as by dynamic propagation in the causal model. Qualitative states resembling the alarm levels *low-low* / *low* and *high* / *high-high* are propagated along the causal connections in the model. Both causes and consequences can be identified that way by traversing opposite or with the direction of the causal relations respectively. The propagation is constrained to a meaningful scope by the following limits:

1. *Number of node occurrences in a path*: The same node may occur with the same state, considered special case forming a "loop". Inference of an opposing state in the same path is deemed invalid.
2. *Number of edge traversals in a path*: Given above constraint, an edge can only occur once in each traversed path.
3. *Maximum path length* is not restricted.

4.2 Inference Maintenance

Information from the plant is considered as evidence in the reasoning system, while information generated by inference is referred to as assumptions. Both, an evidence and an assumption, refer to a node, i.e. MFM function, and the qualitative state. For clarity, inferred assumptions can be distinguished between cause direction and consequence direction, concerning previous events and future events, respectively. Besides the inference direction assumptions for causes and consequences are treated identically. Each new assumption is validated to comply with the scope of the analysis by above constraints by the process outlined in Figure 4.2. If an assumption creates a contradictory path it will be retraced. Since all inferred assumptions are causally

linked to preceding assumptions, all assumptions that exclusively lead to a contradiction cannot be valid and the inference path will be traced back to an assumption that supports other paths as well.

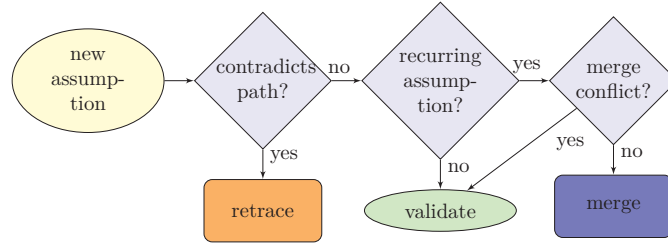


Figure 4.2: Validation process for assumptions

Any valid assumption is further evaluated with respect to already generated assumptions to utilise the connections between during propagation already. As the evidence from the plant may evolve over time a recursive devalidation scheme is put in place to keep the inference results synchronised with the actual plant situation. The remainder of this section details the procedures and constraints for merging and devalidation.

4.2.1 Merging

Propagating causally connected evidence e_1 and e_2 will lead to the inference of identical assumptions a_1 and a_2 . Given that a_1 and a_2 were inferred by traversing the same relation, further assumptions based on either a_1 or a_2 will be identical as well. To ensure that the inferences are complete and consistent the merging cannot create contradictions, nor can it omit results that would be valid.

The merging procedure is based on the pre-existing assumption a_1 being fully propagated when the merge with candidate a_2 is evaluated. The causal tree originating from a_1 is described by the tree $\mathcal{T}_{a_1} = \{A_1, J_1\}$ with the sets of assumptions A_1 and justifications J_1 , where each justification $j \in J$ describes the preceding assumption and traversed relation for the corresponding element in A_1 . A_1 contains the disjoint sets of valid assumptions A_v and contradicting assumptions A_c . To preserve consistency the inference path to a_2 , $\mathcal{P}_{a_2} = \{A_2, J_2\}$ cannot contradict or create a loop with any part of \mathcal{T}_{a_1} . Effectively this is guaranteed by prohibiting that any of the functions traversed in \mathcal{P}_{a_2} occur in A_v . In the same fashion, the completeness is verified by ensuring that the functions F_c covered by contradictory assumptions in A_c actually occur in \mathcal{P}_{a_2} or A_v . If the merging candidate is found to be consistent and complete by this validation the inference paths are linked by effectively replacing a_2 with a_1 in the inference path \mathcal{P}_{a_2} and thereby adding \mathcal{T}_{a_1} to the inferences from e_2 .

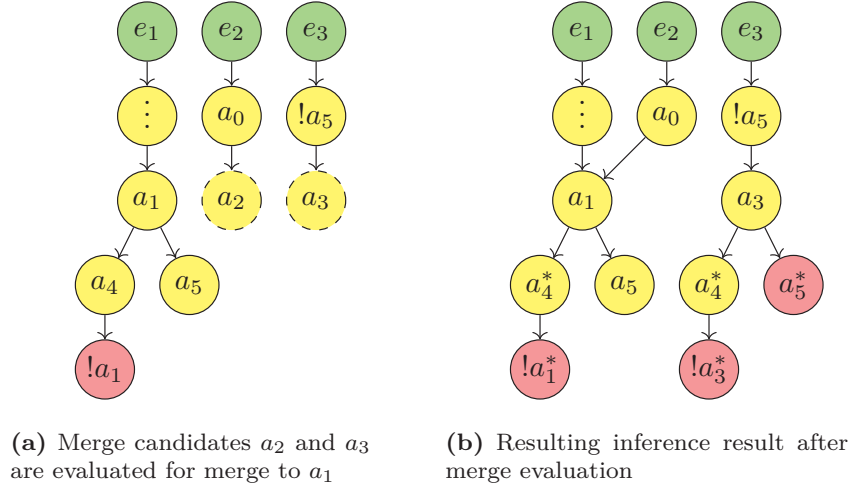


Figure 4.3: Exemplary merging situation. Evidence (green), assumptions (yellow) and contradictions (red) are illustrated. Contradictions are labelled as negation (!) of the assumption they contradict, and subsequently retraced. Retraced assumptions (*) are not part of the valid result.

4.2.2 Devalidation

In the effort of maintaining only valid and relevant results the devalidation mechanism outlined in Figure 4.4 is implemented. The previous assumptions or evidence supporting an inference are stored with the inferred assumption as justifications. Whenever the justifications change, the validity of supporting assumptions or evidence is evaluated and no longer justified assumptions are accordingly devalidated. Through the inference path the devalidation will propagate further until reaching an end point. A devalidated assumption that does not justify any other inference is then removed which ensures the reliable removal of irrelevant or invalid assumptions.

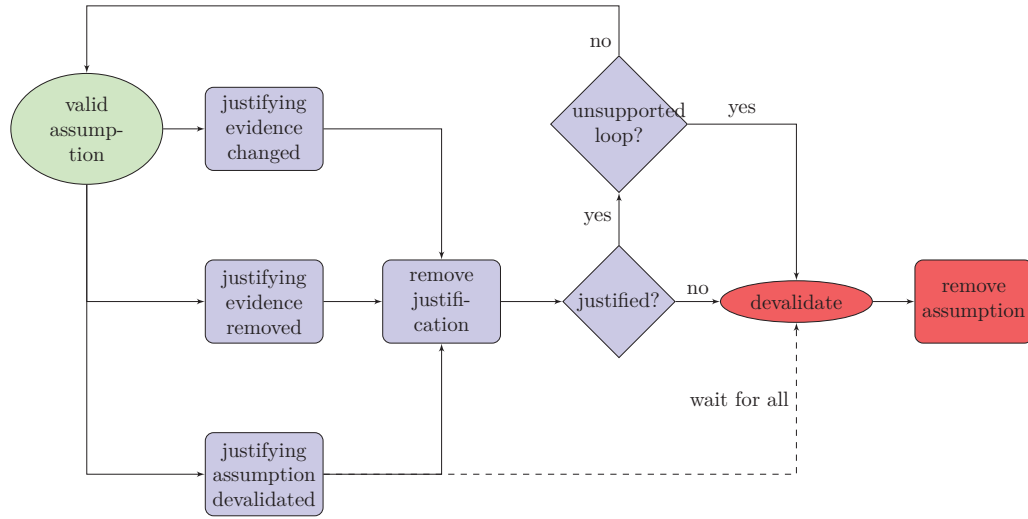


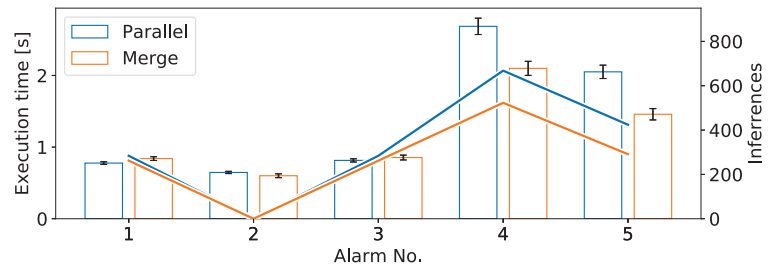
Figure 4.4: Devalidation logic for result clean-up

4.3 Case Study

To validate the advantage of the proposed method an emerging situation in the Tennessee Eastman benchmark (Downs and Vogel, 1993) is analysed. The data is based on the control strategy suggested by Ricker (1996) with the simulation data available at Ricker (2019). The situation is described by the alarm log in Figure 4.5(a). To evaluate the reasoning performance identical implementations of the propagation with and without the merging mechanism are compared with the result shown in Figure 4.5(b). While the propagation took slightly longer with 8% increase in execution time for an individual fault due to the additional checks introduced by the merging algorithm, execution time was reduced by up to 29% for subsequent evidence. The number of assumption repetitions is consistently reduced by the merging algorithm.

No.	Event	
1	F9	low
2	F9	RTN
3	F9	high
4	F5	low
5	T _{cc}	high

(a) Alarm log of condenser coolant fault



(b) Execution time (bars) and assumptions per update step (line)

Figure 4.5: Scenario and performance results of the case study

4.4 Summary

The presented method efficiently utilises the connection of events in emerging abnormal situations like alarm floods. By combining propagation paths during the result generation the execution time and size of the resulting graph are consistently reduced when the analysed events are connected. As recurring and connected paths are directly related in the generated results this method also decreases the effort of identifying common causal paths between occurring events as these emerge naturally from the structure of the results.

CHAPTER 5

Analysing Models and Results

Using qualitative models for causal analysis and prognostic inference provides a comprehensive set of possible causes and consequences. MFM models are applied in support tools for design (Rossing et al., 2010; Us et al., 2011) and diagnosis of industrial plants (Larsson et al., 2007; Zhang, 2015). By comparing the model to established methodologies for propagation analysis like fault and event tree analysis the model can be validated (Nielsen et al., 2018a).

This section describes graph interpretations of both MFM models and results. First, the extraction of a SDG with equivalent causality from an MFM model is described, according to paper D. Then the visualisations of the inference results for comparison with fault and event trees, as well as a combined visualisation first presented in paper E are discussed.

5.1 Multilevel Flow Models as causal graphs

Causal process representations as SDG have been demonstrated as knowledge representation for qualitative diagnosis (Peng et al., 2014; Wan et al., 2013). However, generation of SDGs is not straightforward and the diagnosis is usually limited to process variables with relations that can be described mathematically or identified from data (Yang et al., 2012). MFM on the other hand facilitates structured modelling of the process based on the process understanding.

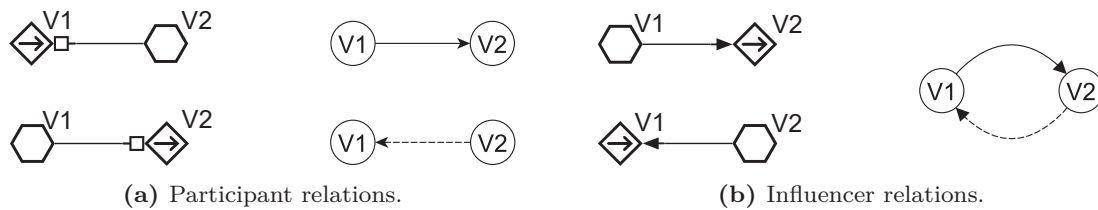


Figure 5.1: Representation of Signed Directed Graph (SDG) equivalents of basic MFM relations. The arcs in the SDG represent positive (solid) and negative (dashed) propagation.

The vertices of a signed directed graph typically correspond to process variables such as actuated and measured variables. The arcs are directed from 'cause' to 'effect' with the sign denoting how deviations propagate. A positive state on a vertex results in a positive state along a positive arc and a negative state along a negative arc. In contrast the propagation in MFM is defined by the patterns of functions and relations

connecting them, defined in a rule set (Zhang, 2015). Since the majority of model patterns in MFM propagate in a linear way comparable to the signs of an SDG, it is possible to generate a SDG from a given MFM model. Figure 5.1 illustrates a mapping of the most common MFM patterns into the respective SDG.

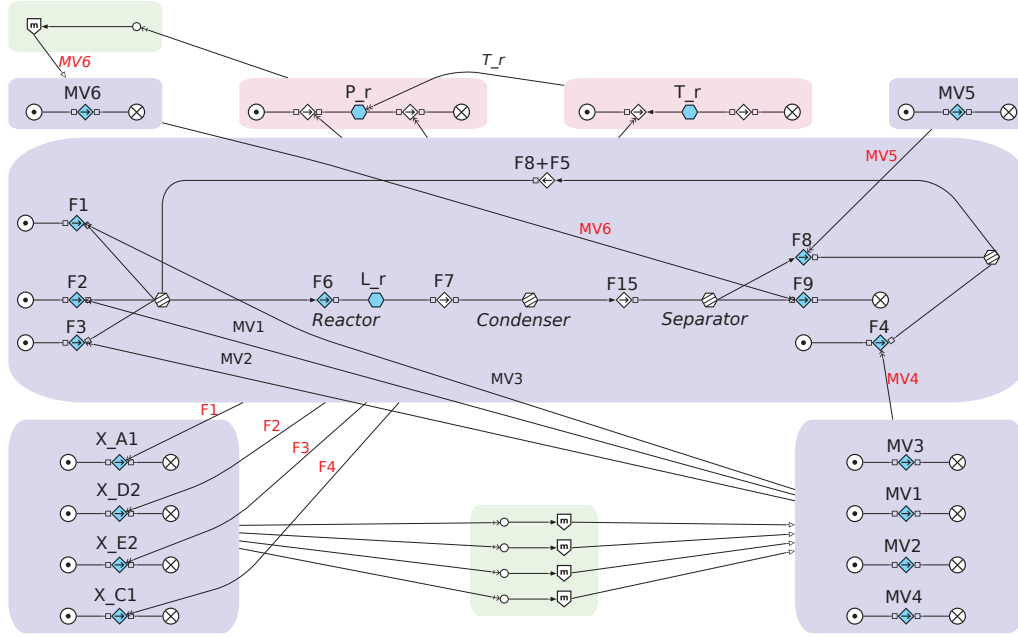


Figure 5.2: MFM model derived from plant topology and process understanding

To diagnose abnormal situations in the plant, the MFM model is linked to the process by adding an attribute to those functions that correspond to process variables. The explicit syntax of MFM contains a number of functions that do not correspond to process variables, in the corresponding SDG only vertices in set I reflect process variables. The MFM equivalent graph (V, E) can be reduced to a SDG of process variables by the two steps shown in Figure 5.3(a), similar to the reduction scheme presented by Kramer and Palowitch (1987). First, leaf nodes identified by in-degree $\bar{\partial}^+ = 0$ or out-degree $\bar{\partial}^- = 0$ are recursively removed, as they do not contain any information regarding the causality between process variables. Second, intermediate nodes are iteratively contracted by removing the node and replacing its connected edges by all combinations of entering and leaving edges. The sign of each new edge is determined as the product of the original edge sign. The resulting graph describes the same causal relation between the monitored variables as the MFM model, but omits any additional elements introduced by MFM.

Generating a signed directed graph from an MFM model enables the use of graph theory for structural analysis of the causal model, as well as the direct application of SDG approaches to diagnosis and a comparison of models and results between

the methodologies. The case study in paper D illustrates the comparability using an MFM model representing the reactor of the Tennessee Eastman Process, which is shown in Section 5.1. The red edges in Figure 5.3(b) correspond to reaction dynamics that are not modelled in MFM but had been considered by Ma and Li (2017). Apart from the reaction aspect not yet regarded, the MFM model provides an accurate equivalent to the differently derived SDG.

Step 1:

```

while  $v_i \in V \setminus I$ ;  $\delta^-(v_i) = 0 \vee \delta^+(v_i) = 0$ 
do
   $V \leftarrow V \setminus v_i$ 
end while

```

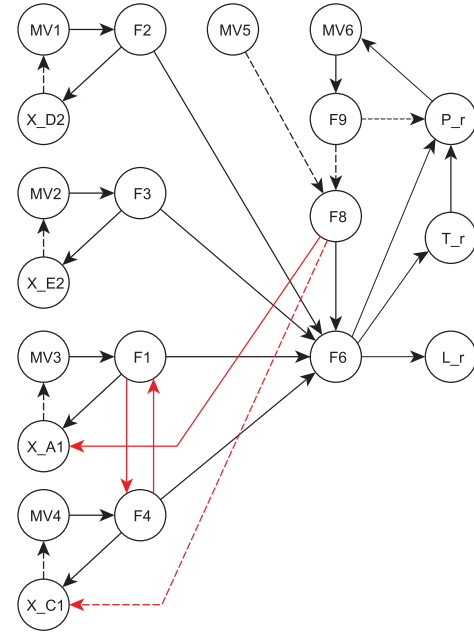
Step 2:

```

while  $v_i \in V \setminus I$  do
  for  $e^+ \in E$ ;  $e^+ = (v_m v_i)$  do
    for  $e^- \in E$ ;  $e^- = (v_i v_n)$  do
      if  $v_n \neq v_m$  then
         $e^* = (v_m v_n)$ 
         $E \leftarrow E \cup e^*$ 
      end if
    end for
  end for
   $V \leftarrow V \setminus v_i$ 
end while

```

(a) Reduction algorithm used to obtain a graph featuring only monitored nodes. V , I , and E are defined as the sets of nodes, monitored nodes and edges, respectively



(b) Signed Directed Graph generated from MFM model

Figure 5.3: Reduction algorithm for MFM derived SDG and reduced SDG for the reactor of the Tennessee Eastman Process

5.2 Visualising inference results

To provide the MFM based reasoning as a tool for a variety of applications, a software implementation with a range of output formats for the inference results has been developed during this project. In previous work with MFM based diagnosis different approaches to the analysis of inference results have been used, where the majority of analyses interpret the results as a fault or event tree. A tree structure of possible causes leading to a detected event or of possible consequences of said event is generated from the inference (Wang et al., 2016b; Zhang, 2015). An alternative is to represent the MFM model as a dependency graph with the causal connections between all

possible states for each function. Based on the inference, the contradictory states and causal relations in the dependency graph are removed (Dahlstrand, 2002), enabling a comprehensive analysis of the combined plant state. The newly implemented user interface provides three basic views: a connected graph view and a tree view for either causes or consequences, and a combined graph of all inferences.

The improved inference system described in Chapter 4 reduces the majority of redundant inferences. The structure of these inference results can be directly viewed as a connected graph, where overlapping propagation paths appear as one since they are merged. A direct causal path between two observations can be readily identified from one propagation direction, as either the cause or consequence paths will overlap.

While the merging increases the reasoning efficiency, it requires the traversal and analysis of the results to derive the equivalent of a conventional fault or event tree that explicitly considers the propagation paths by which a specific event affects the system (Crawl and Louvar, 2011). To derive a complete tree structure from the merged inference results, each merged node is replaced by a copy of the original node. Copying the node recursively copies the entire inference tree derived from that node, yielding multiple versions of the identical inference steps. The developed user interface provides two ways of viewing the inference results as cause or consequence trees: A collapsible tree view to facilitate the investigation of the inference path, similar to a directory tree on a computer, and a graph output of the expanded tree for visualisation and comparison to fault trees.

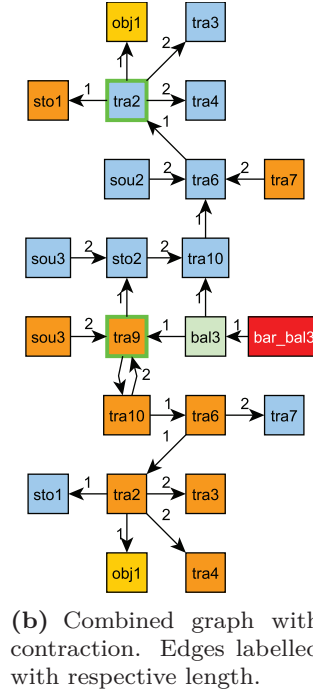
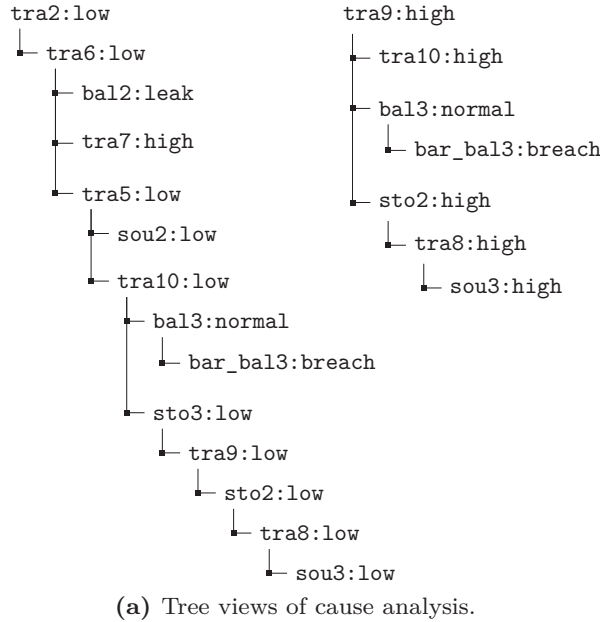


Figure 5.4: Tree views of causes and combined inference for the watermill example.

To further support the analysis of causes and consequences together a combination of both inference directions is considered, similar to a dependency graph (Dahlstrand, 2002). As described in paper E this combination enables the interpretation of causal direction from the paths in the resulting directed graph, while only representing the relations and states relevant to the current situation. In addition to the combination of cause and consequence inference outlined in paper E the developed software contracts propagation paths that do not contain any evidence or branching points to make the graph more readable. To produce the combined graph, the inferred results are projected into a graph with distinct vertices for pairs of MFM function and inferred state, e.g. "sto2:high" and "sto2:low" are distinct vertices, whereas two assumptions corresponding to "sto2:high" are reflected by one vertex only. All edges are initialized to length 1. The contraction is applied iteratively to any vertex in the combined graph that has in-degree $\tilde{\delta}^+ = 1$ and out-degree $\tilde{\delta}^- = 1$, i.e. only one propagation path through these assumptions is possible. By contracting a vertex, the previous and following vertex are directly connected by a new edge maintaining the direction and with the combined length of the entering and leaving edge of the contracted vertex. The combined graph facilitates the analysis of relations between multiple alarms and inferred root causes and final consequences, while maintaining a readable size by contracting intermediate propagation paths.

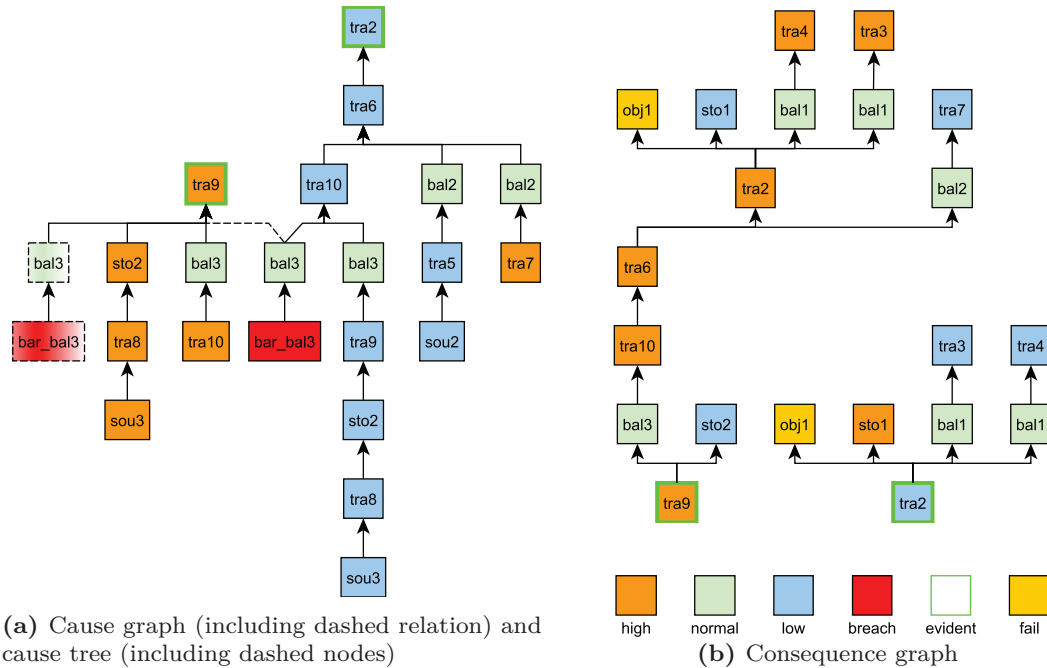


Figure 5.5: Graph representations of cause and consequence inference for watermill scenario.

Figures 5.4 and 5.5 provide examples of the different result formats for a fault

scenario of the watermill described by Lind (2011), the model and inference of this scenario are described in detail in Paper E. Two faults are observed: "tra9:high" corresponding to a high flow of water onto the water wheel and "tra2:low" representing a low intake of wheat for grinding. Figure 5.5(a) shows the close correspondence of the merged result graph and an expanded fault tree: replacing the dashed relation to the merged inference "bal3:normal" by the copy shown with dashed frame yields an expanded tree. The common root cause is found to be loss of water at the water wheel, e.g. by the water spilling over rather than moving down with the buckets of the wheel. There is no overlap in the inferred consequences for the two observations, so the consequence graph is in fact an event tree.

5.3 Summary

To facilitate the further analysis of MFM models and the causal inference results, both can be represented as graphs. The conversion of an MFM model to a SDG reflecting the same causality has been shown to provide the means to efficiently define a consistent model without the need for qualitative information. The graph representation of the model enables the comparison with related publications using SDG as a knowledge representation and further supports application of graph theory, such as structural analysis of connectivity.

Representing the inference results as a fault or event tree supports the analysis usually performed for risk assessment (Crawl and Louvar, 2011; Hu et al., 2015). The combined graph provides an overview of the causal paths and potentially emerging situations, allowing the analysis of meaningful connections between occurring alarms.

CHAPTER 6

Cause Ranking

Too much information overloads operators, reduces their capacities for thorough situation assessment, and leads to inefficient reactions and mistakes (Hollnagel, 2002). Root cause analysis can support operators to make sense of a large number of alarms and events by identifying the origin of the abnormal situation. A qualitative reasoning system can generate an exhaustive set of possible scenarios to explain given observations from the physical system. While it is desirable to consider all possible scenarios to ensure a thorough situation analysis, filtering is required to support operators in their decision process rather than adding on to the large amount of information provided. To focus the operators attention on the most likely scenarios, this chapter outlines a number of approaches to ranking the root causes proposed from the qualitative reasoning system.

Previous work using causal models for root cause analysis suggested to rank root causes based on e.g. the timing of analysed events (Larsson et al., 2006), Bayesian Network analysis (Hu et al., 2015), and the comparison of inferred consequences with sensor trends (Maurya et al., 2007) or relative sensor deviations (Arroyo Esquivel, 2017). Based on the qualitative reasoning implemented for MFM some ranking approaches are investigated. The proposed improvements to the qualitative reasoning in MFM produce a connected result graph that combines the inferences of all observed events. First, the direct analysis of the result graph outlined in Paper A is described. Afterwards concepts for transforming either the causal model or the reasoning results into a BBN for probabilistic analysis are summarized. The generation of BBN was initially presented in Paper F. All presented approaches to root cause ranking are demonstrated on a fault scenario of the Tennessee Eastman Process.

6.1 Distance based Ranking

While the fault propagation in causal models is performed in a deterministic manner to reveal all potential causes and consequences, the propagation path can highlight more likely causes. The underlying assumption is that each propagation step introduces uncertainty in terms of the magnitude and likelihood of the influence. A cause candidate is thus less relevant the longer the propagation path from any observation is. With multiple possible causes for a fault, only one has to be present for the fault to occur. However, a possible root cause that explains multiple observations is more relevant as it is in line with the base assumption of few root causes being responsible for a large number of observed upsets in the plant.

The distance based approach attributes a weight $w_e \in [0; 1)$ to the edges in the result graph. An initial value is attributed to all observed state and degrades along

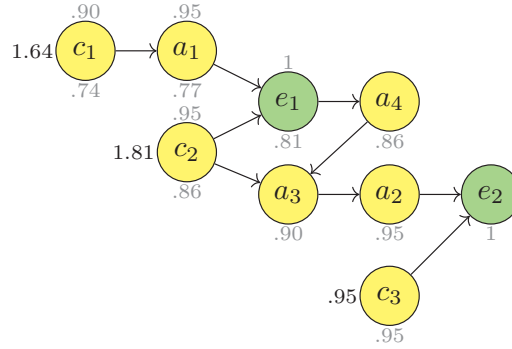


Figure 6.1: Visualisation of distance based ranking of identified causes c_1, c_2, c_3 for observations e_1, e_2 with uniform edge weight $w_e = 0.95$. Assumption labelled with weights related to e_2 (below) and weights related to e_1 (above). Root cause weight (left) determined as sum of evidence related weights.

the propagation path toward a root cause by being multiplied on each edge traversed. Where the paths of multiple observations join the resulting node weight is determined as the sum of the weights from incident paths. Normalizing by the number of present observations, the weight of a root cause can serve as a global measure for importance. Within a given scenario the sum of weights propagated by the algorithm shown in Algorithm 1 allows the distinction of the most relevant causes based on the length of propagation paths between observations and root cause candidates, as shown in Figure 6.1

In addition to the structure of causal connections, MFM also provides a syntax reflecting the design and operations intentions. The syntax can be leverage in the weighting algorithm by varying w_e depending on the MFM relation associated with the propagation. Relations within one flow structure generally reflect physical interactions between process functions of one perspective. In contrast, the relations along the means-end dimension and control perspectives typically reflect intended and designed behaviour that provide the necessary functions to achieve the plant objectives. By attributing a lower value to the propagations along flow relations the meaning of the MFM syntax can be incorporated into the root cause analysis.

Algorithm 1 Weighing causes

```

 $R \leftarrow \emptyset$ 
 $w_a \leftarrow 0 \forall a \in A(\mathcal{G})$ 
for  $(a \in A(\mathcal{G}), \bar{\delta}^-(a) = 0)$  do
   $w_a \leftarrow 1$ 
   $M \leftarrow \emptyset$ 
   $N \leftarrow \{a\}$ 
  while  $N \neq \emptyset$  do
     $B \leftarrow \{b \in A(\mathcal{G}), bn_1 \in E(\mathcal{G})\} \setminus M$ 
     $N \leftarrow N - n_1$ 
    for  $(b \in B)$  do
       $w_b \leftarrow w_b + w_e \cdot w_{n_1}$ 
       $M \leftarrow M + b$ 
       $N \leftarrow N + b$ 
    end for
    if  $\bar{\delta}^+(n_1) = 0$  then
       $R \leftarrow R + n_1$ 
    end if
  end while
end for

```

6.2 Bayesian Networks for Cause Ranking

Bayesian Networks or BBNs are graph models of the dependency between states or observations and the probability related to these dependencies. The Bayesian Network consists of the graph structure, reflected by the causal structure in a fault diagnosis context and the conditional probabilities for all relations in the structure. The conditional probabilities are provided as conditional probability tables (CPT) mapping all considered states of the parent nodes onto the states of the affected node. Based on the a priori probabilities of root events assigned during design and the conditional probabilities, the joint probability distribution for the entire system can be determined. Observations from the system are taken in to give certainty on specific states and the joint probability will be recalculated given the certain observations. The order of probability of root events is then determined by the descending order of probability or a combined measure of a cost function and likelihood. BBN have been proposed as the basis for fault diagnosis based on either fault tree analysis (Milford, 2006) or HAZOP studies (Hu et al., 2015). Other analysis on BBN can give information about the most relevant observation to obtain in order to distinguish closely ranking scenarios by the value of information metric. (Pearl, 1988)

MFM models provide a causal structure for the system, which can be condensed to a representation of causality between process variables described in Section 5.1. The structure of a BBN could be directly derived from such a causal model. However, BBN are constrained to non-cyclic graphs while industrial plants typically contain control

loops and process re-circulations with corresponding cycles in a causal representation. This section describes two fundamental approaches to transforming causal models derived from MFM into diagnostic BBN: A heuristic approach to removing cycles from causal models and the conversion of cause reasoning results into a BBN.

6.2.1 Direct Mapping of Causal Models

In the framework of alarm analysis each variable in the causal graph can be in one of three states: *high* or *low* alarm, or *normal*. A BBN can be created from trinary nodes having these three states and corresponding CPT for the relations in the causal model. Table 6.1 shows the CPTs for node v_j causing v_i with sign s_{ji} on the causal edge. The resolution of cycles in the causal structure is based on these assumptions:

- Reciprocal influence between adjacent variables has a dominant direction, defined by the process control and design during normal operations.
- Faulty sensors, defective controller implementations or broken actuators can cause a fault in a control loop which will can be reflected as fault in the set point of the actuated variable as symptom for a fault in the control loop.

Accordingly the recipe for resolving cycles in the causal model during BBN generation consists of the following steps:

1. Maintain only the dominant influence in reciprocal relations.
2. Break control loops by removing the influence from controlled to actuated variable.
3. Generate BBN from remaining graph with CPT based on the edge sign and the corresponding mapping from Table 6.1.

This approach, outlined in Paper F, considers the fault propagation as a logical rather than a probabilistic process due to assigned conditional probabilities. The root cause ranking is thus a balancing between the described propagation paths favouring the least contradictory path for the given observations.

Table 6.1: CPT for node v_i based on parent v_j

$v_i \backslash v_j$	$s_{ji} = -I$			$v_i \backslash v_j$	$s_{ji} = I$		
	high	normal	low		high	normal	low
high	0	0	1	high	1	0	0
normal	0	1	0	normal	0	1	0
low	1	0	0	low	0	0	1

6.2.2 Mapping of Reasoning Results

As the assumptions for the direct mapping remove some of the causal relations and with that also remove propagation paths, the connection between occurring events could be hidden from the diagnostic BBN through the suggested direct mapping. Instead it seems more meaningful to analyse cycles based on a given observation. To analyse if the cause tree for a given observation results in causal loops, the graph needs to be traversed, which is equivalent to the fault propagation performed by the MFM reasoning system. Consequently, the graph structure resulting from the merging reasoning approach provides a meaningful basis for generating a BBN by removing encountered cycles. During the propagation encountered cycles are marked as causal loops which can be removed during BBN generation. The resulting BBN is composed of binary nodes representing either a *high* or *low* faults of variables with the states *fault occurred* and *normal*.

The possible causing faults for each node are connected with an OR gate, reflecting that a *fault occurred* in either of the parents can lead to the node assuming *fault occurred*. To incorporate a notion of probability the degrading weight of edges similar to Section 6.1 can be applied to the propagation paths between considered variables. This means, that the edge weights w_e considered for the respective MFM relations are multiplied when contracting edges to produce the signed directed graph. The resulting weight is then assigned as probability of the fault propagating by using the nosiy-OR gate (Antonucci, 2011), which allows for the affected node to remain *normal* even though a parent assumed *fault occurred*.

The propagation results are based on a specific fault for a variable and all nodes in the resulting BBN refer to a variable and state. In contrast to the direct mapping with trinary nodes covering the mutually exclusive states of a variable, a binary BBN generated from a fault tree does not express the relation between mutually exclusive states of the same variable. For instance, both a *low* and *high* fault on a specific variable could occur in the BBN and their likelihood would be computed independently since no relation describes that their joint probability is constrained since they describe the same variable. Lampis (2010, p. 96 f) describes the use of a n-ary parent node that links the probabilities of the possible states of a variable. In the BBN generated from propagation results this can be applied to the identified root causes by adding a trinary parent node with the states *low*, *normal*, and *high*. Table 6.2 shows corresponding CPT for the node corresponding to *high* or *low* fault, respectively.

Table 6.2: CPT for binary fault node with trinary parent node

trinary low	high	normal	low
	high	normal	low
fault occurred	0	1	1
normal	1	1	0

trinary high	high	normal	low
	high	normal	low
fault occurred	1	1	0
normal	0	1	1

6.3 Case study

The Tennessee Eastman Process (TEP) (Downs and Vogel, 1993) serves as a case study for the proposed ranking approaches. The MFM model of the thermal behaviour, excluding reaction dynamics, is shown in Figure 6.2 together with the derived SDG model. The diagnosis is applied to disturbance scenario IDV12 with a high coolant temperature in the condenser. During the emerging situation the most notable alarm activations are *F9 high* and *F5 low*, determined by a 2% band around the steady state value using the simulation data available from Ricker (2019).

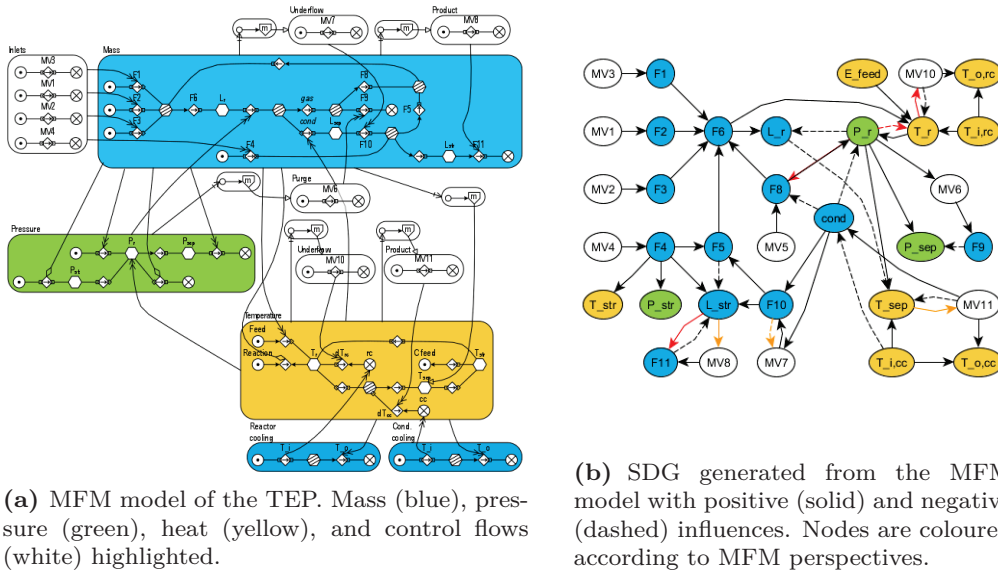


Figure 6.2: Causal models of the Tennessee Eastman Process

The distance based ranking is performed in real-time on the emerging situation with the alarm occurrences shown in Figure 6.3(a) with the top ranked root cause and the actual root cause evolving as shown in Figure 6.3(b). The BBN based root cause ranking is only performed on the two final alarms with the results shown in Figure 6.4. The direct mapping (A) is achieved by removing minor reciprocal influences (red) and control relations (yellow) in Figure 6.2(b). Methods B1 and B2 resemble the generation of results without and with merging respectively. And the trinary parent nodes are introduced in method C.

No.	Event	
1	F9	low
2	F9	RTN
3	F9	high
4	F5	low
5	T _{cc}	high

(a) Alarm log of condenser coolant fault

No.	Pos. Weight		Top Cause	Top Weight
1	15	0.377	Low reactor pressure	0.735
3	8	0.599	High reactor pressure	0.735
4	1	1.262	High condenser coolant temp.	
5	1	2.119		

(b) Root cause rank of actual cause and top ranked root cause after each update step.

Figure 6.3: Scenario and results of distance based ranking

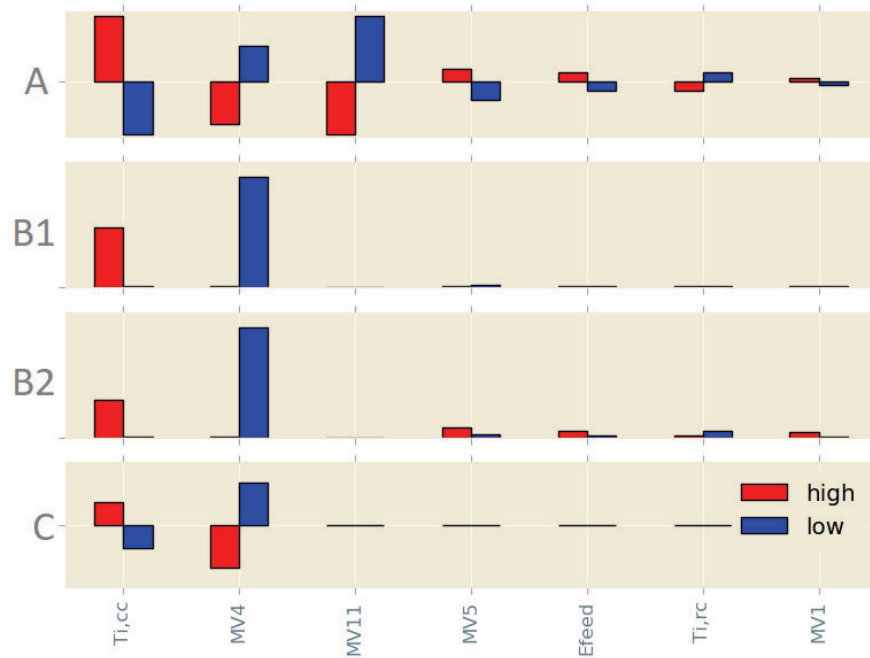


Figure 6.4: Ranking results of the BBN approaches

6.4 Summary

This chapter presents different approaches to ranking root causes based on the causal models. On the one hand, the concept of incorporating ranking into the propagation by degrading the value of a root cause with increasing distance from observations has been integrated with the reasoning system. In the case study this ranking approach showed good results and is flexible to incorporate more of the knowledge implicit to an MFM model. On the other hand, the conversion to BBNs facilitates the inclusion of experience values for fault probabilities and allows to leverage the established methods and frameworks for BBN. While the distance based method is tightly integrated with the MFM framework and reasoning system, BBNs provide a wide range of methods for varied applications. A specific application may require the use of one or the other method.

CHAPTER 7

Operating Modes and Procedures

Industrial plants can be operated in a wide range of configurations. These configurations can reflect different production goals, maintenance, start-up, shutdown, or emergency operations. Tracking these configurations and distinguishing them from fault situations is important for a diagnostic system. In the proposed support system, the dynamic reasoning is in place to facilitate the efficient on-line propagation. Since the reasoning is evaluated on an MFM model at runtime, the model can easily be adapted to match the current state of the plant.

Plant configurations related to operating modes have been investigated in the context of MFM and linked to the hierarchical structure of the modelling framework with a close connection to operating procedures (Lind et al., 2012). The relation of operating procedures and MFM models is investigated further in Paper G. The work proposes the implicit representation of the constraints for each mode as well as a consistent modelling method for successive modes in an operating procedure at the plant wide level. In plants with redundant sub-systems, the modes of these sub-systems can change individually without affecting the plant wide operation. Consequently, a concept for modelling the overall plant with the contributions of multiple redundant sub-systems is needed.

As operating procedures are part of the plant design and documentation, this knowledge is available to incorporate with the support system. In Paper B a representation of operating procedures as automata is suggested. Experience from industrial plants shows, however, that design and implementation can differ a lot. In practice, executed operating procedures deviate from the documentation, e.g. due to inconvenient operations, better ergonomics, or operators experience. To correctly distinguish actual errors in the procedure execution from these adaptations of the procedure, a data analysis based validation of operating procedures is proposed.

This chapter first describes the relation of operating modes and the functional hierarchy in MFM. Then, the approach to functional modelling of modes and procedures presented in Paper G is summarized. The unpublished discussion on modelling redundant sub-systems and components is presented. After that, the representation and validation of operating procedures according to Paper B is outlined. Finally, the suggested modelling and model adaptation approach is applied to an industrial case study. The procedure validation and execution tracking is demonstrated on the industrial case study.

7.1 Functional aspects of Operating Modes

Configuration changes of a plant can be associated to different levels of abstraction in the context of functional modelling. Lind et al. (2012) first established the hierarchical interpretation of operating modes according to the modelling hierarchy of MFM and their importance to situation assessment. The interpretations are based on the mapping between means serving the fulfilment of defined ends. Table 7.1 summarizes the mode distinctions as outlined by Zhang (2015).

On the one hand a mode can be defined by an unchanged purpose, i.e. objective or function, where only the supporting elements are adapted, this will frequently be the case for switches between redundant systems, that could imply different functional implementations with the same objective or identical physical components supporting the same function in the overall system. On the other hand a mode can be defined by changed purposes. Especially during operating procedures like start-up or shutdown the overarching purpose will change throughout the procedures steps, where the same physical components have to be brought to the state of being capable of realizing the respective function during full operation.

Table 7.1: Mode types based on means-end relations in MFM





objective ↑ function	A mode is determined by the set of functions achieving a given objective. Multiple combinations of functions could achieve the same objective.	objective ↓ function	A mode is determined by the objective that a given set of functions is related to. A given set of functions could achieve a number of different objectives.
function ↑ structure	A mode is determined by the set of physical components that realizes a given function. Different component sets could realize that same function.	function ↓ structure	A mode is determined by the function that a given set of physical components realizes. The same set of components could realize different functions.

7.2 Objective Related Modes in MFM

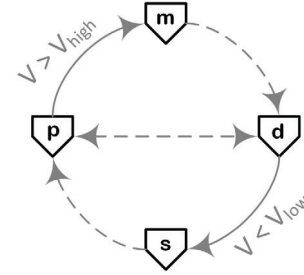
In the context of operating procedures modes can be interpreted as stages with the purpose of establishing a required plant state. The initial study by Lind et al. (2012) showed that if the function or objectives of the plant change each mode requires a different MFM model to be represented correctly. However, consistent functions between modes are not easily identified in an approach that requires an independent model for each mode, and ideally the intention of each mode would be incorporated to explicitly show the mode boundaries by modelling elements. The study presented in Paper G investigated the operating modes during start-up of a thermo-electric power plant. The study revealed that the boundaries of each mode in the considered procedure can be explicitly represented by control objectives in MFM. An approach based on meta-models derived from a common reference model is proposed to facilitate consistency between models of the same plant in different modes. This section outlines the study and the proposed approach to representing operating modes in MFM.

7.2.1 Interpreting Control Function

Four different control functions are defined in MFM. While maintain and suppress are static in the way that they preserve an existing state, the functions of produce and destroy have a defined end state that can be related to operating goals of a given mode. If the end state of one of the latter controllers is reached the control function should logically progress from produce to maintain, and from destroy to suppress. The control functions and the possible sequences are shown in Figure 7.1.

Intention	action	symbol
produce	$[\neg p T p I \neg p]$	
maintain	$[p T p I \neg p]$	
destroy	$[p T \neg p I p]$	
suppress	$[\neg p T \neg p I p]$	

(a) MFM control functions and their action interpretation defined by Lind (2005): state p or its absence $\neg p$ transformed (T) into new state instead of (I) the natural system behaviour.



(b) Implicit (solid) and deliberate (dashed) transitions, for controlled storage

Figure 7.1: MFM control function definitions and sequences

Based on above definitions of control functions only the end state of produce and destroy type control implementations could serve as a boundary descriptor in MFM

models. Additional boundary conditions can be established based on the implicit meaning of the represented control objective. Producing, i.e. increasing, a storage level for instance implies that there is an intended inflow into that storage, that could be attributed with thresholds according to the operation specifications. The MFM model could be analysed to explicitly reason about these constraints as shown in Figure 7.2. The actual operating procedure can then be described by the transitions between modes that follow the violation of any of the given constraints illustrated in Figure 7.2.

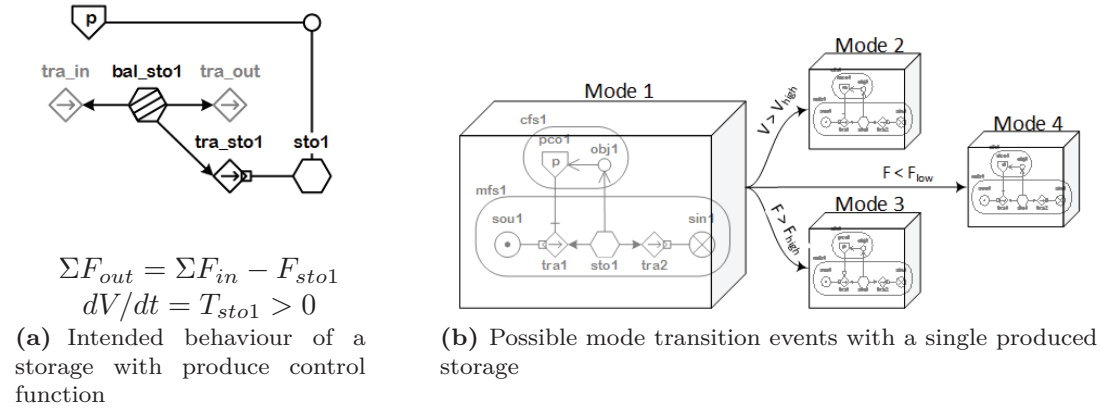


Figure 7.2: Interpretation of controlled function and resulting transition constraints

7.2.2 Modelling Multiple Modes

Gofuku et al. (2006) and Inoue et al. (2015) use additional operating information on component behaviour and alternate functions of components in the system in addition to an MFM model to derive operating procedures. These alternate functions reflect the available structure to function relations, i.e. the possible modes of operation for each structural component. In the simplest scenario such behaviours are the binary states of a valve which is either open with the intention of transporting mass, or closed with the intention to prevent mass flow, i.e. functioning as a barrier. Other components might be entirely offline, removing entire flow structures and their contribution from the system.

Based on that distinction the enable and disable control relations are considered to identify whether a valve acts as a transport or barrier, corresponding to an enabled or disable relation on the specific function. The same logic can be applied to entire structures and the disablement is also propagated along the means-end dimension, e.g. a disabled structure that would provide the means for an energy transport would be interpreted as disabling the corresponding energy transport. Effectively the disabled transport is replaced with a barrier function to maintain a valid model. By way of disablement a reference model containing all possible flow paths and actuation points

can be adapted for a specific mode. The adapted reference model can be seen as a technical meta-model the content of which corresponds to the directly developed model for a given mode. Each mode can then be described by a set of control functions and relations that is adapted with respect to the reference model.

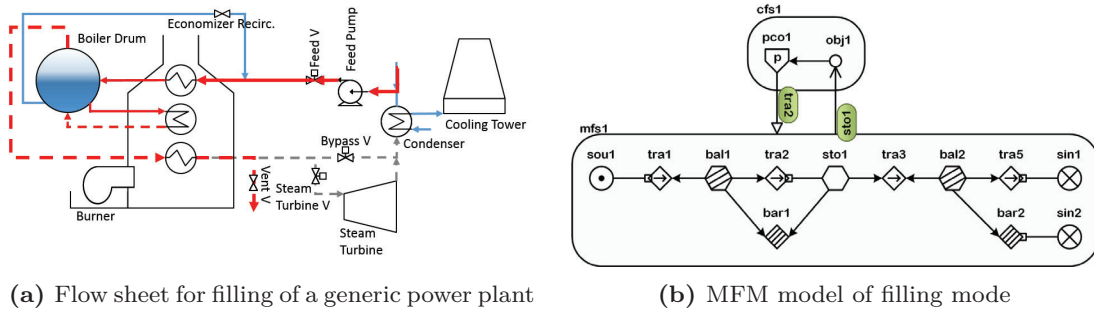


Figure 7.3: Example of an operating mode for a generic power plant

One stage of the start-up procedure of a fictional thermo-electric power plant is selected to illustrate how the proposed approach works. Figure 7.3(a) shows the flow path during the early stage of power plant start-up when the boiler is being filled with water. The boundary of this mode is represented by the produce control function related to *mfs:sto1* representing the water level in the boiler drum. The direct MFM model of that mode is shown in Figure 7.3(b). The complete reference model of the generic power plant in Figure 7.4 contains all possible flow paths and implemented control loops and is adapted according to the control functions and relations representing a specific mode.

Based on the set of control functions in *cfs1* - *cfs4* and the corresponding control relations describing the fill mode, the reference model is adapted, removing inactive parts from the model for reasoning. The only resulting active flow path is in the mass flow *mf1* from the inlet *sou1* through the boiler to the vent valve *tra5*. Both the boiler mass flow *mfs2* and the energy flow *efs1* do not have any active flow paths by being completely disabled or not having any open inlet, respectively. In effect the direct model is identical with the active parts of the meta-model. The main difference between the meta-model and the direct model is the retained energy flow *efs1* which is not relevant to the operating mode. However, since no source is connected in the energy flow and the majority of transports is disabled there is effectively no propagation in the energy flow. This reflects that the filling is done once the water level in the boiler is nominal.

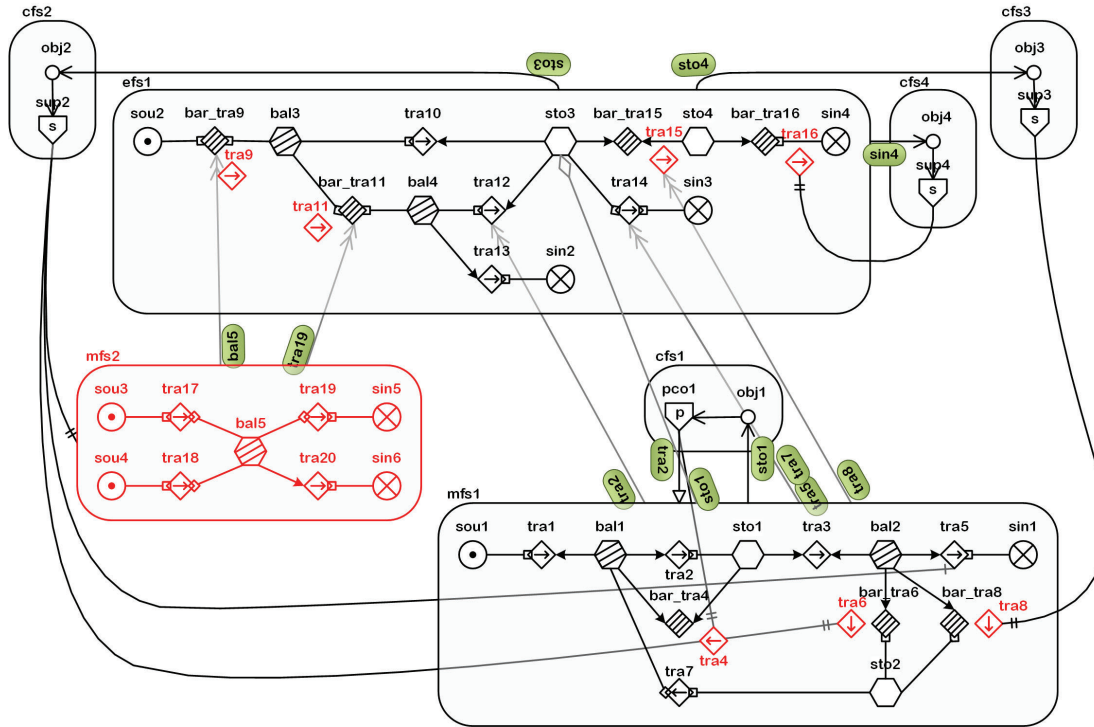


Figure 7.4: Meta-model of the fill mode. Red parts are inactive for reasoning, barrier functions replace inactive transports.

7.3 Modelling Redundant Components

MFM specifically represents the causality between functions fulfilled by the system. A change to one function may affect the availability or capacity of other functions, e.g. by the realizing component undergoing a maintenance procedure and thereby limiting the possible load on the system. This section discusses alternatives of how to incorporate these adaptation into an MFM model and proposes a method of consistently representing the functional contribution of redundant systems.

7.3.1 Aggregation of Redundancies

Connecting the physical components with their function in MFM requires to identify the sensors and physical quantities that correspond to the function in the overall system and can serve as indicators for fault situations. In systems with redundancy multiple components realize the same set of functions and their contributions can be combined to determine the state of the functions. Typically the functional representation of a component consists of a number of causally connected functions with different mass and energy perspectives.

On the one hand relevant system functions can be realized by a varying set of redundant components, reflected by the *function to structure* perspective of modes. For each functional flow perspective the set of contributing physical quantities can thus be determined and aggregated to reflect the correct qualitative state of the functions as reflected in Figure 7.5(a). The function can be accurately represented in MFM for a single unit, since the purpose of redundancy does not contribute to the process functions but rather supports aspects of reliability and economic considerations (Crowl and Louvar, 2011). However, the causality between functions of the same component does not necessarily correspond to the causality of aggregated physical quantities. For example the temperature and pressure are closely coupled in a closed vessel but making any inference from an aggregated representation, e.g. averaged temperature measurement and average pressure, will likely inhibit the detection of individual faults and thereby prevent an accurate qualitative diagnosis.

On the other hand redundant systems are not necessarily identical and combinations of different functional implementations can achieve the same purpose. Interpreting redundancy in this *objective to function* perspective the functions realized by each of the redundant component contribute to a common objective as shown in Figure 7.5(b). In this approach flow functions and physical quantities directly correspond as each physical component is related to an individual functional representation and the MFM model will yield valid inferences about individual deviations. Determining the contribution and effects of faults on the overall system is then a matter of qualifying the resilience of the objective fulfilment to loss of supporting functions. This approach should be favoured as it eases the modelling and mapping of physical components and makes it possible to consider a variety of functionally different implementations fulfilling the same objectives.

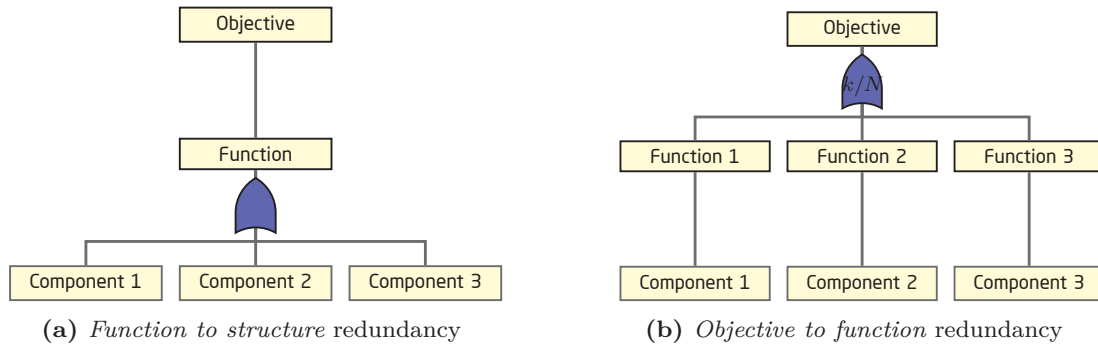


Figure 7.5: Potential interpretations of redundancy in MFM

7.3.2 States of Objectives

The relation of fulfilling functions and objectives has an important role in MFM models and provides more flexibility to model a wide variety of systems. Assuming that

all redundant components contribute equally, the resilience of the objective to a fault can be described by how many of the redundant implementations are unavailable, i.e. failed or in a different mode. This approach fundamentally resembles the voting OR gate established in reliability analysis and risk assessment where fault trees can incorporate redundancy considerations by specifying how many faults in a redundant design will affect the respective top event (Haasl et al., 1981). The voting OR gate is commonly attributed with a fraction k/N interpreted as the number of simultaneous faults k out of N redundant components, where $k \geq N$ will cause the fault to propagate further through the system. In MFM this consideration can be done from both perspectives of either goals or threats. The perspective of threats effectively resembles the consideration in fault trees whereas the goal perspective defines a threshold above which operational performance is affected without posing a safety risk to the plant. Accordingly different fractions will be meaningful for either of the perspectives. Redundant units that are not in the correct operational mode to support the common objective have to be interpreted as not contributing to the goal and not preventing the threat, i.e. deduce $1/N$ from the objective achievement and add $1/N$ to the risk of the threat.

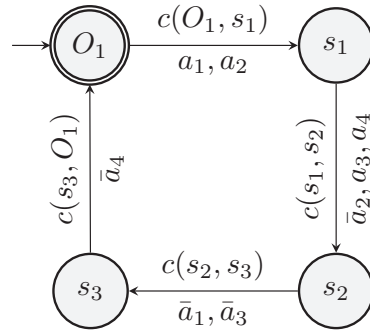
7.4 Operating Procedures and Data

As shown in the previous section, modes can be stages in an operating procedure and the constraints of each of these stages can be explicitly represented in an MFM model, as long as they refer to considered elements such as mass or energy levels and flows. The knowledge about each stage's constraints serves as a mechanism to adapt the model, but requires an additional representation of the possible transitions once any of the constraints are violated. In addition to constraints on physical variables operating procedures frequently use time constraints to trigger the progression from one stage to the next, i.e. the execution of a procedure step. In the context of alarm management, sequence alignment and sequence identification have been proposed for alarm flood recognition and diagnosis (Guo et al., 2017). Apart from alarms, control systems log operations data about state changes, and automatic as well as manual commands to the plant, which can be analysed to track operating procedures.

This section presents the method to represent and validate the constraints associated with the execution of the operating procedure introduced in Paper B. An automaton representation of documented operating procedures is suggested. An efficient method of validating the procedure representation against operations data is described. The validation of the procedure representation is then demonstrated with data from the industrial plant.

7.4.1 Representing Operating Procedures

The steps of an operating procedure can be considered as transitions between states of the plant. The whole procedure can be represented as a finite state machine or



(a) Automaton of example procedure.

Time	Event	
1.0	a_1	
5.0	\bar{a}_1	
5.0	a_3	
5.0	\bar{a}_4	T_{s_3, O_1}
8.0	\bar{a}_3	
10.0	a_1	T_{O_1, s_1}
10.1	a_2	
14.5	a_5	
14.5	a_3	T_{s_1, s_2}
14.5	a_4	
14.6	\bar{a}_2	
20.1	\bar{a}_1	T_{s_2, s_3}
20.1	\bar{a}_3	
24.0	\bar{a}_4	T_{s_3, O_1}

(b) Example event log with occurrences of procedure steps

Figure 7.6: Example of an operating procedure represented in an automaton and an event log excerpt

automaton where the modes are reflected as states and procedure steps are marked by transitions. Each transition $T = \{c, \mathbb{A}\}$ is characterised by set \mathbb{A} of actions performed on the plant as well as the constraint c to be met before it is executed. Any specific operating procedure taking the plant from one operation point O_i to another O_j is reflected as a valid trace in the automaton (Lunze, 2004). During procedure execution intermediate states s_1, \dots, s_n will be traversed, where each transition $T_{s_k, s_{k+1}}$ is typically characterised by monitoring a single variable or a specified timespan before proceeding. The initiating constraint of the procedure, T_{O_i, s_1} , is related to the previously discussed constraints of mode O_i and is likely a composed state described by a number of interleaved process values and states. Figure 7.6(a) illustrates an example of an automaton for e.g. a maintenance procedure that takes the system from mode O_1 back to the same mode. An automaton can be established directly based on the operations manual or similar documentation of the plant.

7.4.2 Identifying Procedure executions

To determine the normal execution pattern of an operating procedure the event logs from the plant can be used to establish a reference. Identifying the sequence and

Time	a_1	a_2	\bar{a}_2	a_3	a_4	\bar{a}_1	\bar{a}_3	\bar{a}_4	a_5
1.0	1	0	0	0	0	0	0	0	0
5.0	0	0	0	1	0	1	0	1	0
8.0	0	0	0	0	0	0	1	0	0
10.0	1	0	0	0	0	0	0	0	0
10.1	0	1	0	0	0	0	0	0	0
14.5	0	0	0	1	1	0	0	0	1
14.6	0	0	1	0	0	0	0	0	0
20.1	0	0	0	0	0	1	1	0	0
24.0	0	0	0	0	0	0	0	1	0
	T_{O_1, s_1}		T_{s_1, s_2}			T_{s_2, s_3}		T_{s_3, O_1}	

Figure 7.7: 1-out-of-N mapping of the alarm log with identified transition occurrences

timing of the normal execution could be done by aligning the event sequences of all occurrences of the procedure in the logs, for instance by the method proposed by Lai and Chen (2017). As pointed out by Guo et al. (2017) complete sequence alignment is computationally expensive, instead they suggest a simple matching of events in a given time window as a coarse indicator for sequence similarity. In line with this approach, the action set of each transition are assumed to appear in a narrow time window in the event log, which allows to create a shorter transition log. The transition log is composed from the occurrences of complete action sets within a defined time window. This is facilitated by a 1-out-of-N mapping of the event log based on the actions relevant to the procedure. Sometimes the recorded executions consistently omit a small subset of the documented actions for a step, e.g. due to operator habits, or undocumented procedure changes. To account for these situations, the largest number of actions within the time window for a given transition is accepted as representing the complete procedure step. Figures 7.6(b) and 7.7 illustrate the translation from an event log through the 1-out-of-N mapping. The transition log corresponds to the marked boxes for each action set with the first time stamp first of any associated event.

The actual procedure executions are considered to be delimited by the first and last transition in the procedure. If one of the delimiting transitions consists of only an individual action, the sliding window approach is adapted again to rely on the first or last pair of transitions, respectively. By way of ensuring a set larger than one action the procedure delimiters are more likely to be distinguished from unrelated isolated events. With the procedure candidates determined as the windows between corresponding start and end delimiters, it is then possible to analyse the implemented execution of the procedure from the transition log.

The sequence of transitions inside each execution candidate window represents a trace in the automaton. If there are no discrepancies between the implemented

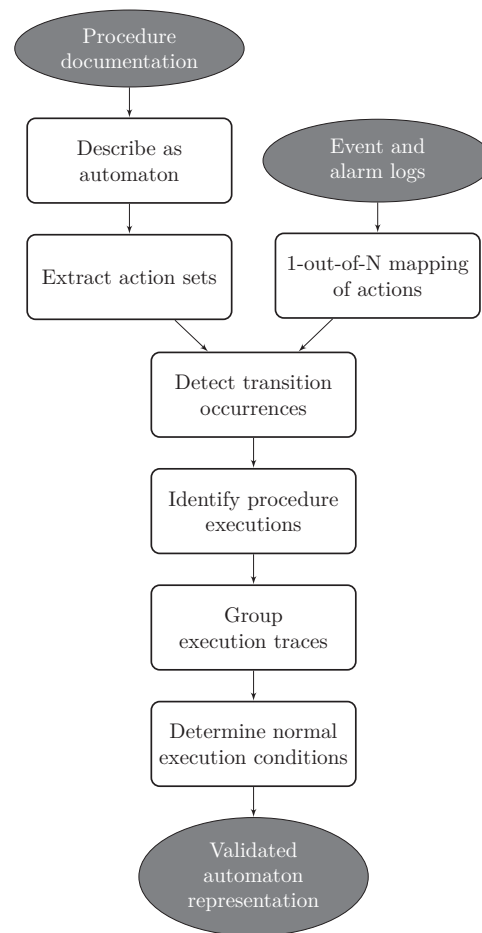


Figure 7.8: Blockdiagram of procedure identification and validation approach

and documented procedure the majority of detected traces should be valid in the automaton derived from the plant documentation. Otherwise the automaton and tentatively the corresponding documentation have to be revised to match the identified trace. The timing statistics across all execution candidates following the same trace can then serve as reference for the normal execution of that procedure. If the procedure execution relies more on process variables than timing, the corresponding variable thresholds could be considered instead. As the transition log is comparably short and focused on the unit under investigation in contrast to the entire event log the analysis of timing and averages can be done efficiently. The procedure execution analysis outlined here is reflected in Figure 7.8

7.5 Industrial Case Study

A sea water fine-filter system with four parallel units is investigated with regards to its operational procedures and mode changes. As it is a redundant system where the individual units can undergo maintenance independently the representation of redundancy becomes very relevant. All four units are designed identically with a capacity of up to 33 % of the total flow capacity, i.e. at least three out of the four units need to be operational at all times. A flow sheet of one of the units is shown in Figure 7.9. The filters share the supply upstream of CV1 and the outlet downstream of CV4, as well as the water or air supply through FCV11 or CV17, respectively. The filters accumulate particles in the filter bed during operation which need to be back-washed once the differential pressure becomes too high, to prevent damage to the filter bed.

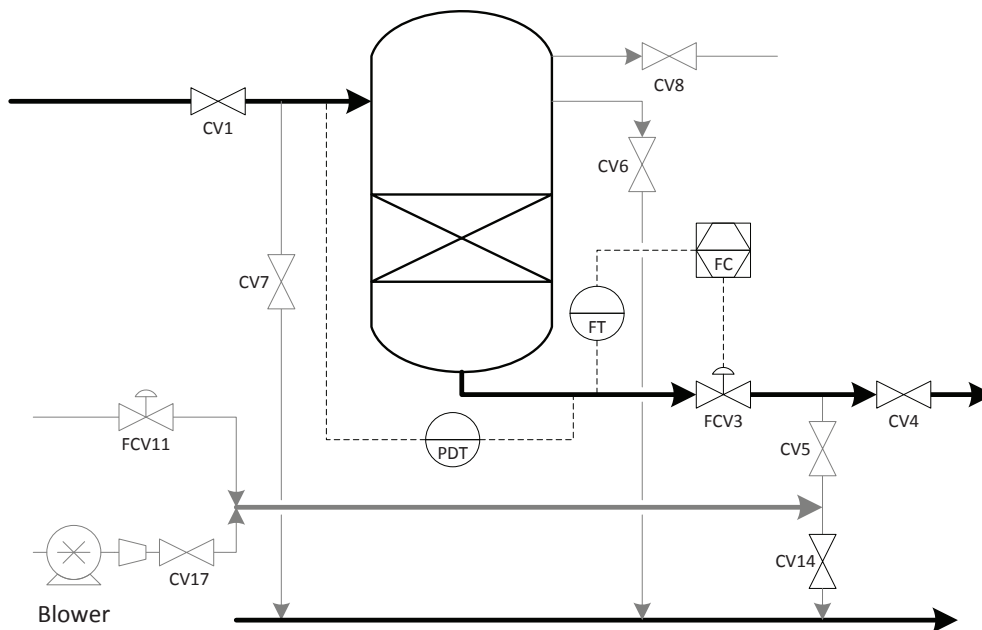


Figure 7.9: Flow sheet of the studied filter unit

The operations manual describes back-washing procedures for either water only, or air scouring followed by back-washing with water. The procedure detailed in the operations manual is described by the automaton in Figure 7.10 and the corresponding action sets in Table 7.2. Per documentation, air-scouring will only be executed in rare cases when the filter is severely clogged while normal back-washing has to be done at least every 24 hours.

A severe plant upset that eventually leads to a safety system trip can be traced back to this filtration system as one root cause. One situation in the filtration system can arise from one filter unit getting stuck in the back-washing sequence and not

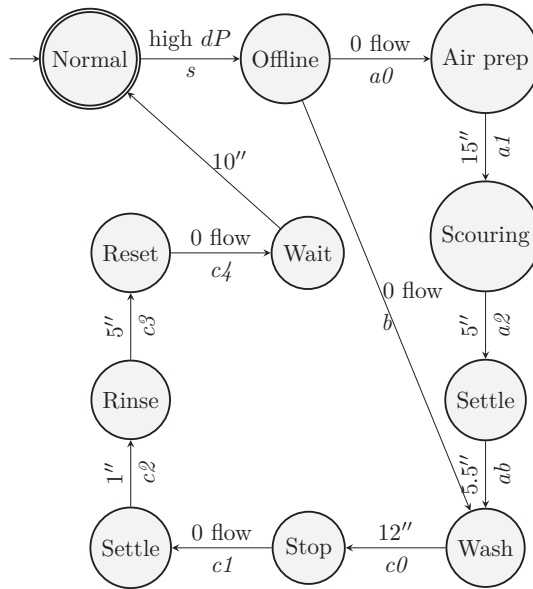


Figure 7.10: Automaton of the filter unit. Edges are labelled with the condition or time in minutes (") before the transition occurs and the transition name.

Table 7.2: Mode transitions of an individual filter unit

Step	Action set
<i>s</i>	3:Off
<i>a0</i>	1:Off, 4:Off, 6:On, 8:On
<i>a1</i>	3:On, 6:Off, 14:Off, 5:On, 17A:On, 17B:Off
<i>a2</i>	17A:Off, 17B:On
<i>ab</i>	8:Off, 7:On, 11:On
<i>b</i>	1:Off, 4:Off, 14:Off, 3:On, 5:On, 7:On, 11:On
<i>c0</i>	3:Off, 11:Off
<i>c1</i>	5:Off, 7:Off
<i>c2</i>	1:On, 5:On, 14:On, 3:On
<i>c3</i>	3:Off
<i>c4</i>	5:Off, 4:On, 3:On

returning to normal operation. Over a long period of time the other filters will capture increasingly more solids to the point that eventually all three remaining units would cross the threshold corresponding to a "high DP" alarm. The system is designed to shift the load to the less clogged filters once the high differential pressure limit is reached, until the newly back-washed unit is available again. In the long run, however, the capacity of the clogged filters is not enough to maintain downstream operation and an emergency shutdown is triggered.

7.5.1 Modelling the Operating Modes

The first step to analysing the operational modes of the filter unit is to understand the objectives of each mode. Normal operation and back-washing are the two main operating modes the filter can be in. During normal operation the filter achieves the target of providing filtered water to the downstream operation. The threat to the filter operation is the filter clogging, which functionally corresponds to a high level of captured solids, however, there is no measurable quantity directly representing the clogging. The threshold for clogging and the critical property is instead linked to the differential pressure across the filter. The implemented control system is set to reduce the flow to on increasing differential pressure. Eventually the combination of demanded flow and the captured solids will lead the differential pressure to surpass

the set threshold and thus cross the boundary for normal operation. Figure 7.11(a) represents the direct model for normal filter operation.

During backwashing the target is to remove the accumulated solids from the filter medium. As shown in Figure 7.11(b) the differential pressure does not serve as an indicator in this mode. As the mode is characterised by a destroy control function its boundary is reached once the captured solids are removed from the filter bed. However, there is no physical indicator to determine the progress of back-washing which is also reflected by the operational procedure in Figure 7.10 being defined with time constraints rather than a threshold on a process variable.

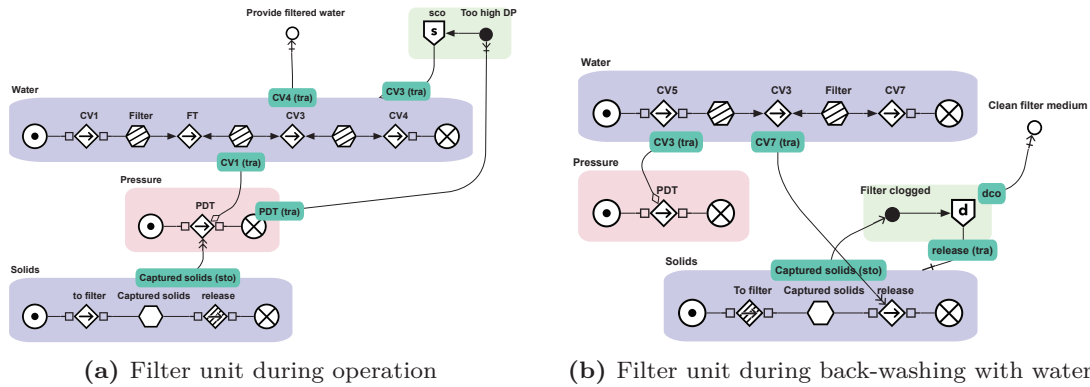


Figure 7.11: MFM models of filter unit in different operating modes

While the explicit boundary modelling in MFM is in principle feasible for this system, the missing link to the physical system in case of the back-washing makes it impractical. This suggests a higher merit of focusing on the automaton representation for mode transitions and maintaining independent models for modes with severe changes in system objectives like the ones treated here. Additionally, the configuration of the control functions is not compatible to establish a common reference model, since the control purpose related to the captured solids varies depending on the available physical indicators. The change from normal operation to back-washing changes the objective of that specific filter unit. In contrast, the operating procedure investigated in paper G works toward the same objective of establishing production, successively providing the necessary functions toward that objective. From these two investigations, it appears that procedures governed by changes of function toward the same objective are well suited for the meta-model approach whereas a change to the objective of a system favours a distinct MFM model per mode.

7.5.2 Modelling redundancy

During operation each filter unit is represented by Figure 7.11(a). All four filter units share the common target of providing filtered water to the downstream process and

the documentation defines that 3 out of 4 units need to be available to maintain operation. In the MFM context this means that the target will be achieved as long as 3/4 of the means-end related functions maintain the objective in a normal state. For the threat of a clogged filter reflected in high differential pressure the operation can tolerate one filter being back-washed and another one reaching the threshold by shifting the load between the units. This means that also 3/4 threat activations are necessary before the threat actually affects the overall plant. In the described scenario this analysis could have enabled the early identification of the filtration system as a bottleneck for the operability of the plant when one unit was stuck in back-washing while the others started getting simultaneous "high DP" alarms.

7.5.3 Validating the Operating Procedures

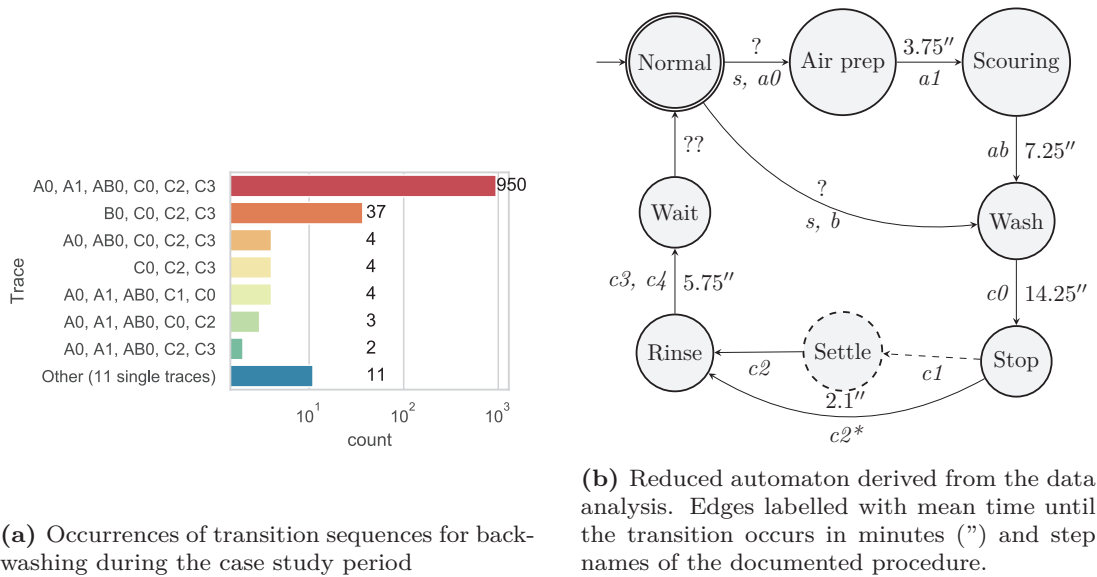


Figure 7.12: Results of procedure identification and validation from event logs

The proposed method of event log analysis was applied for the back-washing procedures documented for the system. The log file for the entire year 2017 for the first of the four units was analysed. Within all execution candidates identified from the first and last transitions the traces were grouped as shown in Figure 7.12(a). The two most frequent traces represent first the back-washing with air scouring and second the back-washing with water only. The validated automaton in Figure 7.12(b) was established as the baseline for diagnosing the correct execution of the procedure. The sequence alignment on a random sample of 50 out of 950 occurrences of back-washing with air scouring additionally revealed controller set point changes in addition to the valve switches considered in the action sets. No other events are consistently

related to the procedure execution, which underlines the proposed methods adequacy to establish a meaningful baseline.

As the operations manual specifies that a high differential pressure in a filter unit initiates the back-washing, the configured "High DP" alarms should be considered as an indicator for the start of the back-washing procedure. If there were a reliable relation between the alarm and the procedure start they should occur at nearly the same frequency during any given period of significant length. However, the results in Table 7.3 show that this relation does not hold due to the interlock of the filter units allowing only one filter to be back-washed at a time.

Event	Alarm	Procedure Start
Frequency	1733	1009

Table 7.3: Occurrences of "High DP" alarm and back-washing procedures during 2017

7.6 Summary

This chapter discusses the representation of operational modes and operating procedures in the context of MFM. Methods for mode representation, relating redundant components to MFM, and representing and tracking operating procedure executions for model adaptation and diagnosis are proposed. The methods are applied in an industrial case study of a water filtration system with four redundant units.

The case study highlights some difficulties of covering all aspects of mode transitions directly within MFM which supports the proposed method of an additional automaton representation of operating procedures and mode transitions. The states in the automaton can be linked to the respective models for each mode while the transition conditions can be more comprehensively described in the automaton.

The case study further revealed that the representation of multiple modes by adapting a reference model is best applied to modes with a consistent objective. Modelling modes with changing objectives, however, the same approach does not present any benefit in terms of modelling or application and independent models should be favoured. To represent redundant systems, a function-object perspective is identified to be the most versatile representation. The means-end relation can then be attributed comparable to a voting OR gate to represent the design limits of the redundant system.

CHAPTER 8

Conclusion

This project established and improved analysis and modelling methods for the situation analysis in industrial plants based on on-line events, i.e. alarms and control system signals. A new inference method for causes and consequences in qualitative functional Multilevel Flow Models has been proposed, which leverages the connection of occurring events to improve computational efficiency. Various approaches to visualisation, interpretation and root cause ranking based on the inferred results have been suggested and compared. Additionally, the implications of changes in the plant have been investigated. Considerations for the adaptation of the qualitative model to accommodate and track changes in the plant due to operating procedures have been discussed. All suggested methods have been demonstrated in case studies with relevance in chemical and petrochemical industry. In the following sections the project is summarized in relation to the research objectives and a perspective on continued research and application based on the findings is given.

8.1 Summary of the Project

The review of established alarm management and decision support approaches summarised at the beginning of this thesis outlined the different knowledge representations applied across the proposed approaches in the domain. MFM has been identified as the knowledge representation with the most varied applications, highlighting its applicability for the proposed advanced operator support system.

To facilitate the real-time application of MFM an improved inference method has been developed, that utilises the connection of occurring events and reduces repeated computation of identical inference paths by merging those paths. The reasoning system is implemented to cover both the causal inference and the updating of the results, including the removal of no longer present observations and associated inferences. Performing the reasoning on-line facilitates the dynamic adaptation of the model depending on the state of the plant. A research tool to produce and visualise the results from the reasoning system has been implemented, which provides a suite of representations of the results. In addition to traditionally considered fault tree inspired representations a number of connected graph visualisations are presented and discussed. A case study on the Tennessee Eastman process showed significant improvements over previous implementations of the reasoning system in a developing situation with connected alarm events.

In the effort to support operators in analysing occurring situations, the comprehensive list of causal scenarios generated from the qualitative reasoning has to be condensed to a manageable list of meaningful situation assessments. This thesis sum-

marised two fundamental approaches to ranking potential root causes and providing the most likely root causes. The first ranking approach calculates a relevance score based on the sum of explained process events weighted to increase the score of closely related causes over causes with a longer causal propagation path. The other proposal is to transfer the qualitative model and inference results into probabilistic BBNs, opening the potential to incorporate prior knowledge on fault probabilities in the diagnosis and determine other metrics developed for BBN. Both methods have been demonstrated to favour the actual root cause in a developing abnormal situation of the Tennessee Eastman process.

As operating procedures dictate changes to the plant operation they are also relevant to a diagnostic system, that needs to be capable of following these changes as well as being able to diagnose errors in the procedure execution in addition to the process diagnosis. The relation of functional MFM models and operating procedures has been discussed and a method for consistently modelling successive procedure steps has been proposed. With the consideration of operating procedures executed on redundant subsystems, e.g. for maintenance, the representation of each sub system at a function level is proposed. The approach has been suggested to incorporate the failure resilience of the redundancy within the means-end structure of MFM. Finally, an automaton representation of documented operating procedures has been proposed. To identify and incorporate discrepancies between documented and implemented operating procedures introduced by operator convenience or experienced efficiency, a data based analysis of the procedure execution has been presented. An industrial case study showed the viability of the redundancy modelling. The case study demonstrated the high efficiency of the proposed data analysis to establish a valid reference for normal executions of an operating procedure. These methods provide the means for tracking the plant state throughout operational procedures as well as the correct execution of the procedure itself.

8.2 Perspectives for Functional Modelling based Operators Support

Within the operator support system outlined in Chapter 1, this project contributed with the real-time inference, the situation evaluation based on the resulting causal analysis, and considerations for incorporating operating procedures and modes. The proposed methods depend on the availability of valid MFM models representing the relevant knowledge. The recent work in continuation of Nielsen et al. (2018b), (2018) provides means of validating MFM models against simulation data or operator experience, ensuring the quality of the model. Lind (2017) outlined the principles of providing modelling libraries based on validated sub-system models. These principles are currently applied in the domain of petrochemical industry. In this project alarms were assumed to be reliable indicators for faults in the plant. However, the difficulties of proper alarm design and management highlight the need for more elaborate

fault detection. Hallgrímsson et al. (2019) presented a machine learning approach to dynamic fault detection. A quantitative approaches like this can improve the identification of fault states at a sub-system level and feed into the qualitative analysis at a plant wide perspective as presented in this work.

In extension to the root cause analysis presented in this work, the operator support system should provide the operators with guidance on mitigation strategies. Ongoing work at DTU investigates the use of MFM models for counter-action planning. The reasoning system could be further elaborated to combine the inference of plant observations with possible counter-actions to get a qualitative forecast of a strategy's viability, which was not in the scope of this work. The work summarized in this thesis comprises methods to apply MFM modelling and reasoning to analyse alarms and operations data occurring in a control room on-line. The methods provide core functionality required to develop an advanced operator support system.

Publications

Dynamic Reasoning in Functional Models for Multiple Fault Diagnosis

Denis Kirchhübel¹, Morten Lind¹ and Ole Ravn¹

¹Department for Electrical Engineering, Technical University of Denmark

Abstract:

Human operators are in charge of the supervisory control of most industrial plants. To provide meaningful support to the operators and avoid information overload root cause analysis of alarm situations has been proposed. Causal graphs are used to represent the cause and effect relations between parts of the system. Thus, the ways faults, indicated by alarms, propagate through the plant can be analyzed. We present an efficient approach to infer causes and consequences for multiple alarms based on a causal graph. Root causes are identified by the dynamic reasoning about observed faults and a ranking of most likely root causes is proposed. The efficiency of the inference and ranking methods is finally demonstrated on an industry process.

A.1 Introduction

Industrial processing plants incorporate many interacting control loops and concurrent processes affecting the productivity and safety of the system. Any modern plant relies on automatic control practices for individual components and processes. In contrast, plant-wide control often faces many uncertainties arising from the environment and interconnected processes. Thus, human operators supervise the vast majority of plants in the energy, petrochemical and chemical industries. To analyze an abnormal situation, operators rely on alarm systems. These systems identify a deviation of the process from the nominal operation parameters and respond by generating an alarm signal about the specific deviation, recording the alarm signal, and informing the operator about the raised alarm. To help the operators focus on the pertinent deviations, rigorous alarm management is recommended for these industries, because of the large risks associated with failures not being detected or not being reacted upon EEMUA, 2013.

Alarm management has been developed in order to reduce the amount of irrelevant alarms. Alarm management procedures scrutinize the necessity and importance of the most frequent alarms. Consequently, where possible, redundant alarms are

D. Kirchhübel et al. (2019b). “Dynamic Reasoning in Functional Models for Multiple Fault Diagnosis”. Computers and Chemical Engineering. submitted in April 2019.

combined or removed. A well-maintained alarm system can avoid operator overload during normal operation, since it is not affected by the most common alarm system challenges: nuisance alarms caused by fluctuations in the process, standing alarms e.g. due to parts of the plant being off-line, and ambiguous alarms which are not immediately actionable by the operator (Rothenberg, 2009; Soares et al., 2016). However, emergencies frequently generate cascades of true alarms throughout the plant that overwhelm the operator with so called alarm floods. To cope with such situations, the relationship between occurring alarms needs to be examined and compiled into concise information to aid the operator in identifying the most relevant and immediate threats. To ensure the safe operation of an industrial plant advanced alarm analysis is necessary in addition to alarm management. (Beebe et al., 2013)

To help operators to quickly identify the relevant information during alarm floods, advanced alarm analysis methods have been proposed. Historical data analysis has been established as being a useful tool to predict certain critical situations, for example by analyzing a sequence of alarms during an alarm flood, e.g. presented by Zhu et al. (2016). However, a critical situation must be well documented and recorded to provide sufficient data for the analysis and to subsequently recognize an equivalent situation. The plant layout is a valuable source of information regarding the connections between different process units and the alarms linked to them. However, the direction and nature of causality between deviations in connected process units is only implicitly represented in the plant documentation (Schleburg et al., 2013). Causality allows a more accurate analysis of situations with many linked alarms (Rodrigo et al., 2016). For an automated support system the causality needs to be represented in a machine readable format that can efficiently be analyzed. Signed directed graphs (SDG) is an intuitive and machine readable representation of causality and can be based on measures of correlation and causality as well as process documentation and operator knowledge (Yang et al., 2014). Multilevel Flow Modeling (MFM) has been proposed as a versatile process representation to analyze the causality of propagating deviations in a plant (Lind, 2013), connecting the physical plant and operator tasks in the analysis. SDG as well as MFM provide the means of diagnosing an abnormal situation (Dong et al., 2010; Hu et al., 2017b; Wang et al., 2014). The causal diagnosis of a situation can help the operators focus their attention on the actual causes of the situation rather than symptoms.

Performing advanced analyses of alarms and improving the performance of the operator interface require knowledge of the process. Connectivity of system parts and causality of influence between process variables are important in the context of alarm management and decision support. Figure A.1 outlines methods proposed for efficient knowledge representation. Ontologies of processes usually represent a classification of components that can be interconnected (Natarajan and Srinivasan, 2014; Schleburg et al., 2013), but other approaches like linked ontologies for alarms and control actions have also been proposed (Basu et al., 2013). Fault trees and Bayesian networks (BN) are graphs representing causes and consequences, where fault trees represent combinations of causes that lead to a specific consequence in a binary logic and Bayesian networks represent the causal relation by conditional probability. Fault

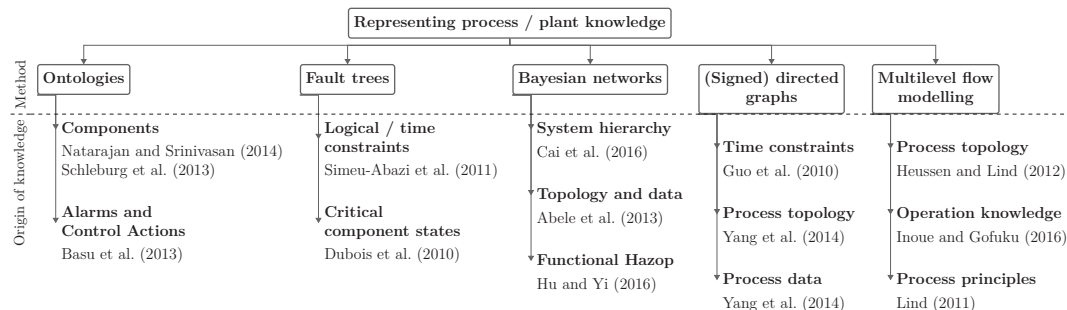


Figure A.1: Summary of knowledge representation methods by source of knowledge

trees are mostly based on an assessment of possibly critical states of a component (Dubois et al., 2010) but can also integrate logical and time constraints (Simeu-Abazi et al., 2011) between faults. BNs are widely used to incorporate operator knowledge, e.g. in a hierarchy of component-wise BNs (Cai et al., 2016) or based on the results of a structured Hazard and Operability Study (HAZOP) (Hu et al., 2015). Abele et al. (2013) propose a combination of component ontology and learned probabilities from historical data. Signed directed graphs (SDG) or their representation as adjacency matrix can be generated from process topology and expert knowledge, or from process data by a variety of correlation and causality methods (Yang et al., 2014). Guo et al. (2010) suggest an SDG representing temporal constraints between alarms, assuming that causality is reflected in the temporal succession of alarms. The functional HAZOP used by Hu et al. (2015) is based on the causal inference in a multilevel flow model. A complete MFM model combines process topology and operation knowledge (Inoue and Gofuku, 2016), but a valid model can also be generated considering the process topology as represented in a P&ID (Heussen and Lind, 2012) and basic process principles (Lind, 2011). While all these methods have different advantages and limitations, a range of alarm management and operator support approaches have been proposed based on each of them (Wang et al., 2016a).

Even though root causes can be inferred based on both MFM (Larsson et al., 2004) and SDG (Zhang et al., 2005), there is a need to accommodate dynamic changes to the model (Kirchhübel et al., 2017b). In this article we present a dynamic propagation method to analyze root causes of multiple observations in a more efficient manner developed for MFM model. Paper A.2 briefly compares the causal representation by SDG and MFM. In Paper A.3 the proposed propagation method is described followed by the proposed root cause ranking. In Paper A.4 the Tennessee Eastman process (Downs and Vogel, 1993) is presented as case study and the set-up for measuring the efficiency of the propagation is outlined. The results of the efficiency measurement and the ranking approach are presented and discussed in Paper A.5. Finally, Paper A.6 summarizes the findings of this article and gives a perspective on further research and future application based on the proposed method.

A.2 Causal graphs

As outlined before, graph representations have been widely proposed as underlying knowledge representation for causal analysis. Specifically, signed directed graphs (SDG) or directed causal graphs (DCG) and Multilevel Flow Modelling (MFM) will be briefly discussed here.

SDGs represent the causal interaction between process variables. Each node corresponds to one process variable, e.g. pressure, temperature, flow rate. The edges are directed from cause to effect, with the sign $+$ (or $-$) of each edge describing an increase (or decrease) in the effect node when the cause node increases, or vice versa. An SDG can be used to identify propagation paths of deviations in a process variable. By traversing the graph backwards, i.e. in opposite direction of the edges, causes for a deviation can be inferred, while forward traversal yields a tree of possible consequences. Most approaches for alarm management or fault diagnosis perform a backward propagation from each observed deviation, limiting the search e.g. by defining a maximum path length (Arroyo Esquivel, 2017) or assuming a small number of unobserved nodes while all observed nodes can be used to validate or disregard a proposed cause (Lv and Wang, 2007). Subsequently the identified root causes are analysed by comparing the observations with forward propagated consequences or by means of data-based methods (Lv and Wang, 2007; Wan et al., 2013).

While MFM also represents causal interaction in the system as a graph, the nodes and edges reflect the functions of the system. These functions originate from the design intention of the process. In an MFM model goals to be achieved and system functions supporting the goal are hierarchically decomposed which is referred to as the means-end dimension. The part-whole dimension reflects the decomposition of each system function into basic material and energy flow functions. Figure A.2 shows the function, i.e. node, and relation primitives available for modelling. (Lind, 2013) The highest level of the hierarchy represents the purpose of the entire system which is decomposed into interconnected structures reflecting necessary supporting functions. To perform an analysis on a MFM model deviations of function states are propagated along the graph based on connection patterns between the functions. Zhang (2015) defines these patterns for both cause and consequence propagation. Further analysis of the root cause candidates gained from backward inference has been proposed using a database of common mistakes (Wang et al., 2016b), a ranking based fuzzy logic (Dahlstrand, 1998) or the order of occurrence (Larsson, 2002)

Both MFM and SDG reflect the causality throughout a process plant. While the nodes in an SDG can be mapped directly to process variables, MFM encodes process variables by the semantic primitives for flows, levels, etc. Figure A.3 shows examples of the mapping of MFM patterns to a SDG representation. Since each type of MFM functions implies a specific behavior a pattern of two functions and their relation reflects how failures are propagated (Petersen, 2000). Similar to a signed directed graph (SDG) (Yang et al., 2014) two flow functions are connected by a relation, that describes how the states associated with the respective functions affect each other. The fundamental concepts of analysis by qualitative fault propagation and the method

























Functions					function structure  <i>energy/ mass / control</i>
Mass and Energy Flow			Control		
source 	transport 	storage 	steer 	trip 	
sink 	barrier 	balance 	regulate 	suppress 	
Relations					Goals
Influence	Means-End	Means-Goal	Control		
influencer 	mediate 	produce 	enable 	threat  objective 	
participant 	producer-product 	maintain 	disable 		
		suppress 	actuate 		
		destroy 			

Figure A.2: Modeling primitives of MFM (Lind, 2011)

proposed here hold in both kinds of models (Yang et al., 2014; Zhang, 2015).

Similar to the decomposition into mass, energy, and information flows in MFM, the dynamic causal directed graph (DCDG)(Arroyo Esquivel, 2017) splits the perspectives for different flows. This decomposition enables a structured approach to building the causal model accessible for domain experts, like process engineers and operators. Since the functions used in MFM are closely related to flow sheets and diagrams commonly used by these experts, MFM is promoted here as basis for incorporating expert knowledge into the operator support system.

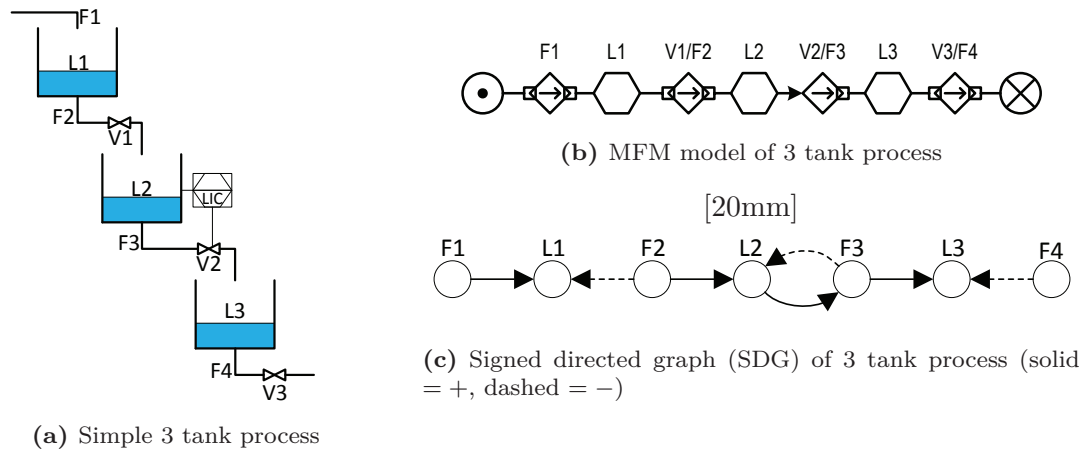


Figure A.3: MFM and SDG representation of simple 3 tank process

A.3 Method

In this section the concepts of fault propagation as well as the proposed dynamic updating and presentation of results are described.

A.3.1 Deviation propagation

For on-line diagnosis the MFM model of a plant needs to be adapted to the current configuration of the diagnosed plant (Kirchhübel et al., 2017b). To allow for dynamic changes to the model a dynamic reasoning system is established, rather than per-compiling the causality for all possible scenarios. In the MFM reasoning only qualitative states similar to alarms of *high/high-high* and *low/low-low* are considered. A negative edge in the equivalent SDG, like in Figure A.3, links a causing *high* to a consequential *low* and vice versa, where the edge is directed from cause to effect. Zhang (Zhang, 2015) presents a comprehensive description of MFM patterns and the respective failure propagation. Iterative propagation of a deviation along the causal connections in the model yields a tree of inferences respectively for causes and consequences of the specific deviation comparable to fault and event trees commonly used to analyze accident scenarios in processing industry.

The scope for the analysis is defined to cover qualitatively observable situations during alarm floods and diagnosis in a plant-wide perspective. While the propagation in any causal model can formally produce cycles, those are rarely meaningful for the analysis of alarm floods. Considering that alarms are usually filtered or delayed to move relatively slowly, oscillations would not be observable based on alarms or be at a scale where the oscillating character is not relevant to identification of causes. The propagation scope is limited by the following properties:

1. *Number of times each node occurs in a path:* Recycles and control loops can cause cycles in the propagation. The same node is only allowed to occur with the same inferred state, which is separately treated as a "loop". Oscillations inferred as the same node with an opposing state are deemed invalid.
2. *Number of times each edge is traversed in a path:* Due to above limitation of node occurrences each edge can only be reached once per path.
3. *Maximum path length* is not restricted in the current implementation.

A.3.2 Inference maintenance

The reasoning is based on two basic concepts, evidence and inference. All observations from the system are considered as evidence which could be alarms or states detected by other methods. An evidence consists of a specific MFM function in the model and its observed state. Starting from each evidence, inferences of possible causes and consequences are generated by traversing the patterns in the model. Each new

inference also consists of an MFM function and the inferred state. An inference is validated by the reasoning system to comply with the scope of the analysis, before it triggers further propagation. Figure A.4 shows the decision tree for inference validation.

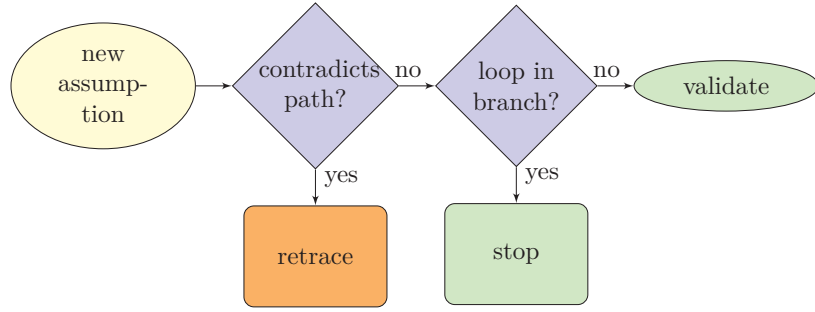


Figure A.4: Validation of Inferences

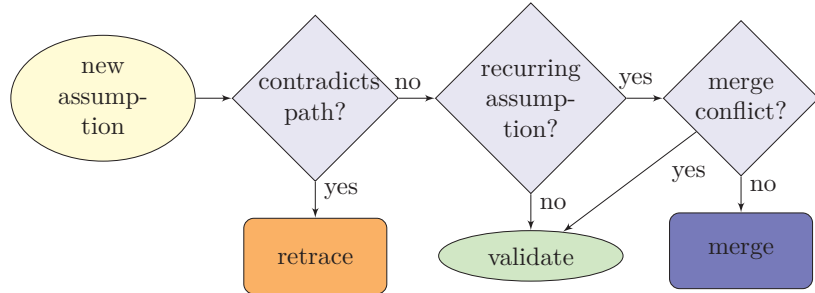


Figure A.5: Validation of inferences with merging

The previous implementations outlined in literature frequently consider one evidence as a root cause and prune the resulting event tree based on other evidence (Zhang, 2015) or identify a fault tree for a given evidence and compile a fault analysis for each scenario (Wang et al., 2016b). In the context of alarm floods, however, the large number of simultaneous alarms does not allow the reliable identification of one root event to base the analysis on. Hence, all plausible analyses have to be run to identify the most likely causal scenario. In previous implementations (Larsson, 2002; Zhang, 2015) this could be realized by running a separate parallel analysis of causes and consequences for an evidence generated from each alarm. The parallel and independent analysis of each alarm, however, disregards that alarm floods are closely related and caused by the interconnection of the system, therefore the analysis for two adjacent alarms will be identical for most parts.

Utilizing the inherent connection of alarm floods a new approach to reasoning about multiple events in MFM is proposed. Propagating two connected evidences e_1 and e_2 independently along the graph will lead to identical inferences a_1 and a_2 , where the causal paths of both evidences coincide. The inferences a_1 and a_2 about the same function in the model, inferring the same state, and were inferred by traversing the same relation. All inferences based on a_1 or a_2 are bound to be identical as well, since the same paths in the graph are traversed with the same deviation. Based on this rationale only one of the identical inferences needs to be propagated and the inferences from e_1 and e_2 are merged at $a_1 = a_2$. Figure A.5 shows the modified validation of inferences.

Considering only relevant inferences is essential for effective operator support. To comply with the defined scope of the analysis mechanisms for retracing contradictory results and merging recurring inferences are outlined in the following. Furthermore, the devalidation mechanism ensuring that no longer observed evidence is not considered in the inferred results is described.

A.3.2.1 Retracing

If an inference leads to a contradictory or oscillating loop in the causal path, the contradictory inference is retraced. Since all inferences are causally linked to the states they were inferred from, the inferences that lead exclusively to the contradiction are retraced as well. The retracing is done iteratively until an inference that leads to more, not-contradictory inferences is reached.

A.3.2.2 Merging

If two inferences a_1 and a_2 about the same function and the same state are inferred using the same relation they can potentially be merged. Assume that a_1 is a valid inference, that has been fully propagated and a_2 is a new inference equal to a_1 regarding the function, state and propagation direction. The inference tree $\mathcal{T}_{a_1} = \{A_1, J_1\}$ inferred from a_1 contains a set of valid inferences A_v concerning functions F_v and a set of contradictory inferences $A_c = \{F_c, S_c\}$, where $A_v \cap A_c = \emptyset$. To ensure the validity of the results, the merging has to comply with the same conditions as any individual inference. The path $P_{a_2} = \{A_2, J_2\}$ leading to a_2 traverses the functions F_2 . In order to preserve the consistency of the resulting inference there can be no contradiction between \mathcal{T}_{a_1} and P_{a_2} . To that end, there can be no contradictions between the path to a_2 and the inferences from a_1 , i.e. $F_v \cap F_2 = \emptyset$. In the same manner all functions in F_c have to form a contradiction with F_2 , i.e. $F_c \subseteq F_2$. With above conditions fulfilled a_2 is effectively replaced by a_1 and the inferences leading to a_2 become justifications for a_1 .

A.3.2.3 Devalidation

To only maintain valid results, a devalidation mechanism as shown in Figure A.6 is implemented. An inference, that is no longer supported is marked as devalidated and subsequently removed from all inferences that were based on it. If no other inferences depend on a devalidated inference it is completely removed from the results. Through dependencies of each inference this mechanism reliably cleans up all results that are no longer relevant.

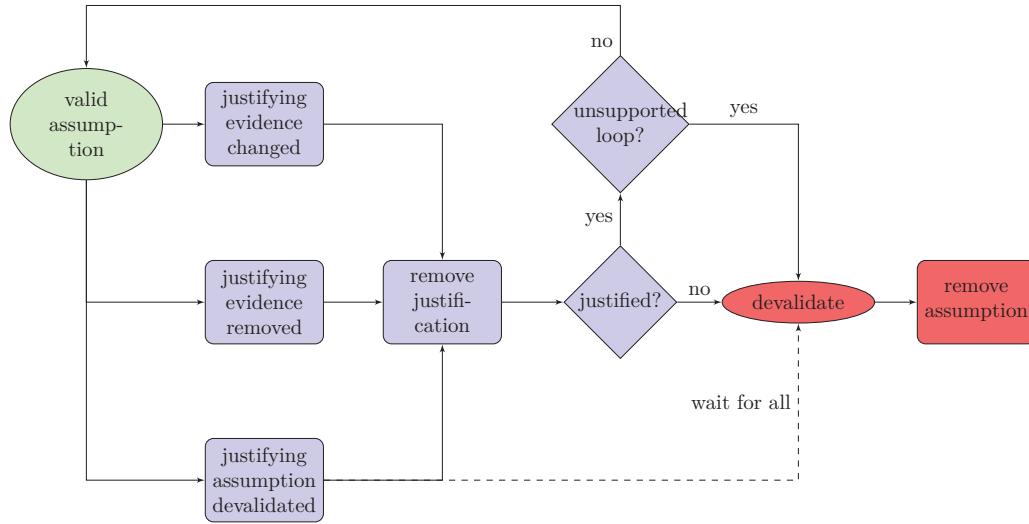


Figure A.6: Devalidation logic for result clean-up

A.3.3 Result interpretation

Previously the results of the inference were mostly represented in a tree form starting from a single triggering evidence and cropped according to all other evidence. The inferred causes or consequences are then represented in a form similar to fault and event tree, respectively. A tree representation, however, is not meaningful when the complex connections of multiple evidences are analyzed by the reasoning method described before. The efficient dynamic update and combination of results is realized by a multi-linked dependency structure for each inferred state. A directed graph representing the causality between evidences and inferred states is a natural choice for the representation. Finding the root cause in this graph is a matter of finding a minimal tree, which includes as many observations as possible, while obeying the directivity of the graph.

The best explanation for a given scenario is the root cause that is common to the inferred cause tree for multiple observed faults. In order to select the most appropriate cause a ranking for the root causes is proposed. Assuming that many of

Algorithm 2 Weighing causes

```

 $R \leftarrow \emptyset$ 
 $w_a \leftarrow 0 \forall a \in A(\mathcal{G})$ 
for  $(a \in A(\mathcal{G}), \bar{\delta}^-(a) = 0)$  do
     $w_a \leftarrow 1$ 
     $M \leftarrow \emptyset$ 
     $N \leftarrow \{a\}$ 
    while  $N \neq \emptyset$  do
         $B \leftarrow \{b \in A(\mathcal{G}), bn_1 \in E(\mathcal{G})\} \setminus M$ 
         $N \leftarrow N - n_1$ 
        for  $(b \in B)$  do
             $w_b \leftarrow w_b + w_e \cdot w_{n_1}$ 
             $M \leftarrow M + b$ 
             $N \leftarrow N + b$ 
        end for
        if  $\bar{\delta}^+(n_1) = 0$  then
             $R \leftarrow R + n_1$ 
        end if
    end while
end for

```

the observations are connected, an inferred cause becomes more meaningful the more of the observations are explicable as its consequence. In addition, a causal connection is less relevant the longer the propagation path between the cause and the observation is. Accommodating these two premises each edge is weighted at $w_e < 1$. Tracing the inference, i.e. considering edges from consequence to cause for backward reasoning, the weight of each node can be calculated sequentially as outlined in Algorithm 2. By this weighing a cause that is closely related to some observations will be ranked higher than a cause that is vaguely related to a larger number of observations. The edge weight w_e can be adjusted to emphasize the relevance of number of explained assumptions or the length of the propagation path. The highest ranking cause can be considered a root cause and a proper explanation of the situation can be compiled as a consequence tree rooted in the common cause and spanning the observed faults.

The weighing is performed by traversing the resulting graph by the algorithm outlined in Algorithm 2. All nodes are initialized to weight $w = 0$. For each evidence, represented by a node with out-degree $\bar{\delta}^- = 0$, the weight is initialized to $w_a = 1$ and the evidence is added to the set N of nodes to be searched, whereas the set of already traversed nodes M is initialized as empty set. Taking the first node $n_1 \in N$ the inferred causes B for n_1 are considered, excluding already traversed causes. The weight of each inferred cause is incremented by the product of the edge weight and the weight of the preceding inference $w_e \cdot w_{n_1}$. The set of root causes R comprises of all inferences that do not have any inferred causes, i.e. $\bar{\delta}^+ = 0$.

A.4 Case study

To validate the improvement by the proposed method, tests are performed on an MFM model of the Tennessee Eastman challenge benchmark process initially presented by (Downs and Vogel, 1993). The control strategy presented by (Ricker, 1996) and the corresponding data available are considered as test scenario (Ricker, 2019). Propagation performance is compared between matching implementations of the propagation algorithms both with and without the proposed merging rules. This section describes the test case and outlines the performance measure applied.

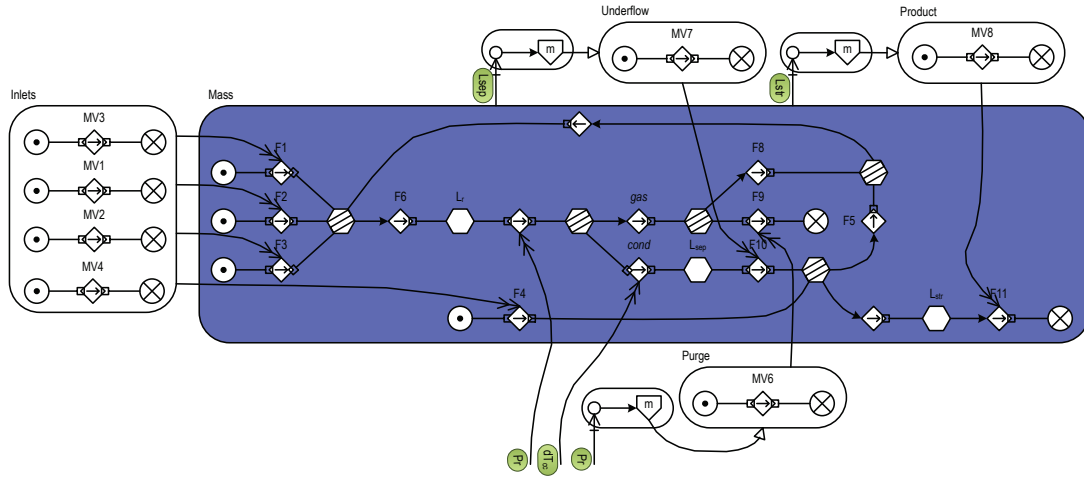


Figure A.7: Mass flow of the Tennessee Eastman process, supporting heat and pressure flows are shown in A.6

A.4.1 Tennessee Eastman process

The Tennessee Eastman Challenge process has been proposed as the basis for comparable studies on plant wide control and fault diagnosis strategies, as such it lends itself well as a demonstration for alarm analysis. To generate alarms the operating limits given by Downs and Vogel (1993) are considered. For the remaining process variables a 2% band around the base case value is set for low and high alarms as suggested by Ma and Li (2017). Table A.1 summarizes the applied thresholds and variable tags. A simplified MFM model is shown in Figure A.7 and in Figure A.10. As a demonstration of the propagation method rather than the modeling framework, only the thermal process is considered in the model. The contribution of the reaction and the control of compositions is omitted in this demonstration.

Table A.1: Process variable tags and alarm thresholds for low (L) and high (H) alarms

Variable number	Description	Tag	L	base case	H	unit
1	A feed	F1	0,24	0,25	0,26	<i>kscmh</i>
2	D feed	F2	3589,00	3664,00	3735,00	<i>kg h⁻¹</i>
3	E feed	F3	4419,00	4509,00	4599,00	<i>kg h⁻¹</i>
4	C feed	F4	9,16	9,35	9,54	<i>kscmh</i>
5	Recycle flow	F8	26,36	26,90	27,44	<i>kscmh</i>
6	Reactor feed rate	F6	41,49	42,34	43,19	<i>kscmh</i>
7	Reactor pressure	P _r	0,00	2705,00	2895,00	<i>kPagauge</i>
8	Reactor level	L _r	50,00	75,00	100,00	%
9	Reactor temperature	T _r	0,00	120,40	150,00	°C
10	Purge rate	F9	0,33	0,34	0,34	<i>kscmh</i>
11	Separator temperature	T _{sep}	78,50	80,10	81,70	°C
12	Separator level	L _{sep}	30,00	50,00	100,00	%
13	Separator pressure	P _{sep}	2581,03	2633,70	2686,37	<i>kPagauge</i>
14	Separator underflow	F10	24,66	25,16	25,66	<i>m³ h⁻¹</i>
15	Stripper level	L _{str}	30,00	50,00	100,00	%
16	Stripper pressure	P _{str}	3040,16	3102,20	3164,24	<i>kPagauge</i>
17	Stripper underflow	F11	22,49	22,95	23,41	<i>m³ h⁻¹</i>
18	Stripper temperature	T _{str}	64,42	65,73	67,05	°C
19	Stripper steam flow	F5	225,70	230,31	234,92	<i>kg h⁻¹</i>
20	Compressor Work	W _c	334,60	341,43	348,26	kW
21	Reactor cooling water outlet temperature	T _{rc}	92,71	94,60	96,49	°C
22	Condenser cooling outlet temperature	T _{cc}	75,75	77,30	78,84	°C

A.4.2 High coolant temperature fault

As scenario the alarm sequence produced by case IDV12 is analyzed. Since the presented MFM model omits the composition of different streams it lends itself best to the diagnosis of a thermal disturbance. This test case introduces a step change to the condenser cooling inflow temperature. Figure A.8(a) shows the evolving alarm situation over the first 7.5 hours of the data set. The offset is introduced at 1:00, leading to an immediate alarm state of low purge flow F9. As the major application of this method is expected to be alarm floods the alarms occurring due to the offset are considered in direct succession rather than the actual time frame to underline the real-time applicability of the method. The applied alarm sequence is shown in Figure A.8(b).

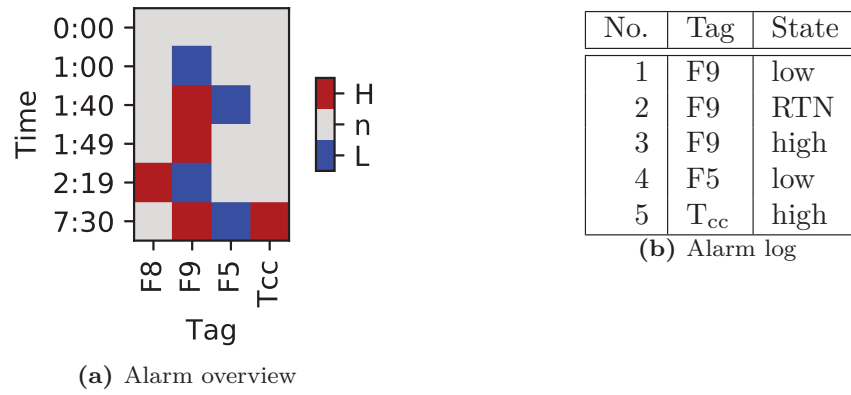


Figure A.8: Evolving alarm states during high coolant temperature situation

A.4.3 Algorithm performance

The propagation and maintenance algorithm is implemented in a Java framework, which implies large uncertainties when evaluating the execution time (Georges et al., 2007). To yield comparable results the execution time of each update step is measured in an individual experiment. The setup phase before each experiment includes all preceding update steps to identify the difference in performance in emerging situations. A bench-marking harness is used to warm-up and evaluate the Java virtual machine. Table A.2 lists the relevant parameters for the benchmark. As a measure of performance the execution time and number of intermediate assumptions for each update of the alarms are recorded. Additionally the proposed ranking is applied to the results.

Table A.2: Parameters for benchmark execution of each update step

Forks	Warm-up runs	Measurement runs	total considered runs
3	3	10	30

A.5 Results and Discussion

To illustrate the efficiency of the proposed method both the proposed merging method and the same propagation implementation without merging are compared. This section presents the results and evaluates the improvements gained by the proposed method. The execution performance is presented, as well as the application of the proposed cause ranking approach. Finally, limitations and perspectives of the presented work are discussed.

A.5.1 Execution performance

The recorded execution performance shown in Figure A.9 underline that the proposed method consistently reduces the number of intermediate inferences by up to 31% in the given scenario. In contrast the execution time of the merging method is longer by up to 8% when propagating an isolated fault, while multiple connected alarms can be propagated significantly faster at up to 29% of the execution time. The discrepancy between number of inferences and propagation time is due to the additional evaluation necessary to ensure correct merges and complete propagation. In the given scenario the reduced propagation more than compensates for the increased evaluation procedure for any causally connected alarms.

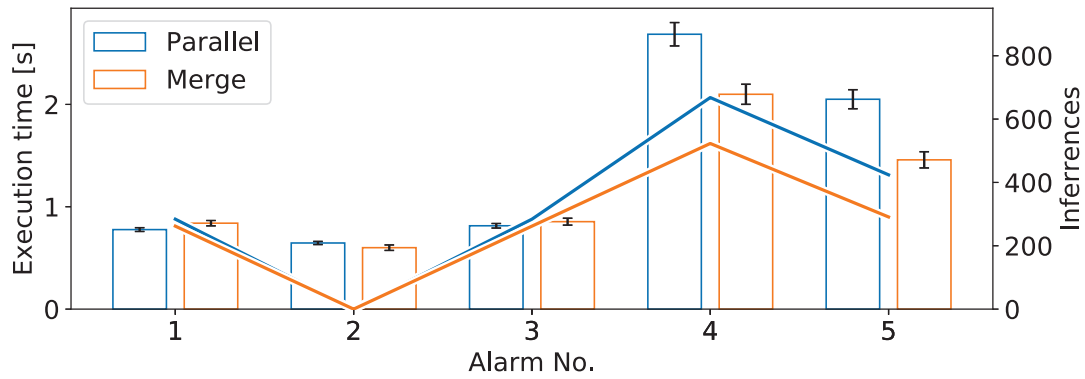


Figure A.9: Execution performance by time for update (bars with confidence interval) and intermediate inferences generated (lines) per update step

A.5.2 Result Ranking

For each update step the propagation results are weighed by the proposed method to point out the most likely root causes. Table A.3 shows how the actual root cause of "high condenser cooling inlet temperature" represented by "Condenser:T_i:tra:high" in the propagation results is weighed after each update step. The connected structure of the results generated using the merging method lends itself very well for this ranking, as root causes concerning the same function and state only appear once in the graph. In contrast, propagating individual paths without merging would repeat the same state and function for each possible path increasing the complexity of calculating the weight and comparing the individual paths.

Table A.3: Root cause rank of actual cause and top ranked root cause after each update step

No.	Position	Weight	Top Cause	Top Weight
1	15	0.377	Low reactor pressure	0.735
3	8	0.599	High reactor pressure	0.735
4	1	1.262	High condenser coolant temperature	
5	1	2.119	High condenser coolant temperature	

A.6 Conclusions and perspectives

The presented method of combining the propagation of multiple observations in a causal graph was shown to significantly increase the efficiency of the propagation for causally connected observations. This is an important step toward the application of causal graphs for real-time analysis in alarm flood situations when many related alarms are caused by a few faults in the system. In addition to the efficient propagation the presented method reduces the size of the propagation result and eliminates identical causal paths, even for single observations. Thus, the further analysis of root cause candidates is easier, as individual causes are not repeated within the result graph. Methods like Bayesian Belief Networks are well established for the analysis of fault-tree structures and can be applied for further analysis, such as updating the likelihood of root cause candidates and identify the most relevant information in order to distinguish root cause candidates by determining the value of information. The fault-trees generated by parallel inference pose a problem for this kind of analysis, as identical paths would occur independently, hiding relevant links when mapped directly into a Bayesian network. In contrast, the graph resulting from the merging method links recurring paths together and can, hence, be directly mapped into a meaningful Bayesian representation.

The proposed ranking of root causes was shown to produce meaningful results and reliably put the actual cause among the top candidates. While the length of the propagation was uniformly penalized in the presented case study, the ranking method provides the flexibility to incorporate available knowledge, like the MFM model syntax by varying the edge weight for different relations hence giving higher penalties on relations representing physical connections and lower penalties on intentionally designed couplings like implemented controllers.

Appendix: MFM model

For easier visualization the mass and energy perspective of the MFM model have been separated. The labels on the open relations indicate how the two parts of the model are to be connected. For example the pressure input to the stripper is realized by the inflow of gas to the stripper, i.e. $Mass:F_4$, which represents the C feed stream.

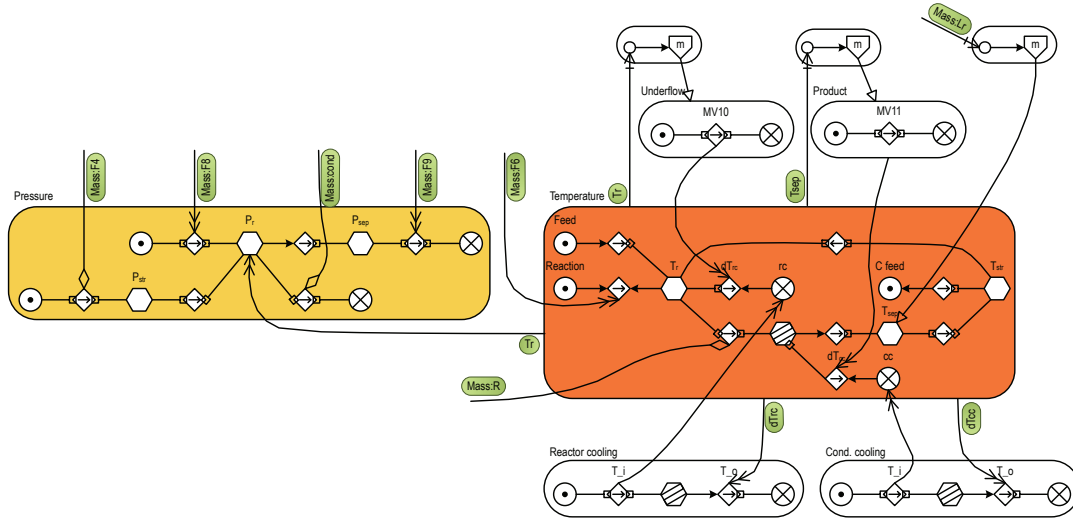


Figure A.10: Heat and pressure flows of the Tennessee Eastman process

PAPER B

Combining Operations Documentation and Data to Diagnose Procedure Execution

Denis Kirchhübel¹, Morten Lind¹ and Ole Ravn¹

¹Department of Electrical Engineering, Technical University of Denmark

Abstract:

Established control room systems in processing industry are prone to overload operators during severe plant upsets. Automatic diagnostic assistants have been proposed to assist operators in high load situations. To ensure a correct diagnosis the assistant system needs to be aware of configuration changes. Standard Operating Procedures detail when and how the plant configuration is to be changed and are established during plant design to ensure consistent and safe operation. Tracking the correct execution of a procedure is relevant to both, detecting errors during procedure execution and adapting diagnostic models. In this contribution we describe methods for representing documented procedures as automata and validating the procedures against log files of the control system. We propose a fast approach of detecting procedure executions and action sets associated with procedure steps. The presented methods are demonstrated in an industrial case study of a filtration system and validated against complete sequence alignment.

B.1 Introduction

Alarm systems are the primary interface for human operators in industrial plants in case of abnormal situations and yet poorly managed alarms can cause severe operator overload and have been identified as the cause for a number of large accidents across different industry domains Goel et al., 2017; Wang et al., 2016a. In consequence guidelines and indicators for successful alarm management have been developed by industry. Norms published by the ISA (2009) and IEC (2014) complemented by guidelines and best practise handbooks, e.g. by EEMUA (2013), Hollifield and Habibi (2006), and Rothenberg (2009), highlight the industrial relevance of alarm management. Studies have shown the effectiveness of rigorously implemented alarm management Soares et al., 2016.

D. Kirchhübel et al. (2019a). “Combining Operations Documentation and Data to Diagnose Procedure Execution”. *Computers and Chemical Engineering*. accepted in Nov 2019 pending revision.

The most crucial failures of alarm systems are alarm floods that overwhelm the operator with a large number of nearly simultaneous alarms. The multitude of information frequently leads the operators to start treating symptoms while missing out on the root causes Hollnagel, 2002. A range of approaches to handle alarm floods have been proposed, with a dominant trend of employing data analysis methods. The common methods include pattern matching, sequence matching Lai and Chen, 2017, and analysis of temporal dependence Folmer et al., 2014. In addition, plant connectivity and causality measures have been proposed as the basis for causal analysis of the plant Abele et al., 2013; Hu et al., 2017c. Once a causal structure of the plant is established, consequential alarms can be suppressed to help operators focus on the alarms associated with root causes Larsson et al., 2007. Given that a plant or systems experiences recurring abnormal situations and consequently a number of widely similar alarm floods, data based methods provide a good tool for examination and interpretation by process experts Hu et al., 2017b. However, the lack of prior data would cause a purely data-based solution to fail, while analysis based on causality would still be applicable.

Since plants frequently operate in more than one configuration Quiñones-Grueiro et al., 2019, these causal models need to adapt in the same manner as Beebe et al. (2013) have identified the need to dynamically manage alarm configurations to account for different operational modes. In the context of fault detection and isolation Quiñones-Grueiro et al. (2019) summarize a variety of approaches for tracking the current operational mode, i.e. configuration, of the system: Most of the presented approaches use one model per mode and detect the current mode as the best matching between observations and mode modelling Quiñones-Grueiro et al., 2019. The alternative approach is to track the system transitions and determine the current mode based on the combination of individual variable transitions Srinivasan et al., 2005. At a plant wide level, transition information such as opening and closing of valves is readily available from the supervisory control and data acquisition system (SCADA). Logically, a mode-tracking approach based on transitions at supervisory level can be based on available SCADA events.

In the interest of a comprehensive operator support Kirchhübel et al., 2019c, tracking the correct procedure execution is just as important as identifying the correct operational mode at all times. The majority of changes to operational modes in a plant are related to well documented procedures, such as start-up or maintenance operations Quiñones-Grueiro et al., 2019. Modes can be distinguished more concisely at different abstraction levels in functional process models, where mode transitions happen at the level of operational goals, changed relations between operational goals and supporting functions, or a different combination of components and realised functions Lind et al., 2012; Zhang, 2015. The authors' previous work Kirchhübel et al., 2017a presented a concept of explicitly including the limitations of each mode in the respective functional model. However, the data in the study presented in the present work suggests that additional conditions may inhibit or cause a mode change. This paper presents a framework for tracking operational modes during procedure execution based on SCADA events. We present a formalism for representing operation

procedures. Similar to Abdallah et al. (2018) this automaton representation can direct the adaptation or change of underlying diagnostic models, independent of their implementation. By using data logged from SCADA the representation can be refined and attributed with reference constraints for normal execution, enabling on-line diagnosis of the procedure execution.

In the following section the representation of operational modes, related methods of mode-dependent alarms and sequence alignment, and the proposed methods for identifying and analysing procedure executions are outlined. The industrial case study is then introduced and its results are presented and discussed. Finally, the validity of the proposed method and the advantage for a rapid deployment of advanced support systems are summarised.

B.2 Methods

This section details the relevant methods for representing operational procedures, determining triggering alarms, and identifying the execution sequence from data. Then we propose methods for detecting candidates for the execution of a specific procedure as well as a fast analysis method to validate the execution from available log files.

B.2.1 Representing Operational Procedures

Operations procedures are an essential part of plant operation, as they ensure a consistent execution and high quality and safety of operation EPA, 2007. In emergency situations as well as during transients in operation, like start-up or shutdown, operation procedures define the sequence of actions and expected outcome in the system. These procedures are implemented in the automatic control system or executed by human operators. Operational procedures are thoroughly documented in terms of the specific steps and the timing or thresholds between each step to the extent of integrating computerised procedures with the operator interface. The steps of an operational procedure correspond to state transitions of the plant and can thus be represented as a finite state machine or automaton notation.

In an automaton notation each state corresponds to the current configuration and settings of the plant. Transitions are characterised by a condition and by the set of changes made to the plant. Each transition $T = \{c, \mathbb{A}\}$ describes the necessary condition c to execute the step and the set of actions \mathbb{A} associated with the transition. One specific operation procedure to move the plant from operation point O_i to another operation point O_j is represented by a valid trace in the automaton Lunze, 2004. The trace will normally contain a number of intermediate states s_1, \dots, s_n . Like the operation points these states are characterised by the configuration of valves and set points of the controllers throughout the plant. The condition to initiate the procedure, i.e. transition T_{O_i, s_1} may be tied to a complex condition of multiple

variable thresholds. In the interest of reliability in a stressful situation for the operator all following transitions $T_{s_k, s_{k+1}}$ are commonly conditioned on a time constraint or the behaviour of an individual process variable. Figure B.1 illustrates the representation of an operational procedure as an automaton.

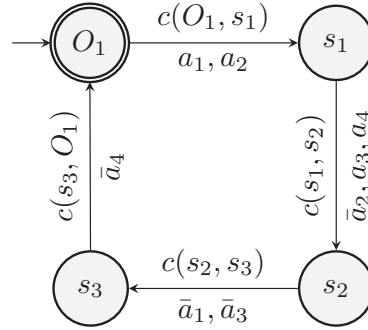


Figure B.1: Example automaton of an operation procedure. Procedures leading back to their starting state are typical for maintenance operations.

B.2.2 Alarm dependent modes

Since the execution of operational procedures changes the plant configuration or mode of operation, individual components will reach the configured alarm limits. These mode dependent alarms could be suppressed given a reliable association of a mode change to subsequent occurrence of the alarm. Inversely, the execution of specific procedures is closely related to specific situations in the plant which could be indicated by an alarm. Hence, alarms could serve as indicators for the transition T_{O_i, s_1} .

The association rule discovery proposed by Hu et al. (2017a) defines an alarm to be dependent on mode change based on two conditions: The alarm occurs frequently enough to be relevant and the probability of the alarm following the mode change is close to 1. Swapping the datasets for alarms and mode indicators allows the application of this method to discover alarms x_i that trigger a procedure execution y_i . Consequently the nomenclature of the conditions is adapted as follows. The number of mode occurrences ξ exceeds the user-defined threshold F_{th} and the approximated probability $\hat{p}(y_i|x_i)$ of the mode change occurring after the alarm is 1 or close to 1.

$$\xi(x_i \rightarrow y_i) \geq F_{th} \quad (B.1)$$

$$\hat{p}(y_i|x_i) = \frac{\xi(x_i \rightarrow y_i)}{\xi(x_i)} \approx 1 \quad (B.2)$$

The time window W_{th} determining whether an occurrence of x_i is followed by y_i should be set sufficiently large to account for the operator's reaction time. Further-

more, for the alarm to be a reliable precursor for the specific procedure execution, the procedure should only be executed if the alarm occurred, introducing the additional constraint that, effectively, the number of occurrences of the alarm and the procedure execution match $\xi(x_i) \approx \xi(y_i)$.

B.2.3 Sequence alignment

Besides the start of a procedure, it is desirable to diagnose the intermediate execution steps of the procedure. To identify the normal execution the recorded data from the alarm and event logs are considered as reference. Considering each procedure execution as a sequence of events in the SCADA system, all sequences for the procedure can be aligned to provide a reference for the events and conditions related to the procedure.

Sequence alignment has been proposed to identify occurrences of similar alarm floods as a method for situation assessment during alarm flood incidents Lai and Chen, 2017. Sequence alignment enables multiple aspects of the analysis: New sequences can be compared in similarity to known sequences as a basis for classification. Additionally, given a group of similar sequences, their common events and respective timings can be used as features for classification of new sequence observations. The method proposed by Lai and Chen (2017) is based on dynamic programming for optimal alignment of multiple sequences. For n sequences of similar length M the complexity is estimated as $\mathcal{O}(n \cdot M^2)$.

A sequence is considered as a set of tuples (e_k, t_i) , where $k \in \{1..K\}$, $i \in \{1..M\}$ with the number of event types K and M event occurrences in the sequence. First the sequence is represented as a $K \times M$ time distance matrix representing the time span between the event assigned to the column and the event type represented by each row. A weighing function is then applied to the time distance matrix:

$$w_{ik} = \exp\left(-\frac{d_{ik}^2}{2\sigma}\right)$$

where σ can be chosen to blur the order of the event by considering close-by matches in terms of timing as well Lai and Chen, 2017.

While the similarity of sequences for an operational procedure should be considered a given, the common sequence alignment can be used to refine the reference for the sequence. Potentially the sequences reveal additional events whose correlation with the sequence is not obvious from plant documentations but can be used for diagnosis. A draw-back of a complete sequence alignment approach like this one is the analysis complexity of finding the total alignment of a large number of procedure executions. Guo et al. (2017) propose a simple matching based on a sliding window around the time projection of one sequence onto the other to determine the likelihood of two sequences to match. However, to establish a reference pattern for a procedure the actual alignment is most relevant.

B.2.4 Execution detection

In order to identify the sequences representing a procedure, time windows with the known plant action can be detected in the log files based on the expected events, such as valve control signals. A transition log is compiled from the event sets associated with the respective transitions, i.e. procedure steps. To distinguish the procedure steps from isolated manipulations a sliding window of size W_{th} needs to contain the complete set. This is illustrated by the first event a_1 in Figure B.2. To additionally account for undocumented changes to the actual implementation of the procedure, this constraint is relaxed so that all instances where the window contains the maximum number of events associated with the transition are considered as occurrences.

Time	Event	
1.0	a_1	
5.0	\bar{a}_1	
5.0	a_3	
5.0	\bar{a}_4	T_{s_3,O_1}
8.0	\bar{a}_3	
10.0	a_1	T_{O_1,s_1}
10.1	a_2	
14.5	a_5	
14.5	a_3	T_{s_1,s_2}
14.5	a_4	
14.6	\bar{a}_2	
20.1	\bar{a}_1	T_{s_2,s_3}
20.1	\bar{a}_3	
24.0	\bar{a}_4	T_{s_3,O_1}

Time	a_1	a_2	\bar{a}_2	a_3	a_4	\bar{a}_1	\bar{a}_3	\bar{a}_4	a_5
1.0	1	0	0	0	0	0	0	0	0
5.0	0	0	0	1	0	1	0	1	0
8.0	0	0	0	0	0	0	1	0	0
10.0	1	0	0	0	0	0	0	0	0
10.1	0	1	0	0	0	0	0	0	0
14.5	0	0	0	1	1	0	0	0	1
14.6	0	0	1	0	0	0	0	0	0
20.1	0	0	0	0	0	1	1	0	0
24.0	0	0	0	0	0	0	0	1	0
	T_{O_1,s_1}		T_{s_1,s_2}			T_{s_2,s_3}		T_{s_3,O_1}	

Figure B.2: Example event log with occurrence candidates

Figure B.3: 1-out-of-N like mapping of the log with detected transition occurrences (gray boxes) and end of procedure indicator (dashed box) highlighted

Given this transition log, identification of the procedure executions is a matter of finding all sequences where the transitions occur in order, or at least the majority of the procedure execution matches. If all transitions of the procedure are associated with time conditions, the documented procedure can directly be interpreted as sequence and all potential executions are detectable by a sequence projection onto the transition log as used by Guo et al. (2017). More generally, the procedure is considered to be delimited by its first and last transition. Consequently, all time windows contained between a detection of the first and last transition are considered execution candidates. If the first or last transition consist of only one action, the sequence

of the first or last two transitions respectively determines the limits of an execution candidate, to maintain the distinction from individual actions. Additionally, missing start or end identifiers can be substituted by an expected execution time to perform analysis of faulty executions. Alternatively, the candidates can be discarded as incomplete executions. Figure B.2 and Figure B.3 show a log file and the respective detection windows for transition candidates. Since T_{s_3, O_1} only contains one event in the considered automaton shown in Figure B.1, the last two transitions mark the end of an execution candidate to avoid confusion with isolated events.

B.2.5 Fast sequence analysis

Once the execution candidates are determined the actual trace in the automaton is considered, yielding a number of groups of actual executions. Assuming an accurate documentation, the trace occurring most resembles the correct procedure execution. Other traces might be related to manual operations or could hint toward common mistakes if they appear in significant numbers.

The complete analysis for representing a procedure as an automaton validated by logged data is outlined in Figure B.4. First the documented procedure is represented as an automaton with the expected sequence of transitions and the action sets corresponding to each transition. Based on the established transition sets, the event and alarm log from SCADA for a reference period is transformed into a transition log as shown in Figure B.3. Procedure execution candidates can then be identified by corresponding pairs of the start and end transitions as described in the previous section. For each execution candidate the trace is identified from the timed sequence of the occurring transitions and classes of identical traces are determined. Subsequently, the automaton structure and event sets per transition can be refined using the relevant classes of traces. Finally the reference statistic for the transition executions can be determined from the occurrences within each class.

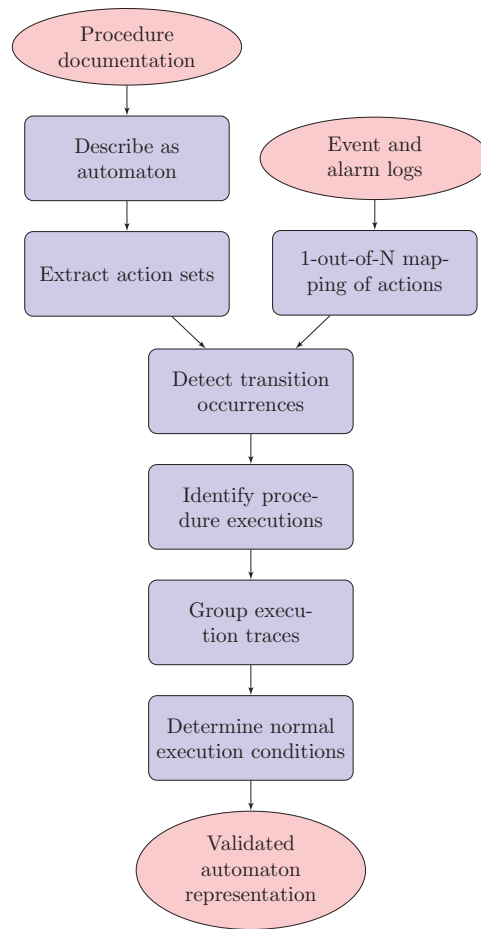


Figure B.4: Block diagram of the sequence identification

B.3 Industrial Case Study

As a case study we investigate the automatic back-washing modes of a sea-water fine filter unit, shown in Figure B.5. The filtration system consists of 4 parallel fine filter units, three of which are always operational, while one may be back-washed. The sequence of valve operations is outlined in Table B.1 and Figure B.6. These operations can be executed in two different procedures, leading either from normal operation to air scouring or directly to back-washing with water.

Per the operations manual air scouring would be used in rare cases, when the filter is severely clogged. Otherwise the filter is back-washed with water when the differential pressure exceeds a preset threshold or at least once a day.

In the case study the proposed sequence identification is applied to detect sequence candidates. The sequences are then analysed in terms of alarm dependency using the

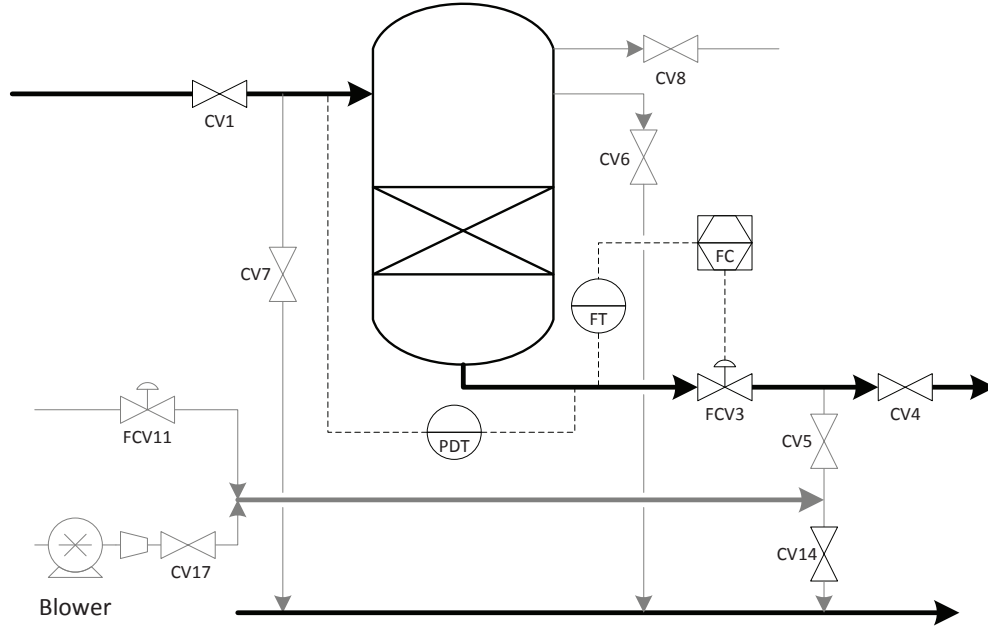


Figure B.5: Flow sheet of the studied filter unit CV17A/B?

presented pattern mining approach and the proposed fast sequence analysis is used to determine the reference automaton for normal execution which is evaluated for its completeness by multiple sequence alignment.

Table B.1: Mode transitions of an individual filter unit

Transition	Action set
<i>s</i>	3:Off
<i>a0</i>	1:Off, 4:Off, 6:On, 8:On
<i>a1</i>	3:On, 6:Off, 14:Off, 5:On, 17A:On, 17B:Off
<i>a2</i>	17A:Off, 17B:On
<i>ab</i>	8:Off, 7:On, 11:On
<i>b</i>	1:Off, 4:Off, 14:Off, 3:On, 5:On, 7:On, 11:On
<i>c0</i>	3:Off, 11:Off
<i>c1</i>	5:Off, 7:Off
<i>c2</i>	1:On, 5:On, 14:On, 3:On
<i>c3</i>	3:Off
<i>c4</i>	5:Off, 4:On, 3:On

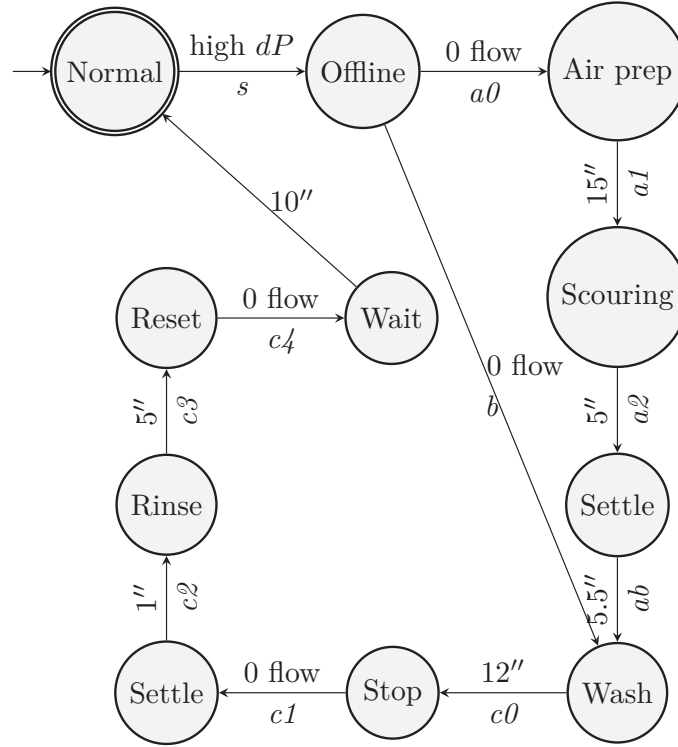


Figure B.6: Automaton of the filter unit. Edges are labelled with the conditions before the transition occurs and the transition name.

B.4 Results

Identifying the documented transitions with window size $T_{th} = 30 \text{ seconds}$ from the A & E log the transition log and sequence candidates were established. Table B.2 summarises the transition log. The two paths in the automaton are marked by the combinations $(s, a0)$ and (s, b) as start of a sequence candidate, while any occurrence of $c4$ terminates a sequence candidate. Since the respective high differential pressure alarm occurred over 70% more often than the procedure was executed the conditional probability can never approach 1. This discrepancy was likely caused by the additional logic restricting back-washing operation to only one out of four parallel units. Therefore, this sequence could not be reliably related with the alarm occurrence.

The timing of the transitions conditioned with 0 flow could be up to 3 minutes according to the operations manual. However, the data analysis revealed these transitions to consistently occur after about 20seconds after the preceding transition. This period is within the chosen detection interval for the action set of the same transition. Thus the combinations $(s, a0)$, (s, b) , and $(c3, c4)$ were considered as complete transitions leading to the reduced automaton shown in Figure B.9. Furthermore, transition

Table B.2: Summary of Transition log, corresponding alarm and resulting sequence candidate

Event	Occurrences	
<i>s</i>	3447	
<i>a0</i>	974	
<i>a1</i>	963	
<i>a2</i>	3841	
<i>ab</i>	973	
<i>b</i>	41	
<i>c0</i>	1027	
<i>c1</i>	10	
<i>c2</i>	1014	*
<i>c3</i>	3447	
<i>c4</i>	1009	
DP alarm	1733	¹
Start	1009	²
* 5:Off never detected		
¹ high differential pressure		
² (s,a0) and (s,b) sequences		

c1 is rarely detected and no complete occurrences of *c2* were found. Upon closer inspection, it was found that in fact the occurrences of *c2*^{*} reliably contain the action set [7:Off, 1:On, 14:On, 3:On], effectively combining (*c1*, *c2*).

The identified transition traces in the execution candidates are visualised in Figure B.8, based on the reduced automaton. Some traces are incomplete since the succession of two start or end markers respectively was corrected by limiting the sequence to a 45min interval. The grouping reveals back-washing with air scouring as the most frequent operating mode and back-washing with water as the second consistent trace. The time constraints shown in Figure B.9 were based on the two most frequent traces, which were considered the ground truth for the plant operation. To verify the identified traces the alignment of all sequences identified as back-washing with air scouring was determined. Figure B.7 shows the pairwise similarity scores and the respective sequence length, in terms of number of events during the execution period. All scores are relatively high at above 0.5 and the larger clusters show some correlation to the sequence length. The common alignment of a random sample of ca. 5% of sequences with the most frequent trace shown in Figure B.10 does not indicate any consistent deviations from the considered action sets. While the action sets are consistent, the order of the specific events varies which is reflected by the slight offset of the sequence start from time 0. The reported set point changes of flow controllers FC3 and FC11 appear as additional information, which could not be

anticipated from the procedure documentation.

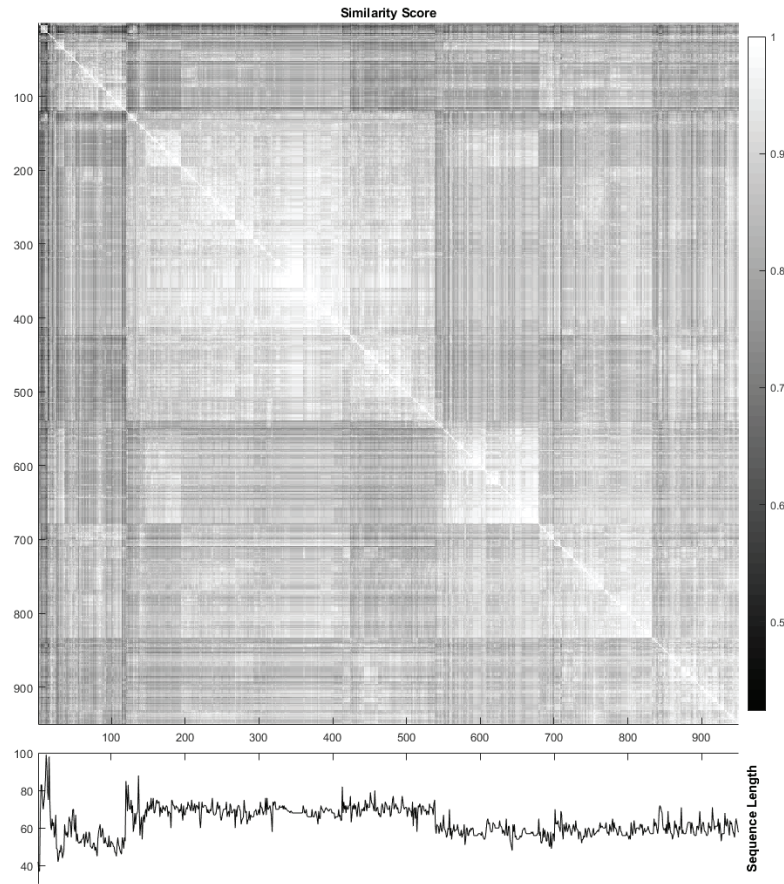


Figure B.7: Pairwise similarity scores and sequence length. Ordered according to highest similarity clusters.

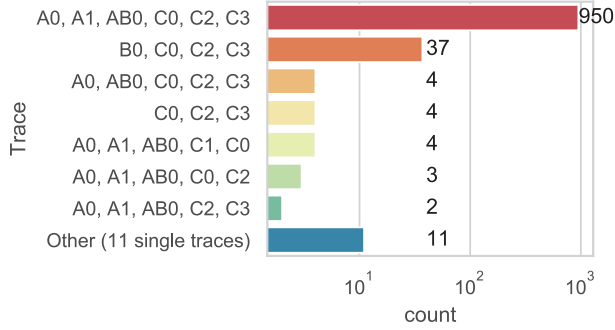


Figure B.8: Occurrences of transition sequences for back-washing during the case study period

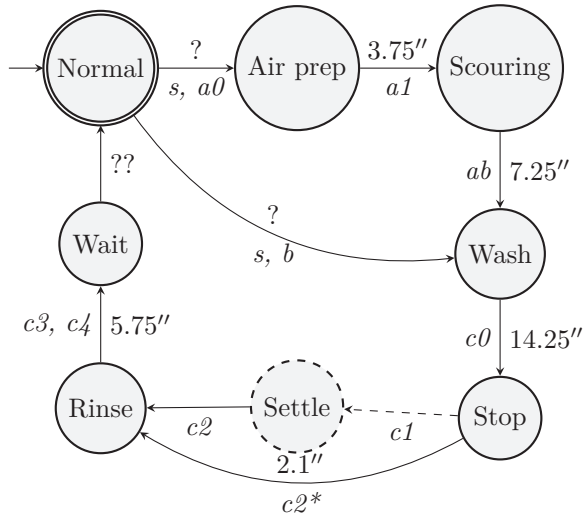


Figure B.9: Reduced automaton derived from the data analysis

Time	Event	
00:13.880	CV4:Off	} <i>s, a0</i>
00:14.880	CV1:Off	
00:16.880	CV6:On	
00:18.980	CV8:On	
00:38.840	CV3:Off	
03:23.960	CV3:On	} <i>a1</i>
03:54.980	CV6:Off	
03:56.039	CV14:Off	
03:58.000	CV5:On	
04:00.999	CV17A:On	
04:03.000	CV17B:Off	
11:08.080	CV8:Off	} <i>ab</i>
11:08.080	CV7:On	
11:11.080	CV17B:On	
11:11.879	CV17A:Off	
11:14.535	FC11:AUTO	
11:14.535	FC11:SP3	
11:49.536	CV11:On	
25:19.535	FC11:SP3	} <i>c0</i>
25:19.535	FC11:SP1	
26:08.160	CV3:Off	
26:10.397	CV11:Off	
28:18.400	CV7:Off	} <i>c2*</i>
28:20.400	CV1:On	
28:24.539	CV14:On	
28:26.540	FC3:SP3	
28:26.540	FC3:AUTO	
28:53.000	CV3:On	
32:28.420	FC3:SP3	} <i>c3, c4</i>
32:28.420	FC3:AUTO	
32:53.020	CV3:Off	
33:18.360	CV5:Off	} <i>c3, c4</i>
33:20.360	CV4:On	
33:11.800	FC3:SP3	
33:23.360	FC3:SP1	
33:23.360	CV3:On	

Figure B.10: Total sequence alignment of a random sample of 50 sequences of back-washing with air scouring. Times in minutes, seconds, and milliseconds ("mm:ss.000") reflect the average relative to the start of the procedure execution.

B.5 Conclusion

The transitions between operational modes as described in the procedures of the operations manual can be used to define diagnostic automata and identify relevant transitions in A&E logs. While the actual execution should follow the operations manual, our case study revealed some significant deviations allowing the consideration of a simplified automaton for the procedure diagnosis. The plant documentation provides a reliable basis for analysis, but due to operator habits or undocumented changes to the plant operation the ground truth represented in daily operation can differ. This highlights the complementary nature of engineering documentation and available data.

The presented method for execution detection and the action set based analysis of procedure executions enables the quick design of diagnostic automata validated by data from the specific plant. Incorporating well documented operational procedures into a diagnostic system allows more comprehensive support, for instance by tracking the proper execution or considering operational procedures in a prognosis for developing faults. The proposed validation process can be a valuable tool to revise operation culture if deviations from the designed procedures affect efficiency. In continuation of the presented work we are investigating the on-line adaptation of the causal presentation for accurate on-line diagnosis.

Acknowledgements

This project is funded by the Danish Hydrocarbon Research and Technology Centre.

We would like to express our gratitude for the collaboration and discussion with Tongwen Chen, Shiqi Lai and Wenkai Hu at the Advanced Alarm Management and Design group of the University of Alberta. And not the least, we thank our co-workers in the Automation and Control group at the Electrical Engineering Department of the Technical University of Denmark for proof reading and constant input to the project.

Toward Comprehensive Decision Support Using Multilevel Flow Modeling

Denis Kirchhübel¹, Morten Lind¹ and Ole Ravn¹

¹Department for Electrical Engineering, Technical University of Denmark

Abstract: The complexity of modern industrial plants poses significant challenges for the design of effective operator interfaces. Although established practices can significantly reduce the frequency of alarms, operators often cannot resolve the failure cascades commonly occurring during emergency situations.

Automating control rooms by incorporating design and operation knowledge about the systems can significantly improve operator efficacy. Intelligent support systems should reduce the amount of information and provide more context to the operators. The operators focus should be shifted from information acquisition to taking informed decisions about mitigation steps.

This contribution gives a brief review of the development of Multilevel Flow Modeling (MFM) and its application to provide operators with decision support and situation awareness, focusing on implementations directly utilising the knowledge represented in MFM. Finally, current efforts toward a comprehensive intelligent human machine interface for operators are outlined.

C.1 Introduction

Operators controlling industrial plants mostly rely on the alarm system to detect off-sets requiring an action. Alarm system should be maintained in a state that does not overload operators during normal operation. However, during emergency situations the connections throughout a processing plant frequently lead to cascades of true alarms overwhelming the operator by presenting alarm floods (Beebe et al., 2013). To deal with alarm flood situations, the relation between the occurring alarms has to be analysed and presented to the operators as concise as possible. An intelligent operator decision support systems guides the plant operators to the region of the plant where the cascade originated from and offer assistance on how to mitigate the situation (Rothenberg, 2009). A timely analysis and suggestions for counter-action can help operators drive the process back to normal operation.

D. Kirchhübel et al. (2019c). “Toward Comprehensive Decision Support Using Multilevel Flow Modeling”. In: *5th IFAC Conference on Intelligent Control and Automation Sciences*. Belfast, UK: IFAC-PapersOnLine.

Level of Automation		Acqui- sition	Analysis	Decision	Execu- tion
Triggered execution	5	+			
Single solution	4				
Selected alternatives	3		+		+
Complete set	2	o		+	
No assistance	1		o	o	o

Figure C.1: Current (o) and envisioned (+) level of automation based on Parasuraman et al. (2000)

In addition to established alarm management practices in industry, mostly data driven alarm analysis methods have been proposed to reduce the strain on operators in abnormal situations. However, incorporating design and operation knowledge into the operator support can help operators with further prognostic information and a more concise understanding of the situation and its consequences (Wang et al., 2016a). The analysis of recorded incidents is an established tool to predict recurring critical situations, for instance Zhu et al., 2016 propose matching the patterns of previous alarm floods. However, these methods depend on reliable data records and the assumption that those critical situations occurred before. A combination of alarm records with connectivity information from plant documentation is shown by Schlegel et al., 2013 to support the alarm analysis where only little data is available. While the plant documentation provides information about the connectivity of components in the plant the nature and direction of causality between deviations is necessary for an accurate analysis of closely linked deviations (Yang et al., 2014). Besides the identification of causal relations between alarms (Larsson et al., 2006), knowledge about the process can be used to automatically generate mitigation procedures for the current situation (Gofuku, 2011).

Traditionally the level of automation at a plant-wide level is characterised by a large cognitive load on the operators who only get alarm and trend information from the human-machine interface without any context. Parasuraman et al. (2000) outline the trade-offs to consider to define the level of automation. Fig. C.1 illustrates the current state of plant operation and the target for a meaningful operator support tool. The goal is to provide a comprehensive solution to reduce the loads on operators and to guide them in critical situations. Therefore, the processing tasks of identifying the situation from a multitude of alarms and continuous signals should be hidden from the operators. Instead, operators will be provided with a short list of the most likely situation analysis and provided with a complete set of tentative consequences to base their decision on. Finally, a set of relevant mitigation procedures will be generated based on the operators diagnosis.
























Functions					function structure <div></div> <i>energy/ mass / control</i>
Mass and Energy Flow			Control		
source 	transport 	storage 	steer 	trip 	
sink 	barrier 	balance 	regulate 	suppress 	
Relations					Goals
Influence	Means-End	Means-Goal	Control		<div>threat </div> <div>objective </div>
influencer 	mediate 	produce 	enable 		
participant 	producer-product 	maintain 	disable 		
		suppress 	actuate 		
		destroy 			

Figure C.2: MFM function primitives adapted from Lind, 2013. Flow function primitives are used in several flow structures. Functions are connected by influence relations inside a flow structure and by means-end relations across decomposition levels representing the contribution to another function or the link to an objective by means-goal functions.

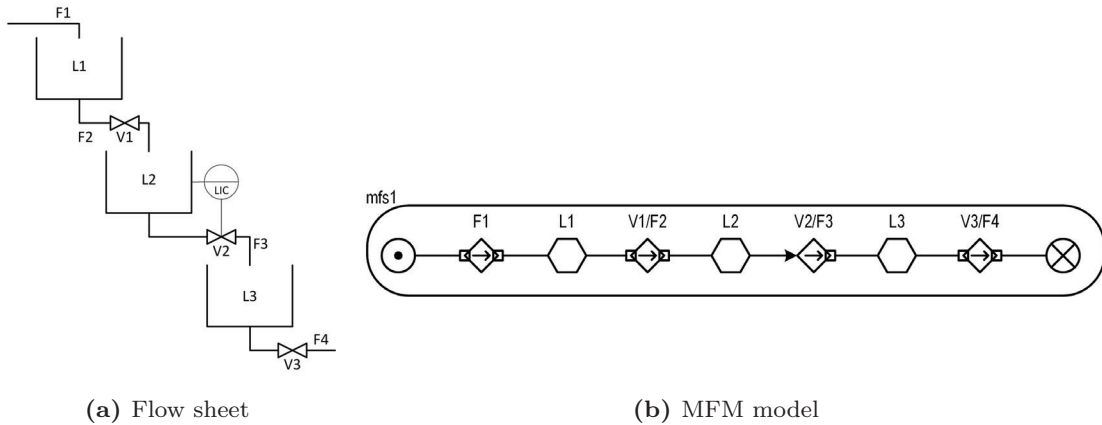


Figure C.3: MFM modeling example of 3 tank system. Being an experimental setup, the process is not assigned any objectives.

Multilevel Flow Modeling (MFM) has been proposed as a modeling methodology for all aspects of operator support. The method was originally developed to represent designers' and operators' understanding of the process and it was gradually extended to provide a comprehensive causal representation of an industrial plant. MFM provides an abstract representation of the connected mass and energy flows in a processing plant as a set of functions. A MFM model explicitly includes the causality between the functions fulfilled by the process units. A MFM model is a hierarchical decomposition of goals to be achieved by certain functions of the system, as well as a part-whole decomposition of each system function into basic material and energy flow functions. MFM provides a graphical modeling language with symbolic representations of these basic flow functions and the relation between functions and objectives of the system. (Lind, 2013) Similar to other graph models, like bond-graphs (Borutzky, 2010) or signed directed graphs (Yang et al., 2014), MFM captures the causal connections throughout the process. However, it also takes a more contextual approach by analysing the plant at the plant-wide level relevant to operator decisions in control rooms rather than the mathematical detail required for other applications.

As an intuitive example a simple 3 tank system is shown in Fig. C.3. The mass flow in itself is only composed of the water source, transports between storages, and a sink. The participant relations toward the transports reflect that the set point of the valves is the only determinant of the flow with the exception of V2, which is determined by the level controller on tank L2. This example illustrates the readily understood syntax underlying all MFM models, where multiple flow structures are usually combined in a hierarchical manner supporting the overarching goals of the plant.

Based on the knowledge in an MFM model, intelligent systems can be developed to assist operators in assessing the state of the plant. The major aspects of intelligent operator support are alarm filtering, root cause analysis and identifying mitigation procedures. Concepts and implementations for each of these aspects are found in the literature. However, no complete system covering the whole range from alarming to mitigation suggestions has been presented to date. The following section outlines a chronology of the research aiming at the application of MFM for online operator support in one of the mentioned aspects. Finally, a conclusion of the past efforts and an overview of our current efforts at the Technical University of Denmark toward a comprehensive operator support tool based on MFM is given.

C.2 Chronology of MFM based approaches to Operator Support

This section focuses on works that directly apply the MFM representation for different aspects of operator support. Approaches such as the diagnosis based on a functional Hazop (Hu et al., 2015), are closely related to the issues of operator support, but do not use the MFM model in an online fashion.

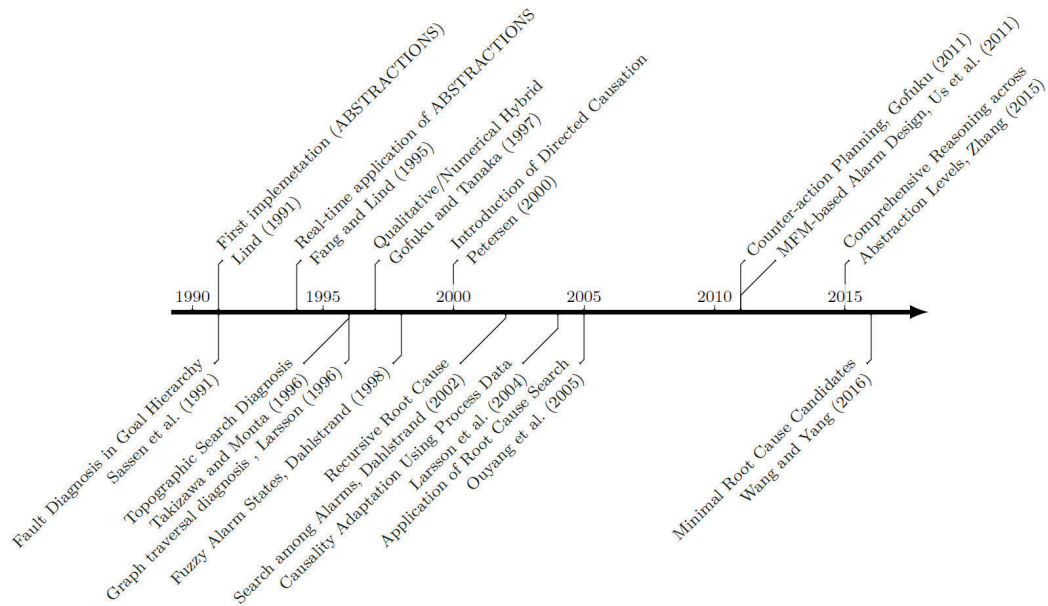


Figure C.4: Chronology of MFM implementations for operator support and situation awareness.

The fundamentals for using MFM in an automatic support system were established by Lind, 1991 with the first implementation of generic reasoning in an object oriented structure of MFM concepts. The ABSTRATIONS framework made it possible to dynamically reason about the propagation of faults through a MFM model based on a generic rule base that could be applied to any given model and fault situation (Lind, 1991). Fang and Lind, 1995 present a real time application of the ABSTRATIONS framework through an interface to the programmable logic controller (PLC) of a pilot process that provides a causal diagnosis by propagating faults along the relations inside the MFM model.

In contrast, Sassen et al., 1991 proposed an efficient hierarchical search inference of possible root causes. The inference uses a reduction of the MFM model to a hierarchy of goals and sub-goals essentially reflecting a fault tree. The fulfilment of each of these goals can be evaluated against the actual state of the plant and causes can be traced deeper in the hierarchy until the root cause is identified. In the same manner local faults, which do not affect the plant as a whole, can be analysed by searching the respective sub-tree. Similar to this goal decomposition and the hierarchical search through the goals of the system, Takizawa and Monta, 1996 introduce a hierarchical search in MFM models. An efficient diagnosis within the MFM model is realised by first tracing the fault to a specific flow structure in the hierarchy. The inferred fault propagation within the flow structures can be evaluated against the actual system state. Inconsistencies between the inference and measured deviations are used to identify the location of anomalies. They further presented heuristics to estimate

measurements for components without instrumentation to establish more detailed diagnoses.

The application of an MFM based expert system for alarm based root cause analysis and sensor validation was demonstrated by Larsson, 1996. The system is applied to group alarms according to the causality represented in MFM. The alarms are determined to be primary alarms close to the root cause of the disturbance or consequential alarms which are caused by a disturbance represented by another alarm. The evaluation of the state is proposed as interactive questions to the operator. However, these interactions slow down the system and impede the real-time applicability. Hence, the system is suggested to be used in an on demand manner to understand occurring situations. Taking into account that alarms are not necessarily configured correctly, Dahlstrand, 1998 proposed a fuzzy assignment of the fault states before performing the alarm analysis described by Larsson, 1996. This analysis was reported to yield more robust results that can cope with common issues like chattering alarms.

While the MFM modeling of goals and functions had been well established, Petersen, 2000 identified a need to refine the representation of causality between flow functions. The distinction between direct and indirect influence and a comprehensive set of propagation rules for patterns in the MFM syntax are defined by Petersen, 2000. Larsson et al., 2004 advocated for dynamic adjustment of causality in MFM models rooted in the consideration that the process dynamics are adjusted for different operation modes. The proposed method determines a pairwise correlation measure of local features in the process data. A low correlation measure indicates that the causal connection of the respective functions should be inhibited. Thus, the same model can be applied to the diagnosis of a process in different stages, given that the differences between operation modes only affect the causality and not the structural link of functions to components. (Larsson et al., 2004)

Dahlstrand, 2002 expanded on the causal alarm analysis to identify minimal sets of root causes that fit the observed alarms. The analysis is done by reduction of causal dependency graph covering all function and state combinations in a given MFM model. The resulting causal paths can cover observed as well as unobserved alarms making the method robust against chattering alarms. The method produces a number of explanations that can help narrow the operator's focus to the correct process regions. Ouyang et al., 2005 demonstrated the application of MFM for the diagnosis of design accidents in a nuclear reactor.

Gofuku and Tanaka, 1997 propose to augment the functional model with operational knowledge to include alternative behaviours of specific parts of the system. They realise this extension by generating a quantitative simulation model using Hybrid Phenomena Theory based on the abstraction in MFM to facilitate prognostic operator support. Furthermore, they propose an operator support interface utilising the design intention incorporated in MFM models to explain abnormal situations and augmented by mitigating actions. These possible counter-actions could be identified from the operational knowledge and verified by the quantitative simulation model. Expanding on their previous work, Gofuku, 2011 demonstrated the use of additional knowledge in combination with the causal reasoning in MFM to generate linguistic

explanations of an analysis in the model. They also reiterate a simplification method for the model previously outlined by Fang, 1994. The simplification contracts functions that are not directly linked to components and thus reduces the paths included in the explanation for the operator.

Incorporating similar information to operational knowledge proposed in (Gofuku and Tanaka, 1997), Us et al., 2011 suggest an alarm design method based on MFM. External conditions and disturbances for individual functions of the system are used to identify points of mitigation and early warnings for arising alarms, creating a dependency structure of possible faults. The proposed alarm system considers only alarms associated with the modelled function of the plant and incorporates the consequence reasoning to predict alarms that will soon be triggered due to the propagation through the plant. (Us et al., 2011)

Zhang, 2015 has presented the most recent set of propagation rules for MFM models and applied it to the diagnosis of a nuclear power plant. The work also explores the adaptation of the model or its links to the process to accommodate different modes of operation as previously pointed out by Larsson et al., 2004. In contrast to Larsson's approach, the mode adaptation of process-function and means-ends relations is proposed rather than causalities inside the repetitive flows.

Finally, Wang and Yang, 2016 outline an implementation of an MFM based expert system similar to Dahlstrands reduction of a causal dependency graph. However, they additionally include a link between modelled faults and common operator mistakes to represent the identified set of root causes in a more natural language than the underlying MFM model.

C.3 Ongoing research

As outlined in Section C.1 the operator tasks can be split up into the four parts: data acquisition, situation analysis, decision and counter-action execution. Some work has been published concerning the data acquisition and linking it to the causal analysis, e.g. (Dahlstrand, 1998) and (Larsson et al., 2004), but in general most of the work related to MFM considers the input to be valid alarms. Instead, the majority of applications of MFM focus on the second step of situation analysis. Most notably the groups of Lind and Larsson have proposed methods of cause analysis and more recently Wang and Yang, 2016 have outlined an online system using MFM to identify root causes. The recent work of the group of Gofuku has been focused on using MFM as the basis for generating operation procedures. Either in unknown situations or to automate the generation of procedures the methods outlined by Gofuku, 2011 can guide the execution of mitigation procedures once a diagnosis is established. While all of the research outlined above contributes to the different aspects of control room automation, each aspect has been researched mostly in isolation. Fig. C.5 outlines the envisioned process for implementing a comprehensive operator support system.

To get meaningful results from the proposed knowledge based system the initial knowledge needs to be accurate. Nielsen et al., 2018a are proposing a framework for

model validation by comparing the inference generated from an MFM model with the propagation documented by experts in e.g. a Hazard and Operability Study (HAZOP) or acquired from numerical simulation or process data. As outlined by Lind, 2017, the creation of a model library will facilitate the modeling process. A library for different processes in the oil and gas sector is currently being developed at the Technical University of Denmark (DTU). By providing validated models for common subsystems in engineering documents of a specific application domain the overall model consistency can be improved.

In the control room the support system has to diagnose the situation and provide suggestions within a time frame of minutes or below to enable the operator to react before the system trips. In (Kirchhübel et al., 2017b), the authors outline a new propagation method that reduces the computational effort for the graph based inference of multiple concurrent offsets. The accuracy of the model can be further increased by the extension of the inference rules to include diverse implementations of control loops under investigation by Zhang and Lind, 2017. To overcome the uncertainties introduced by heterogeneous alarm configuration, the detection of faults by data analysis methods and machine learning are considered as interface between the process and the operator support system.

While a set of actual root causes can help focus the diagnosis, the estimation of tentative consequences and the ensuing risk is just as relevant to prioritise further steps and take appropriate actions. The operator can be provided with a range of plausible explanations for the situation based on the inference. The authors suggested a preliminary ranking method of identified root causes to determine the most relevant causes for the operator to consider (Kirchhübel et al., 2017b). In continuation of the considerations in (Zhang, 2015) the adaptation of the model used for the inference to the current situation is further being investigated in terms of knowledge representation (Kirchhübel et al., 2017a) and the identification of the current situation. Future research will further concern the loop closure from actually observed situations and operator reactions to the underlying model.

As the final stage of the operator support system the knowledge represented in an MFM model can be used for automatic planning of procedures to mitigate a detected deviation. Based on the concepts proposed in (Gofuku, 2011), Song and Gofuku, 2017 outlined a planning method using the MFM based causal inference. This branch of investigation is also pursued by the group at DTU.

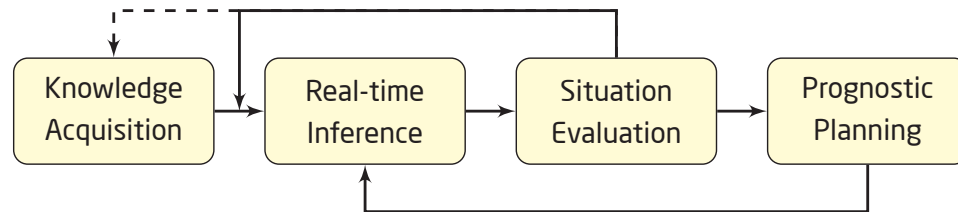


Figure C.5: Development and application process for a knowledge based advanced operator support system.

C.4 Conclusion

The presented chronology shows that a number of implementations and applications have been reported continuously since the first implementation of MFM. However, the complementary elements of alarm management and root cause analysis and reaction suggestions have been widely separated in the research. The current research efforts at the Technical University of Denmark and collaboration partners aim to combine the whole range from initial offset detection to alarming and finally counter-action generation. Within the context of operator support the integration of diverse methods with knowledge representation in MFM are under investigation. The current research projects and partners as well as recent publications can be found on the research group's website <http://mfm.elektro.dtu.dk>.

Generation of Signed Directed Graphs Using Functional Models

Christopher Reinartz¹, Denis Kirchhübel¹, Ole Ravn¹ and Morten Lind¹

¹Department for Electrical Engineering, Technical University of Denmark

Abstract: Intelligent fault diagnosis systems can be a major aid to human operators charged with the high-level control of industrial plants. Such systems aim for high diagnostic accuracy while retaining the ability to produce results that can be interpreted by human experts on site. Signed directed graphs have been shown to be a viable method for plant-wide diagnosis that can incorporate both quantitative information about the process condition as well as qualitative information about the system topology and the functions of its components. Their range of application in industrial settings has been limited due to difficulties regarding the interpretation of results and consistent graph generation. This contribution addresses these issues by proposing an automated generation of signed directed graphs of industrial processes in the chemical, petroleum and nuclear industries using Multilevel Flow Modeling; a functional modeling method designed for operator support. The approach is demonstrated through a case study conducted on the Tennessee Eastman Process, showing that Multilevel Flow Modeling can facilitate a consistent modeling process for signed directed graphs. Finally, the resulting benefits regarding qualitative reasoning for plant-wide diagnosis are discussed.

D.1 Introduction

The operation of industrial plants in the chemical and petroleum industries poses a significant challenge to human operators on site. Especially the operation in abnormal plant states, which is characterized by e.g. the failure of one or multiple components, can be problematic. The production loss due to the operation in such abnormal plant states, which accounts for up to 18% of the total production loss (Crowl and Louvar, 2011), could be severely reduced if correct recovery strategies were executed by the operators in time. One of the main reasons for delayed or incorrect reactions by the operators during plant recoveries is that the information provided to them is not tailored to support quick, informed decision making. The alarm systems currently

C. C. Reinartz et al. (2019). "Generation of Signed Directed Graphs Using Functional Models". In: *5th IFAC Conference on Intelligent Control and Automation Sciences*. Belfast, UK: IFAC-PapersOnLine.

featured in industrial plants provide a descriptive overview of the plant state, featuring alarm signals, which indicate a deviation of the process from its nominal condition. Information about the potential relation of these alarms is not displayed (Rothenberg, 2009). It is imperative to provide the operators on site with a decision support system, which can generate comprehensive and contextual information about the current plant state to improve the overall performance of operations on industrial plants. This information should include probable root causes of existing disturbances and potential actions countering their effects. Due to the high degree of connectivity in most industrial plants, a local fault may propagate through large parts of the system. Most observed disturbances and alarms are in fact the result of such a propagation and do not necessarily give any indication about the actual fault. A system for fault-diagnosis in large-scale processes needs to capture the causal connections of the system to be able to reason about the origins of faults. Additionally, the diagnosis must be comprehensible for the human operators on site, if the software is intended for decision support (Yang et al., 2012).

Model-based (Venkatasubramanian et al., 2003a; Venkatasubramanian et al., 2003b) and process history based methods (Venkatasubramanian et al., 2003c) for fault-diagnosis have been presented in the past. This paper will focus on Multilevel Flow Modeling (MFM) and Signed Directed Graphs (SDG). Both methods belong to the field of qualitative, model-based analysis and both possess the capability to capture causal connections within complex industrial processes. Research into improving the results obtained from SDG-based methods using quantitative approaches has been conducted with promising results (Maurya et al., 2007; Peng et al., 2014; Wan et al., 2013; Yang et al., 2012). A drawback of using SDGs is that both the model generation and the interpretation of obtained results is not straightforward. Multilevel Flow Modeling is capable of producing results that can be interpreted by process experts, but the combination of the method with quantitative modeling approaches, though possible (Hu et al., 2015; Kim and Seong, 2018; Larsson et al., 2004), has not been researched extensively yet. A mapping between SDG and MFM models offers the potential to make results obtained by one method accessible to the other and thereby extend the application range of both methods. The basic concepts of both methods are outlined in Sections D.2 and D.3, followed by an explanation of the method for the automated signed directed graph generation in Section D.4. The method is tested on the Tennessee Eastman process in Section D.5 and the results are discussed in Section D.6.

D.2 Representing causal relations using Signed Directed Graphs

Signed directed graphs provide a generally applicable means of representing qualitative causal models (Venkatasubramanian et al., 2003a). A mathematical expression for SDG models $G = (V, E, \phi, \psi)$ is defined by Bondy and Murty (1976). The nodes

(V) of a SDG represent system variables and the arcs (E) connecting the nodes represent the effect of these variables on each other. Each node has an assigned qualitative state $\psi : V \rightarrow \{+, 0, -\}$, which indicates whether the state of the variable represented by the node is higher, equal or lower than nominal. Each directed arc has either a positive or a negative sign $\phi : E \rightarrow \{+, -\}$, which is determined by the direction of effect between two variables. The direction of the arc is determined by the cause-effect relation of the connected nodes. Each arc points from the 'cause' node to the 'effect' node. Figure D.1 displays the two basic connection types that can be expressed using signed directed graphs. States are propagated in SDGs by traversing



Figure D.1: Basic positive (a) and negative (b) connection types between two nodes of a signed directed graph.

the edges connecting the nodes. A state $\psi(v_i) = '+'$ will propagate to all nodes directly connected to v_i . Nodes connected via positive arcs will assume a high state (+) and nodes connected via negative arcs will assume a low state (-). The benefit of SDGs is that causal dependencies are explicitly expressed through the signed edges. Because of that, the state propagation is straight forward and easily traceable. SDG can furthermore be applied very broadly, since the modeling language is not derived from processes of any specific industry. A drawback of using SDG is that the direct representation of causalities does not enable a direct inference about the reason for the causality. This limits the usefulness of obtained results for further utilization by human experts.

D.3 Multilevel Flow Modeling

Multilevel Flow Modeling is a functional modeling method designed to model industrial processes such as nuclear and chemical plants. The benefit of MFM is that both the models and the reasoning results can be interpreted by process experts in the field, since the modeling language is based on concepts that they are already familiar with, such as mass and energy flows. MFM models are meant to capture the functionality and the causal relations between system parts rather than the topology on a component level. In Multilevel Flow Models, processes are divided into mass, energy, and control flow structures. These contain more detailed process representations, which are modelled using the basic function- and relation types displayed in Fig. D.2. Each basic function type is assigned a qualitative state, which corresponds to commonly used terminology for alarm states used in industrial applications. The functions Storage, Transport, Source and Sink can assume the states high, normal and low. High and low states signify that the function variable is outside of its nominal range. The

barrier function can assume a normal or a breach-state. The balance function has only a normal state and serves as a flow-distribution function. Lind (2011) provides an overview of the basic model components and modeling principles. MFM is different from other qualitative modeling techniques like SDGs because causal relations are expressed implicitly. The causal relation between two MFM-functions

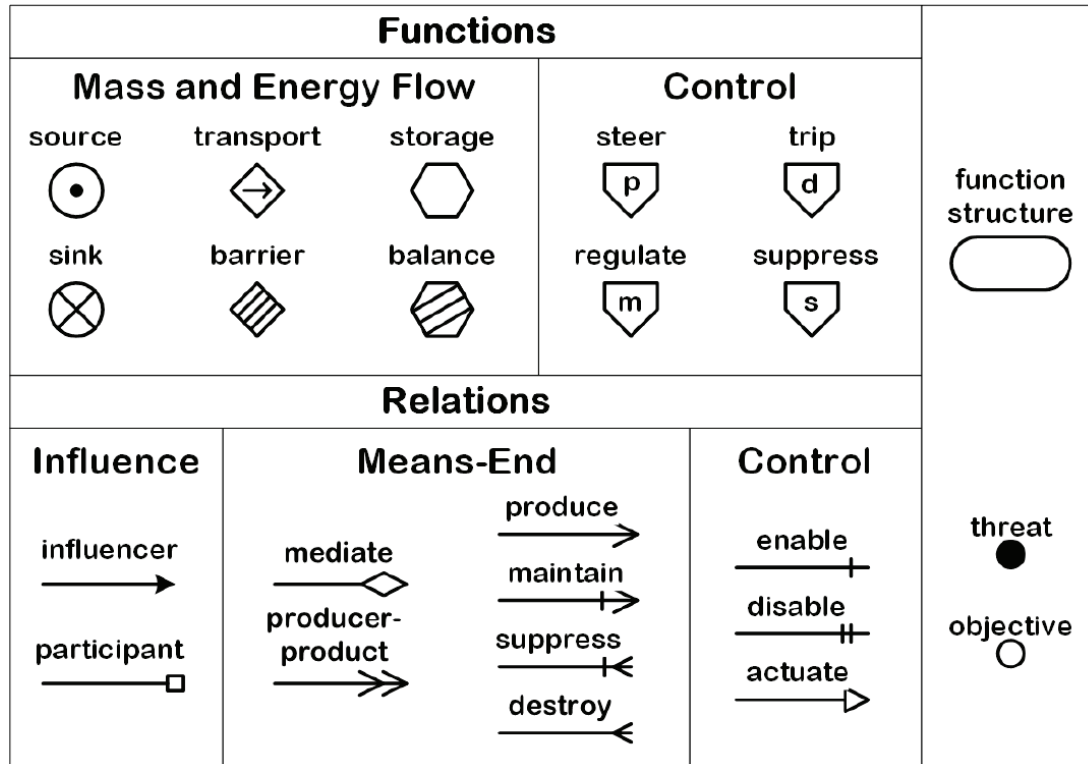


Figure D.2: Basic MFM Functions and Relations (Zhang, 2015).

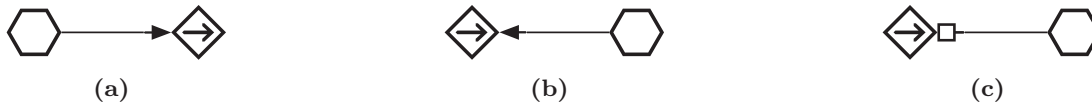


Figure D.3: Influencer (a,b) and participant (c) relations between "storage" and "transport" functions.

is determined by the respective function types (Storage, Sink, Source, Balance, Transport), the connection types (influencer, participant) and the position of the functions towards each other, considering the direction of flow in the system (upstream, downstream). The direction of flow in the system is indicated by the arrow in the transport function symbol. Transport-type functions (transport, barrier) affect adjacent

functions by default. Non-transport functions (storage, sink, source, balance) affect adjacent functions, if they are connected via an influencer relation. Figure D.3 shows three MFM models, which represent different explicit causal relations. The cases of a "high" state of the transport function and a "high"-state of the storage function are considered as examples. Table D.1 summarizes the causal inference for the models in Fig. D.3. In the case shown in Fig. D.3a, a "high" state of the transport signifies an outflow of the storage, which is higher than expected. The result is a decrease and eventual "low" state of the level of the storage. The explicit cause-effect relation from the transport to the storage therefore has a negative sign. In the cases provided in

Table D.1: Inferences for MFM-models in Fig. D.3

Scenario	Initial state	Inference
Case "a"	transport: high	storage: low
	storage: high	transport: high
Case "b"	transport: high	storage: high
	storage: high	transport: low
Case "c"	transport: high	storage: high
	storage: high	transport: not affected

Figures D.3b and D.3c, a "high" state of the transport function will lead to an eventual "high" state in the storage, since the storage will fill faster than expected due to the higher than nominal input. The explicit cause-effect relation from transport to storage has a positive sign in these cases. The case D.3c is different from D.3b, because it features a participant relation, which signifies that the state of the storage does not affect the transport function directly. Reasoning about causality in MFM requires the implementation of a fixed set of propagation-rules for each combination of MFM-functions (Zhang et al., 2013).

D.4 Conversion of MFM models to Signed Directed Graphs

The aim of the conversion is a signed directed graph that captures the explicit causal relations, which are implicitly expressed in the MFM model. Such a graph can be generated in three steps, if an MFM model is available. First, n SDG nodes are created, where n equals the amount of functions in the MFM model and each node corresponds to a specific function. The explicit causalities between the MFM-functions are then extracted by using the rules described by Zhang et al. (2013) in the second step. The signed edges of the SDG can be generated based on the information about explicit

causalities that was extracted in step two, as shown in Fig. D.4 for the most basic MFM models. Fig. D.4a shows participant relations between transport and storage

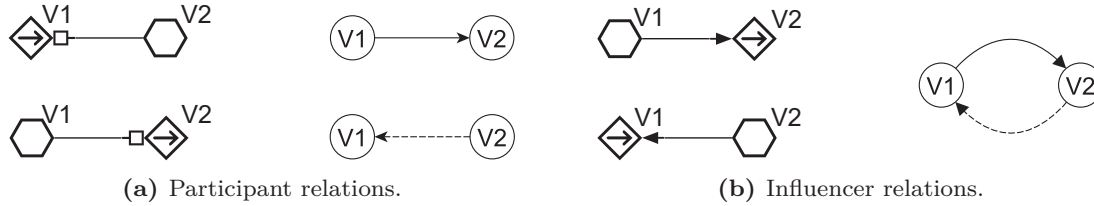


Figure D.4: Representation of Signed Directed Graph equivalents of basic MFM relations.

functions. It can be observed that the nodes representing the storage function in the SDG have no outgoing edges, since the participant relation implicitly states that the state of the storage does not have a direct effect on the state of the transport function. The lack of outgoing edges on the nodes representing the storages states the same relation explicitly. The same model with influencer instead of participant relations is shown in Fig. D.4b. In this case both MFM models result in an identical signed directed graph representation, which is a direct result of the MFM-reasoning that is illustrated in Table D.1, considering that the node "V1" corresponds to the storage function in the upper model and to the transport function in the lower model in Fig. D.4b. It provides a good example for why Multilevel Flow Models are better suited for human interpretation than signed directed graphs. The MFM models in Fig. D.4b represent two different physical processes, e.g. a tank being drained by an outflow (top) and a tank being filled by an inflow (bottom), which is not apparent in the signed directed graph representation. Because the signed directed graph does not capture and thus does not contain such implicit information about the process, it is not possible to generate a MFM model from a SDG directly. Reasoning results obtained from SDGs can, however, be transferred to MFM models if a mapping between the nodes of both models is established, which is always the case if the SDG is generated from an MFM model.

D.4.1 Reduction of Signed Directed Graphs

As mentioned above, a primary purpose of MFM is the representation of the process in a human-readable format. This necessitates that elements of the process, which are not monitored but still vital for the human operator's understanding, have to be considered in MFM models to ensure a comprehensible output. Signed directed graphs do not necessarily need to keep to this restriction. In most cases only process variables which are monitored or can be actuated are considered in SDG representations, since they can provide direct feedback to the reasoning system. Signed directed graphs generated using MFM will initially feature all variables that are captured in

the MFM model, including those that are not monitored. The MFM representation does, however, include information about which MFM-functions are directly connected to process measurements using a "process variable" tag, thus defining a set of monitored nodes I . This information can be used to reduce generated signed directed graphs to exclusively include nodes representing monitored variables. The applied reduction scheme consisting of two main steps is described in Fig. D.5 and illustrated by a simple example in Fig. D.6. Nodes, whose indegree δ^- or outdegree δ^+ is equal

Step 1:

```

while  $v_i \in V \setminus I$ ;  $\delta^-(v_i) = 0 \vee \delta^+(v_i) = 0$  do
   $V \leftarrow V \setminus v_i$ 
end while

```

Step 2:

```

while  $v_i \in V \setminus I$  do
  for  $e^+ \in E$ ;  $e^+ = (v_m v_i)$  do
    for  $e^- \in E$ ;  $e^- = (v_i v_n)$  do
      if  $v_n \neq v_m$  then
         $e^* = (v_m v_n)$ 
         $E \leftarrow E \cup e^*$ 
      end if
    end for
  end for
   $V \leftarrow V \setminus v_i$ 
end while

```

Figure D.5: Reduction algorithm used to obtain a graph featuring only monitored nodes. V , I , and E are defined as the sets of nodes, monitored nodes and edges, respectively.

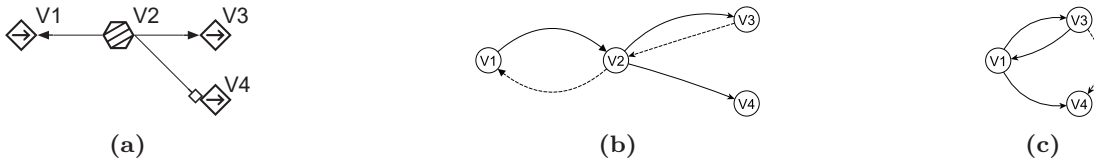


Figure D.6: Multilevel Flow Model (a), equivalent Signed Directed Graph representation (b) and SDG representation excluding "V2" (c) of a hypothetical process with branching causal relations.

to zero are removed in the first step, unless they represent a process variable. The remaining nodes that are not connected to process variables are recursively removed in the second step. A similar reduction scheme was previously described by Kramer and Palowitch (1987). The reduced graph represents the same causal relations as the

original in regard to the measured process variables, but representations of internal connections between unmeasured nodes may be lost. This is acceptable if the target application uses the process data as its primary source of information.

D.5 Case Study

The Tennessee Eastman process is a well-known process from the chemical industries that was first introduced by Downs and Vogel (1993). It has since been used as a benchmark problem for studies in fault-diagnosis (Yin et al., 2012), state estimation (Ricker and Lee, 1995), control (Zerkaoui et al., 2010) and others. The five main components of the process are the reactor, condenser, vapour-liquid separator, stripper and the compressor. The open loop process presented by Downs and Vogel (1993) is unstable and needs to be shut down due to high reactor pressure, if left unchecked. Ma and Li (2017) adopt a control scheme presented by D'Angelo et al. (2016) to test the performance of a fuzzy signed directed graph model of the reactor of the TE-process for fault diagnosis. The graph created by Ma and Li (2017) is used as a reference model for this case study, since it has been successfully applied for fault-diagnosis for the Tennessee Eastman reactor and serves as a good benchmark. The flowsheet and control scheme are displayed in Fig. D.8.

The aim of this case study is to compare a SDG that was generated using a MFM model of the Tennessee Eastman reactor to a reference SDG presented by Ma and Li (2017) and reach conclusions about the merit of the method proposed in this article based on the consistency between the MFM model and the generated SDG on the one hand and the similarity between the generated and reference SDGs on the other hand.

The MFM model used as a basis for the graph generation is displayed in Fig. D.9. It was designed from the knowledge represented in the flow sheet using the modeling strategy presented by Lind (2017). The graph shown in Fig. D.7 is the direct result of the application of the principles presented in Sections D.4 and D.4.1. The consistency of MFM and generated SDG is tested by comparing the results of single fault propagation applied to both graphs. To this end, all scenarios resulting from a single fault input are generated using the propagation rules for MFM and SDG, respectively. It is then verified that nodes referring to the same process variable have the same state in all corresponding scenarios. This test was successfully run for all single fault scenarios (positive and negative deviation of each process variable) of the generated Tennessee Eastman Reactor model.

It is apparent that the generated (black edges) and reference (black and red edges) graphs are very similar. The automatically generated graph does not contain additional edges compared to the reference. The reference graph contains four edges and thus for causal relations which are not covered by the generated graph. It was expected that these relations would not be represented in the generated model, since they take the effect of the stripper and vapour-liquid separator into account, which

have not been considered in the MFM model that focuses on the functionality of the reactor. It is noticeable that the generated graph does not feature any edges which imply "false" causal relations, which is important, since invalid causalities can lead to incorrect reasoning and thus impede the fault analysis.

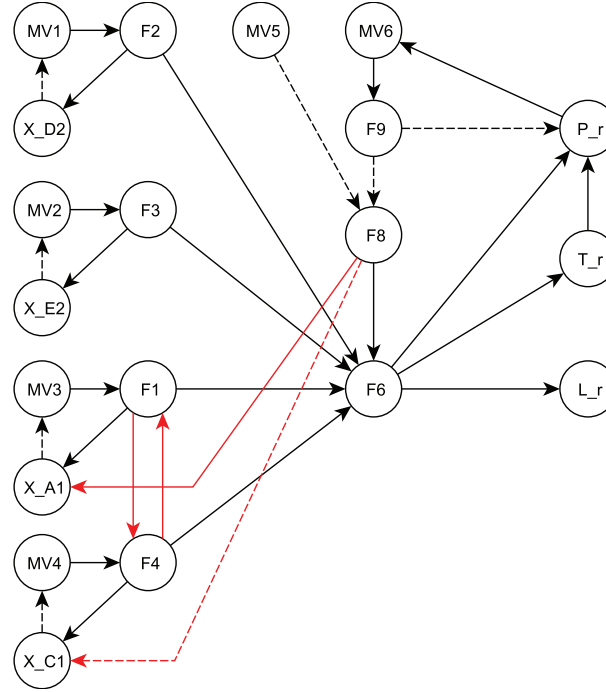


Figure D.7: SDG describing the causalities between measurable process variables that affect the TE-Reactor. Red edges: Causalities described by Ma and Li (2017) that do not appear in the MFM-generated SDG.

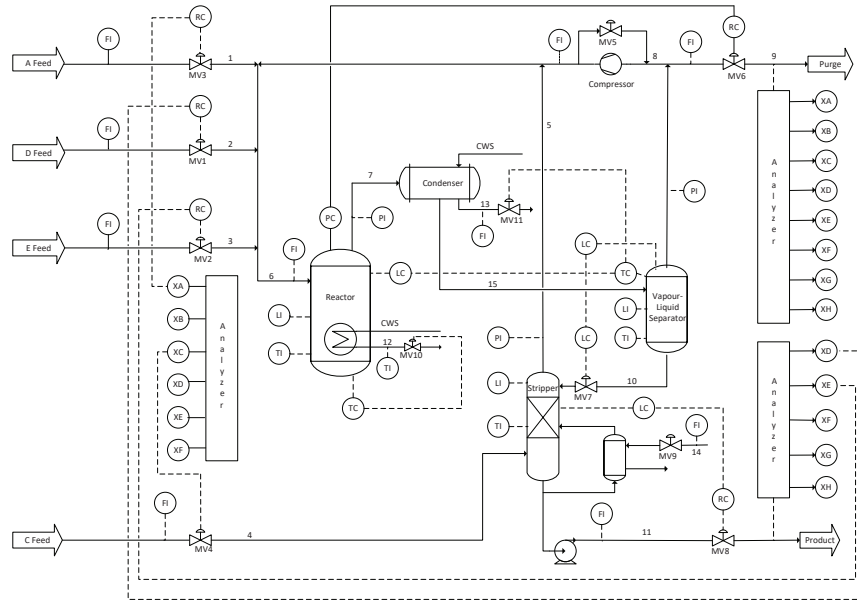


Figure D.8: Flowsheet of the Tennessee Eastman process containing the control structure used by Ma and Li (2017).

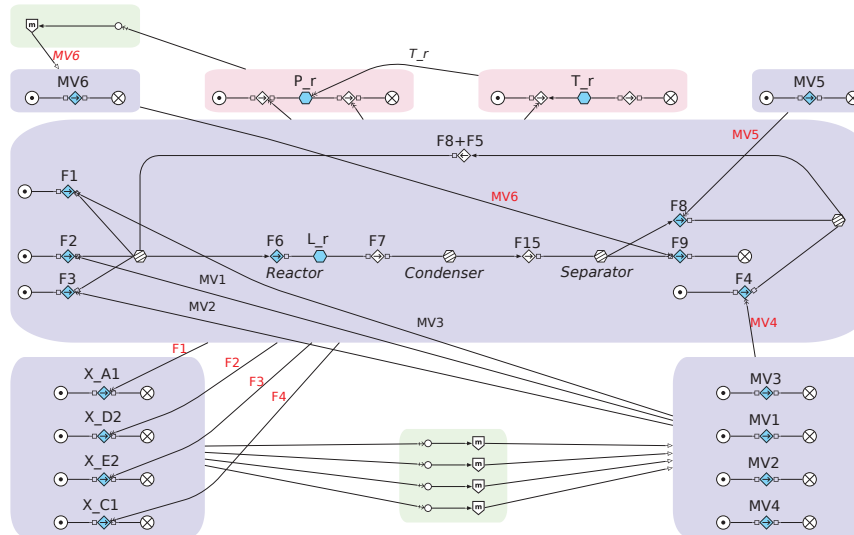


Figure D.9: Multilevel Flow Model of the Tennessee Eastman Reactor. Displayed are functional elements of the process which have either direct or indirect influence on the reaction. Measurable process variables are highlighted blue. Red print indicates the variables belonging to means-end relations. Process variable names are directly adopted from the used reference study conducted by Ma and Li (2017). The process is modelled using the MFM-modeling principles described by Lind (2017).

D.6 Conclusion

The conversion of functional process models to signed directed graphs is generally achieved by translating implicit causalities to explicit ones, as presented in Section D.4. Such an approach is realizable for many functional modeling concepts, e.g. some of those presented by Erden et al. (2008), indicating that the presented method is in principle applicable for other approaches than Multilevel Flow Modeling as well.

Researchers focusing on signed directed graph based fault analysis profit from using functional models in two ways. They can use the existing modeling guidelines developed for the functional modeling concepts that are designed to identify causal structures in specific industrial applications. Following such guidelines ensures consistent model design and thereby consistent design of the signed directed graphs used for the analysis. In addition to that, the opportunity to express the diagnosis output in a functional modeling framework designed for the evaluation by process experts facilitates communication in the respective target industry.

A major benefit for researchers in the field of functional modeling is that the extensively researched field of fault diagnosis using signed directed graphs becomes directly accessible. This provides the opportunity to integrate such research into the development of functional reasoning systems with minimal effort.

The method presented in Section D.4 facilitates the addressed consistent model generation for processes in the chemical, petroleum and nuclear industries. Initial testing on the Tennessee Eastman process has yielded promising results, including a automatically generated, well-formulated signed directed graph model of the Tennessee Eastman reactor which closely matches models presented in literature. Further case studies on systems from the chemical and petroleum industries as well as further research on the integration of signed directed graphs into the Multilevel Flow Modeling concept are planned for the near future.

Identifying causality from alarm observations

Denis Kirchhübel¹, Xinxin Zhang¹, Morten Lind¹ and Ole Ravn¹

¹Department for Electrical Engineering, Technical University of Denmark

Abstract: The complexity of modern industrial plants poses significant challenges for the design of effective alarm systems. Rigorous alarm management is recommended to ensure that the operators get useful information from the alarm system, rather than being overloaded with irrelevant state information. Alarm management practices have been shown to significantly reduce the frequency of alarms in industrial process plants. These practices help focusing the operators' attention on actually critical situations. However, they cannot resolve the cascades of critical situations frequently occurring during emergency situations.

Multilevel flow modelling (MFM) has been proposed as a way of representing knowledge about the industrial process and infer causes and consequences of deviations throughout the system. The method enables the identification of causes and consequences of alarm situations based on an abstracted model of the mass and energy flows in the system. The application of MFM for root cause analysis based alarm grouping has been demonstrated and can be extended to reason about the direction of causality considering the entirety of the alarms present in the system for more comprehensive decision support.

This contribution presents the foundation for combining the cause and consequence propagation of multiple observations from the system based on an MFM model. The proposed logical reasoning matches actually observed alarms to the propagation analysis in MFM to distinguish plausible causes and consequences. This extended analysis results in causal paths from likely root causes to tentative consequences, providing the operator with a comprehensive tool to not only identify but also rank the criticality of a large number of concurrent alarms in the system.

E.1 Introduction

Modern industrial plants contain a large number of interacting control loops and concurrent processes affecting the productivity and safety of the system.

While control practices for individual components and constrained processes are widely adapted in industry, plant-wide control often faces too many uncertainties from the environment and the interconnected processes to be economically feasible Rangaiah and Kariwala, 2012. Human operators who rely on alarm systems to supervise the plant operation thus control the vast majority of plants in the energy, petrochemical and chemical industries. Due to the large risks for humans as well as the environment in case of failures, rigorous alarm management is recommended for these industries to avoid overloading the operators EEMUA, 2013.

Alarm management practices have been shown to significantly reduce the amount of irrelevant alarms presented to the operator by thoroughly scrutinizing the necessity and importance of the most frequent alarms and where possible combining and removing redundant alarms Rothenberg, 2009. A well maintained alarm system can avoid operator overload during normal operation. However, emergencies frequently generate cascades of true critical situations throughout the plant that overwhelm the operator with so called alarm floods. To cope with such situations the relation of those alarms needs to be examined and compiled into concise information to aid the operator in identifying the most relevant and immediate threats. Beebe et al., 2013

To identify relevant information during alarm floods the causality relation of the occurring alarms is a key information. While the analysis of historian data on the alarms gives insight in common correlation between alarm occurrences, inference of causality requires incorporating process knowledge. Wang et al., 2016a

Multilevel Flow Modelling (MFM) provides an abstract representation of an industrial process as a decomposition of connected mass and energy flows Lind, 2013. MFM methodology has been proposed as a versatile process representation to analyze causal patterns in a plant Us et al., 2011. Inoue et al. Inoue and Gofuku, 2016 propose to use MFM for counter action planning in unknown emergency situations. Larsson and DeBor Larsson et al., 2007 and more recently Wang et al. Wang et al., 2016b have demonstrated the application of MFM for root cause identification and alarm reduction based on identified root causes. The combination of dynamic alarm reduction and a system to propose feasible counter-actions would enable operators to react efficiently to any situation in the plant.

As a starting point toward this comprehensive operator support system the extension of the method for root cause identification is described here. The identification of root causes as well as propagation paths based on the causality between observed alarms is discussed in this contribution. The following sections introduce the MFM methodology and the propagation reasoning based on MFM models. Based on that the proposed method for combination is outlined and conclusions for future work are drawn.

E.2 Multilevel Flow Modelling

Multilevel Flow Modelling (MFM) represents the goals and functions of a system by decomposing the mass and energy flows as means and ends of operating the system.

Each flow component along the means-end dimension is described by basic flow functions. By the combination of means-end decomposition of the overall operation and part-whole perspective of individual flows the function of the system is analyzed and can be represented as a graphical model using the MFM concepts shown in Figure E.1. As example the MFM model of a watermill is considered, adapted from Lind Lind, 2011 and shown in Figure E.2.

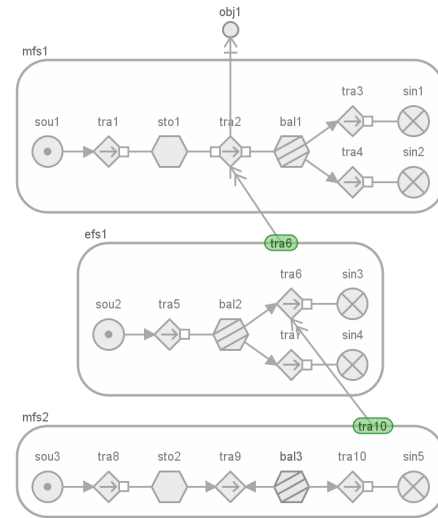
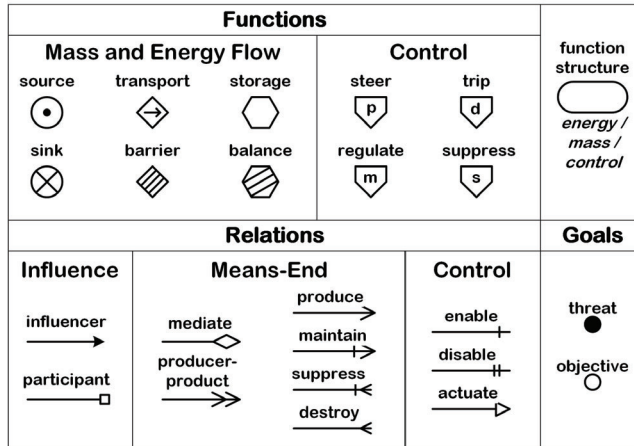


Figure E.2: MFM model of a watermill as described by Lind, 2011

Figure E.1: MFM concepts used for modelling.

The model shows the main objective of grinding grain as *obj1*, which is achieved by the mass flow of grain in *mfs1*. The grains fed into the mill are converted to flour and split-up bran. Energy flow *efs1* reflects the conversion of the energy from the water by the gears and mill stone to energy used for grinding and energy losses not used in the system. This energy in turn is supported by the mass flow of water into the flume across the water wheel represented by *mfs2*. In this way the interacting functions throughout the system are described for the nominal operation.

Industrial plants, however, often have a multitude of different operational situations by design. Each of these operational modes is defined by different nominal functions in the system and thus requires an adaptation of the model Kirchhübel et al., 2017a. As described by Inoue et al. Inoue and Gofuku, 2016 adapting the model also facilitates the investigation of alternative behaviors of the plant.

E.3 Prognostic and Diagnostic Reasoning

Based on the MFM modelling primitives the propagation of failures through the system can be analyzed. Combinations of propagated states and patterns in the model describe the failure propagation in the system. Zhang Zhang, 2015 describes the most recent version of these propagation rules.

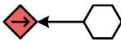
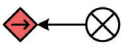
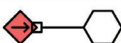
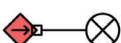
high-high		high-high
high		high
low		low
low-low		low-low
high-high		high-high
high		high
low		low
low-low		low-low

Figure E.3: Downstream consequence propagation of faults on a transport function Zhang, 2015

The rules are defined for both, plausible causes and consequences of an observed state. The example in Figure E.3 shows how a failure associated with a transport function has consequences on connected functions downstream of that transport. Applying all propagation rules to an observed failure, a fault tree of failures in the model can be generated. The resulting tree generally reflects alternative propagation paths at the same level. The alternative paths are not necessarily but frequently mutually exclusive.

In conjunction with a set of truth-maintenance rules the possible propagation paths of each observation present in the system can be dynamically generated. This way changes to the observations as well as the considered configuration of the plant are taken into account at any given moment. The resulting causal paths are limited to plausible scenarios connected to specific observations, whereas a generic fault model as used by Wang et al. Wang et al., 2016b comprises a comprehensive causal representation of all possible states. While the computational burden of this dynamic approach is higher than a precompiled causal graph, it yields more flexibility to accommodate changes of the system behavior.

The propagation analysis for causes of two different faults is illustrated in Figure E.4. Each subordinate level in the tree structure reflects plausible causes for the immediate parent.

The fault *tra2:low* equals to a low processing throughput of the mill, meaning that no grain is being milled. The other fault *tra9:high* corresponds to a too high flow of water from the flume over the waterwheel. The comparison of the two consequence trees reveals, that neither of the two observed alarms can be the cause for the other. In fact, if the low throughput were caused by a fault of the water flow it would be the opposite - low flow instead of a high flow as observed.

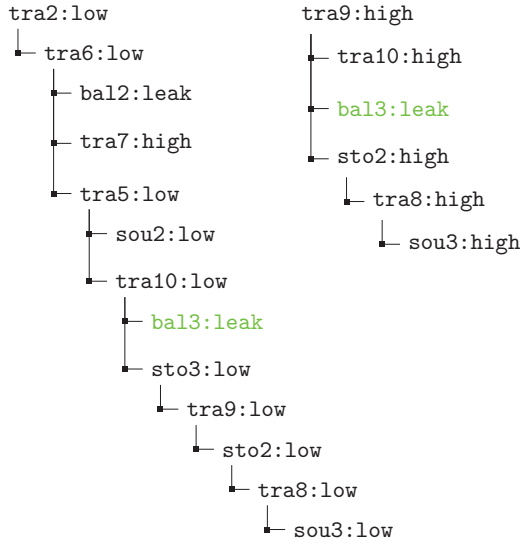


Figure E.4: Cause analysis of two faults in the water mill, common cause *bal3:leak* is highlighted

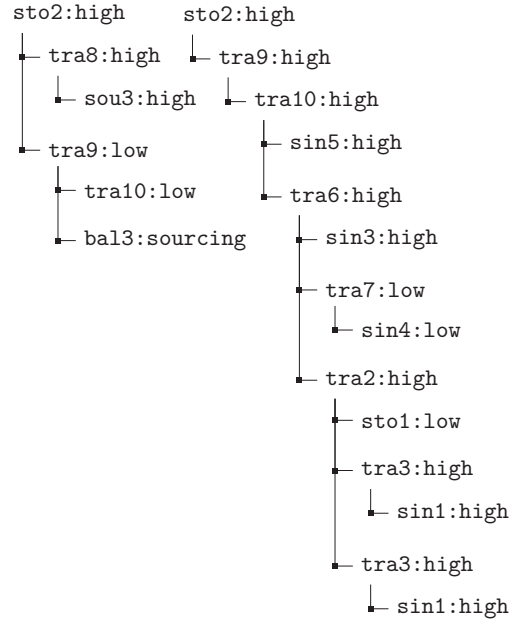


Figure E.5: Cause and consequence analysis for observation of high flume level

In addition, a later observation of the flume level being high – *sto2:high* – is considered (Figure E.5). This observation may well be a direct cause for the high flow of water from the flume. If it were the only fault in the system, however, it could not explain why the production of the water mill is low in the considered situation. Hence, a combined analysis of the possible causes and consequences is necessary.

E.4 Combining Multiple Alarms

By comparing the cause tree representation for the first two considered failures in Figure E.4 the common cause *bal3:leak* can be manually inferred. For a more complex system and a larger number of simultaneous observations, however, the proper inference becomes significantly more complex. This raises the need for a general and structured solution for reliable identification of the best explanation.

Considering the combination of all suggested causes and consequences as a directed causal graph grants a better overview of the whole situation. Furthermore, a directed graph can be systematically analyzed by applying graph theory.

The example of *tra2:low* and *tra9:high* results in the graph shown in Figure E.6. All edges are directed from cause to consequence. The green nodes represent the

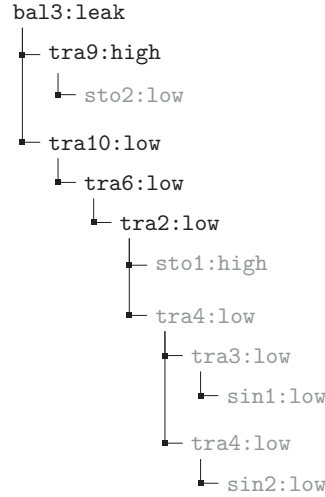


Figure E.7: Consequence tree derived from the first observed scenario

The combined graph can be efficiently updated with new connected observations. Considering *sto2:high*, the complete cause and consequence analysis is already present in the graph and no new inference is necessary. As there exists no causal tree that also includes the new observation, the high flume level has to be an independent contribution to the high flow from the flume.

The diagnosed causes would hence be a high inflow to the flume (*sou3:high* or *tra8:high*) as well as the spill of water represented by *bal3:leak*. The consequences will no longer include a low level in the flume (*sto2:low*) as the flume level has been identified as a likely cause for the situation. This shows, that the combined analysis of fault propagation from the observations yields a clear distinction of the causality between connected functions in the system.

E.5 Conclusion

This contribution outlined a generic method for situation analysis and distinction of causality based on MFM reasoning and graph interpretation.

In the context of alarm management for a complex plant the underlying framework as well as the models have to be adaptive for many different configurations in the plant.

The method proposed here takes in dynamic reasoning results based on an MFM model and has the potential to reliably distinguish the direction of causality as well as identifying the most plausible root causes and tentative consequences of any given scenario.

This method is currently being implemented in a real-time environment of a pilot-scale oil and gas production plant. Further investigation will be dedicated to the

efficiency of the method and the integration of selective advanced signal processing for prognostic analysis of scenarios and fast distinction of situations.

Acknowledgement

The authors would like to acknowledge the support of the Danish Hydrocarbon Research and Technology Center (DHRTC) at the Technical University of Denmark as well as the fruitful cooperation with Eldor Technology, Norway.

Generating Diagnostic Bayesian Networks from Qualitative Causal Models

Denis Kirchhübel¹ and Thomas Martini Jørgensen²

¹Department of Electrical Engineering, Technical University of Denmark

²Department of Applied Mathematics and Computer Science, Technical University of Denmark

Abstract:

Safety and efficiency of modern industrial plants can be improved by providing operators with effective digital assistants to diagnose abnormal situations occurring in the plant. To make sense of a large number of alarms, root cause analysis can help pinpoint the origin of an abnormal situation. We investigate the translation of qualitative causal models into Bayesian belief networks (BBN) to utilize efficient tools for probability inference. The diagnosis result of a fault scenario of the Tennessee-Eastman-Process highlight the feasibility of the principle approach and the ongoing research aims to fully leverage the potential of BBN.

F.1 Introduction

The impact of situational awareness in the decision-making process of operators of industrial processes has long been a concern in safety critical operations. In order to address the challenge of human factors the performance of the operator interface naturally is essential. Situational awareness is necessary to maintain reliability, anticipate events and respond appropriately when or before they occur. Comprehension of the situation is based on a synthesis of perceived information. This comprehension requires an understanding of the significance of the presented elements beyond simply being aware of information. By developing digital assistants to support the operator, the operators can make better-informed decisions.

Reducing the number of alarms presented to an operator, by means of alarm management has been the subject of many improvement efforts in industry. This reflects in the guidelines and standards defined for process industries, most significantly the Engineering Equipment Materials Users' Association (EEMUA) publication 191

D. Kirchhübel and T. M. Jørgensen (2019). "Generating Diagnostic Bayesian Networks from Qualitative Causal Models". In: *24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*. IEEE, pp. 1239–1242.

(EEMUA, 2013). Hollifield and Habibi (2006) have compiled an overview of the current best-practice in industry. It is necessary to combine all available information to provide accurate assistance: recorded data, as well as design and operation knowledge (Wang et al., 2016a).

Digital diagnostic assistants for “human in the loop” real-time operations can be valuable in identifying root causes of failures (Natarajan and Srinivasan, 2014; Nguyen et al., 2016) or even predict the onset of disturbances that could lead to failures (Zhang, 2015; Zhu et al., 2016), thereby reducing downtime and limiting stress for the operators. To establish a model of the system, technical documentation like P&IDs and process flow diagrams, are important (Wang et al., 2016a), but it is also vital to “harvest” expertise and experience from engineers and operators (Cai et al., 2016) or empirical data (Nguyen et al., 2016). Based on this data the causality between offsets in the plant can be modeled and the fault propagation can be analyzed to identify root causes (Zhang, 2015), (Arroyo Esquivel, 2017).

Bayesian Belief Networks (BBN) have been proposed as means of diagnosing faults based causal process representations. BBN yields a way of representing uncertainty about the causal relationships and with efficient Bayesian inference one can update likelihood estimates of the unobserved states and the potential root causes, implying the possibility of ranking them. One approach to establish the process representation on which Bayesian Network analysis can be added is to apply Hazard and Operability Studies (Hazop) (Hu et al., 2015) or fault-tree analysis (Milford, 2006), (Bobbio et al., 1999), which are usually prepared for risk assessment. Peng et al. (Peng et al., 2014) also describe the application of Bayesian inference to distinguish root causes identified by fault propagation in a multi-logic causal model. However, producing and maintaining accurate representations of a process in safety documents in a consistent digital format is a lengthy process involving process, safety and operations experts.

Multilevel Flow Modeling (MFM) facilitates the generation of causal models. Representing the process as a hierarchy of mass, energy, and control flows by functional concepts provides a structured approach to modelling causality. (Lind, 2013) Using the abstract causal model reduces the knowledge engineering effort and fault propagation is used to propose root cause candidates. To identify the actual root cause BBN can then be used, similar to the approach in (Peng et al., 2014).

This paper examines more closely the application of BBN for on-line fault diagnosis based on causal models. The approach is demonstrated on a case of the Tennessee Eastman Process simulator (TEP) (Downs and Vogel, 1993). The paper is organized as follows. First we present a causal model of the thermal aspects in the TEP based on the concept of Multilevel Flow Models (MFM) (Lind, 2013). Next, we outline different approaches to generate a BBN from the causal model. Finally, we obtain the probabilities of all possible root causes and compare the results with the true fault for the investigated TE scenario.

F.2 Causal Model of the Tennessee Eastman Process

The MFM causal model shown in Figure F.1 represents the thermal aspects of the TEP. The MFM methodology decomposes the system into mass and energy flows as a hierarchy of means and supported objectives of the plant. By using abstract function primitives to represent the different flows the model can be related to the design intentions and human understanding of the plant operation (Zhang, 2015). Some of the abstract mass or energy flow functions in the model directly relate to measurable quantities in the process, such as pressures, temperatures, and flow rates, as well as manipulated variables. The implicit causal model in MFM yields a causal di-graph for the measurable quantities of the TEP shown in Figure F.2.

For the diagnosis, we consider the scenario of a high condenser coolant temperature, indicated by two alarms: F9 high and F5 low. The alarms are generated by a 2% band around the steady state value (Ma and Li, 2017) using the simulation data provided by Rikker (Ricker, 2019).

F.3 Bayesian Belief Nets

A Bayesian belief network is a probabilistic graphical model that describes variables and their conditional dependencies based on a directed acyclic graph. Efficient algorithms exist for both inference of probabilities and learning of the causal structure. As such, it represents an established methodology for analyzing complex causal dependencies between faults (Jensen, 1996; Pearl, 1988). There are a number of synonyms in the literature all corresponding to a Bayesian Belief Net (BBN). These include Bayes nets, directed acyclic graphs, and probabilistic networks.

A BBN models the joint probability distribution of the combined states of the system under consideration - in our case given by the nodes in the (combined) fault-tree(s). The BBN is defined by a directed acyclic graph in which each edge corresponds to a conditional dependency and each node corresponds to a unique random variable. In addition to the graph structure, the BBN contains Conditional Probability Tables (CPT) for all nodes having one or more parents and by marginal probability tables for the root nodes (nodes without parents). Generally, one can say that a BBN is a solution to model complex systems because they perform the factorization of the variables joint distribution based on the conditional dependencies. The main objective of BBNs is to compute the distribution probabilities of a set of unknown variables given the observation of one or more other variables. The detailed principles of this modeling tool are explained in (Pearl, 1988), (Jensen, 1996).

Building a BBN involves both a structural part (the graph) and a quantitative part (the probability tables). Both of these parts can be learned from data. However, developing a BBN for a complex system entirely by learning from historical diagnostic cases, although very attractive, is rarely an option due to lack of data. On the other hand updating an existing model from data is often feasible. It is also important to note that the structural part (the backbone of the causal dependencies) is more

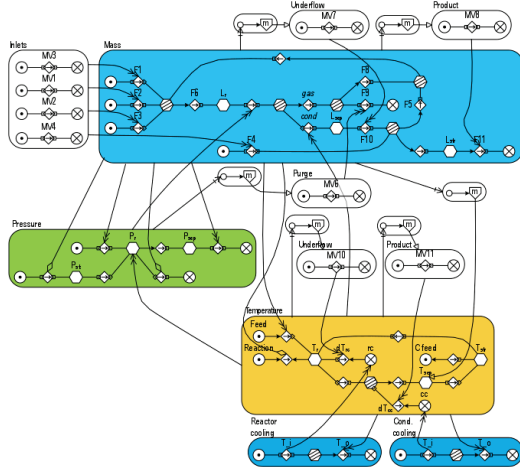


Figure F.1: Multilevel Flow Model of the TEP. Decomposition of the process by flow function primitives (Lind, 2013) for mass flow (blue), pressure (green), heat transfer (yellow), and control loops (white).

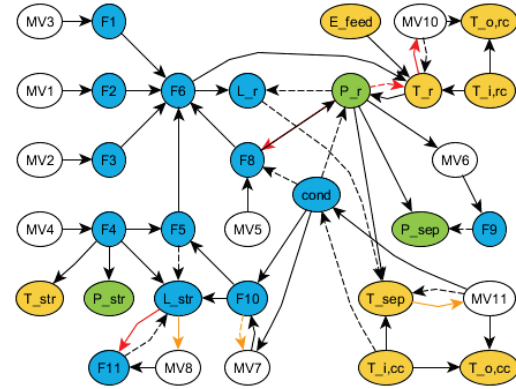


Figure F.2: Signed directed graph generated from MFM model. Positive (solid) and negative (dashed) edges are directed from cause to effect. Minor reciprocal influence (red) and influence of controlled variables (yellow) removed to generate BBN. Nodes are coloured according to MFM perspectives.

difficult to learn than the parameter values of the probability tables. Accordingly, it is advantageous to draw structural information from an available causal representation of the system. There have been a number of publications showing how to map a fault-tree to a BBN (Milford, 2006), (Bobbio et al., 1999). It is in principle straightforward and the OR gates can be directly represented by deterministic CPTs. In addition we can then easily represent uncertainty in the anticipated propagation of causes to consequences by modifying the CPTs using so-called Noisy OR gates (Antonucci, 2011).

F.4 Generating Bayesian Nets

In the following, we outline three different paths of generating a BBN from the presented causal model. Firstly, a recipe for removing cycles in the causal di-graph is presented, generating a BBN with trinary states for all variables in the graph. Subsequently, two different ways of interpreting the fault-trees generated by back tracing into a BBN.

A. Trinary Representation

If it is possible to translate a causal graph directly into a BBN, each variable can be considered as having one of three states: no deviation (normal), too low value (low),

or too high value (high). However, control loops in the system will reflect as cycles in a causal graph, whereas a BBN needs to be acyclic by definition. We propose a recipe to resolve these cycles and create a BBN from a causal di-graph, or signed di-graph. The approach is based on the following assumptions:

- During normal operation, reciprocal influence between two variables has a dominant direction, due to the process and control design.
- Actuated variables are likely root causes, e.g. a fault in the sensor, controller implementation, or the actuator can cause an offset.

Consequently, the following steps are performed to generate a BBN:

1. Identify and keep only dominant influence between reciprocal variables. See Figure F.2.
2. Remove edges from controlled variable to corresponding actuated variable, if they are part of a cycle. See Figure F.2.
3. Generate a CPT for each intermediate node using Table F.1. Each entering edge e_{ji} of node v_i has sign s_{ji} . If v_i has multiple entering edges the cross product of the states is formed as the sum of the high or low observations, the normal state is only probable if all parent nodes indicate normal.

Table F.1: CPT for intermediate node v_i based on parent v_j

$v_i \backslash v_j$	$s_{ji} = I$		
	high	normal	low
high	1	0	0
normal	0	1	0
low	0	0	1

$v_i \backslash v_j$	$s_{ji} = -I$		
	high	normal	low
high	0	0	1
normal	0	1	0
low	1	0	0

Table F.2: CPT for binary root cause based on trinary parent node

trinary	trinary		
	high	normal	low
high			
fault occurred	1	1	0
normal	0	1	1

trinary	trinary		
	high	normal	low
low			
fault occurred	0	1	1
normal	1	1	0

B. Mapping Causal Paths

The assumptions presented in A will often be too simplistic to cover all scenarios in a plant, e.g. if a cycle led to remove the influence of P_r on MV6 by rule A.2 there would be no propagation causing F9 high. Alternatively, we explore a situation specific approach to generating the BBN. Given specific fault observations, the causal model

can be used to trace back the causal path and identify possible root causes. In doing so, only a subset of the graph will be traversed and occurring cycles are detected. The back-tracing spans a fault-tree but common consequences linking branches of the fault-tree are being ignored. To penalize longer propagations uncertainty is introduced for each traversed edge in the causal model, assuming a higher likelihood of local causes rather than long propagation paths in the system. Independent back tracing of multiple faults will lead to individual fault-trees. To reach a meaningful diagnosis nodes representing the same variable cannot be considered independently per fault-tree or causal path, as they refer to the same physical entity. Two ways of combining the generated fault-trees are considered here: (B1) combining all recurring nodes or (B2) maintaining split trees unless the causal paths overlap. The former approach requires re-examining nodes previously detected to form cycles as the combination of independent fault-trees can recreate those cycles. The latter approach avoids these cycles by only combining identical causal paths. In this way, we ensure to have single nodes for root causes but we may have multiple copies of intermediate nodes. The nodes of the mapped BBN each represent a specific deviation with two states – “fault occurred” or “ok”. To incorporate the uncertainty introduced by the propagation Noisy-OR gates define the conditional probability of “fault occurred” if a given parent node also has a “fault occurred” state but with the respective uncertainty depending on the length of the causal path between parent and intermediate node.

C. Connecting Root Causes

While the complete BBN representation of the causal model with trinary nodes correctly interprets states that are mutually exclusive, a BBN generated by combining fault-trees can contain nodes referring to the same variable in mutually exclusive states without representing their relation. As outlined by Lampis (Lampis, 2010) a common n-ary parent node incorporating all fault states and “normal” can link mutually exclusive root nodes.

Extending method B with the notion introduced in A, a trinary node with the states “high”, “normal”, and “low” is added as parent to the root nodes in the BBN created from independent fault-trees. Since the intermediate nodes are not coherently linked in the same manner, the uncertainty according to the propagation length is maintained for the intermediate nodes. Table F.2 represents the CPT for a binary root cause depending on the trinary parent node.

F.5 Diagnosis Results

Table F.3 shows the probabilities inferred by the respective BBNs with the CPTs described before and no prior knowledge about the marginal distributions (“stupid prior”). Hugin Researcher (Kjærulff and Madsen, 2013) was used to design and

diagnose the BBNs. In case of the trinary representations the marginal probability for “high” and “low” is 33.33%. In the binary cases either of the independent root nodes for “high” and “low” can show “fault occurred” with 50%. Accordingly, the deviation from 33.33% or 50%, respectively, after Bayesian inference, indicates the likelihood of the fault being the root cause.

Table F.3: Root cause probabilities (high / low in %) for TEP scenario high condenser coolant temperature $T_{i,cc}$

Root cause	(A) Trinary	(B1) Combined	(B2) Split	(C1) Combined & Trinary
$T_{i,cc}$	56.71 / 14.08	53.37 / 50.04	54.80 / 50.07	35.55 / 31.12
MV11	14.08 / 56.71	---	---	---
MV4	17.96 / 46.11	50.04 / 56.18	50.16 / 64.09	29.24 / 37.43
MV5	37.88 / 26.82	50.05 / 50.13	51.24 / 50.33	33.30 / 33.37
E_{feed}	36.63 / 29.81	50.07 50.04	50.78 / 50.22	33.34 / 33.32
$T_{i,rc}$	29.81 / 36.63	50.04 / 50.06	50.22 / 50.78	33.33 / 33.34
MV1/ MV2/ MV3	34.62 / 31.87	50.07 / 50.04	50.70 / 50.11	33.35 / 33.32

From Table F.3 we observe that all BBNs put the actual root cause as first or second highest probability. However, the significant shortcoming of the fault-tree based BBNs (B and C) is reflected in the consistently higher probability of MV4_low as root cause, since it could immediately cause F5 low. However, the missing link between the mutually exclusive states prevents the interpretation of this contribution. On the other hand, the presented trinary BBN does not contain the same notion of uncertainty for longer propagations represented by the Noisy-OR gates, since the noisy combination of more than binary states is not trivial, and at the same time some causalities had to be removed from the model to create a valid trinary BBN.

The diagnosis achieved by the two approaches of binary BBN yield the same

ranking of causes. The introduction of trinary root causes for the binary model does not affect the ranking of the root causes, but allows a clearer distinction between the fault states of a single variable. It is noted that the required post-processing of the split paths can be significantly smaller, if the overlap of consistent causal paths is already considered during the back tracing in the causal model.

F.6 Conclusion

In summary, the presented investigation reveals a great potential in using BBN to interpret the root cause analysis based on causal models and shortlist the most relevant root causes to support operators. However, generating a general diagnostic BBN from a causal model is limited by the acyclic nature of BBNs. On the other hand, back tracing multiple observations and combining their fault-trees into a BBN disregards causal connections that could improve the diagnostic power of the BBN.

The combination of the comprehensive trinary and simple binary representation of quantitative fault states improves the diagnosis of the binary network but cannot fully capture causality of a given situation. The ongoing work focuses on identifying an efficient method to map a given causal model into a BBN maintaining as much as possible of the causal structure in a given situation while leveraging the possibilities of a trinary representation of process variables.

Acknowledgment

The authors thank Hugin Expert A/S for providing the trial license of Hugin Researcher used to generate the results in this work. This work is funded by the Danish Hydrocarbon Research and Technology Centre (DHRTC) as part of the Operations and Maintenance Technology programme.

Representing Operational Modes for Situation Awareness

Denis Kirchhübel¹, Morten Lind¹ and Ole Ravn¹

¹Department for Electrical Engineering, Technical University of Denmark

Abstract:

Operating complex plants is an increasingly demanding task for human operators. Diagnosis of and reaction to on-line events requires the interpretation of real time data. Vast amounts of sensor data as well as operational knowledge about the state and design of the plant are necessary to deduct reasonable reactions to abnormal situations. Intelligent computational support tools can make the operator's task easier, but they require knowledge about the overall system in form of some model.

While tools used for fault-tolerant control design based on physical principles and relations are valuable tools for designing robust systems, the models become too complex when considering the interactions on a plant-wide level. The alarm systems meant to support human operators in the diagnosis of the plant-wide situation on the other hand fail regularly in situations where these interactions of systems lead to many related alarms overloading the operator with alarm floods. Functional modelling can provide a middle way to reduce the complexity of plant-wide models by abstracting from physical details to more general functions and behaviours. Based on functional models the propagation of failures through the interconnected systems can be inferred and alarm floods can potentially be reduced to their root-cause. However, the desired behaviour of a complex system changes due to operating procedures that require more than one physical and functional configuration. In this paper a consistent representation of possible configurations is deduced from the analysis of an exemplary start-up procedure by functional models.

The proposed interpretation of the modelling concepts simplifies the functional modelling of distinct modes. The analysis further reveals relevant links between the quantitative sensor data and the qualitative perspective of the diagnostics tool based on functional models. This will form the basis for the ongoing development of a novel real-time diagnostics system based on the on-line adaptation of the underlying MFM model.

D. Kirchhübel et al. (Jan. 2017a). "Representing Operational Modes for Situation Awareness". In: *13th European Workshop on Advanced Control and Diagnosis (ACD 2016)*. Vol. 783. 012055. Journal of Physics: Conference Series. DOI: 10.1088/1742-6596/783/1/012055.

G.1 Introduction

Modern complex production plants are becoming increasingly demanding due to the distributed nature of the control system and increased requirements of safe and efficient operations. While every component of a system may be within its operating margins, undesired interaction between certain component states can accumulate to catastrophic situations. It is therefore important to not only consider single components but the combination of all interrelated parts of a complex system. In order to cope with the complexity of systems different perspectives are relevant to represent the interactions within the system beyond different time scopes and levels of detail with regards to structure, function and behaviour (SFB). (Venkatasubramanian, 2011)

For robust and fault-tolerant systems tools, such as structural analysis have been developed to model the interrelations between subsystems and enable failure diagnosis. (Blanke et al., 2016) However, for plant-wide diagnosis industrial plants rely mostly on the experience and diagnostic skills of human operators. The operators establish the state of the plant based on alarms generated for possibly abnormal states indicated by sensor readings. (Wang et al., 2016a) While the concepts for fault-tolerant systems are used on the level of components or subsystems, the majority of improvements for alarm systems reviewed by Wang et al. (Wang et al., 2016a) disregard the process knowledge and rely on data driven methods. This can be related to the problem of high complexity of an overall plant described by Venkatasubramanian et al. (Venkatasubramanian, 2011), especially considering the extension and replacement of components throughout the live span of an industrial plant.

One way to overcome the lack of process knowledge incorporated in alarm systems and the high complexity of mathematical descriptions is presented by the SFB approach. SFB modelling provides an abstraction of the system, that can be used to analyse and diagnose the system based on the interaction of different structures, functions and purpose. One form of SFB modelling is Multilevel Flow Modelling (MFM), which provides a modelling language as well as a diagnostics tool for qualitative cause consequence reasoning to identify how abnormal states propagate through the system. (Venkatasubramanian, 2011; Zhang, 2015)

As addressed by hybrid systems modelling certain conditions or events in the control system lead to different states of a system by changing its behaviour. (Blanke et al., 2016) On a plant-wide level such discrete states can be generated by different configurations of subsystems because of operational procedures or to efficiently use redundant systems. Such configurations can be considered as operational modes of a plant. While operational modes have been the subject of MFM related research (Lind, 1992; Lind et al., 2012), no consistent and easy way of modelling these modes has been proposed. The research of operational modes of a nuclear power plant by Lind et al. (Lind et al., 2012) showed that each mode can be represented in a distinct MFM model with regards to functions and goals of the mode. Those distinct models, however, disregard the operational knowledge on how modes are interconnected and what the boundaries of the modes are with respect to operation procedures.

This paper describes a new way of interpreting control functions and relations as part of the operational knowledge included in MFM. Similar to hybrid systems modelling discrete events are identified that determine boundaries of a mode and facilitates the incorporation of operational procedures in MFM models through the generation of interpreted models that express implicit knowledge. The proposed interpretation reveals, how the constraints for validity of the qualitative model are closely linked to the quantitative aspects of real-time sensors or alarms and will form the basis of the ongoing development to link MFM modelling and artificial intelligence approaches to create a novel plant-wide on-line diagnostics system based on the cause and consequence reasoning in MFM.

In section G.2 different approaches to plant-wide on-line diagnostics are outlined and the basic concepts of modelling and reasoning in MFM are briefly described as well as the state of the art with regards to operational modes, especially in MFM. Section G.3 describes the concepts for linking control functions and modes based on discrete models of a start-up procedure. In section G.4 the mode models generated with the proposed interpretation are evaluated. Finally the conclusions drawn from this conceptual work and the future development based on this concept are outlined in section G.5.

G.2 State of the art

The study of Venkatasubramanian (Venkatasubramanian, 2011) describes how the computerization of industrial processes has lead to robust and fault tolerant control of components and that artificial intelligent approaches have been shown to enhance the maintenance scheduling and degradation diagnostics for specific applications. While these systems help improve the performance of a specific components most diagnostics systems are based on physical or statistical models of the system and would become too complex if all, possibly undesired, interactions between the components of the system are to be considered. From a system safety perspective the robustness of automated system parts does not make the entire system resilient to failure. One weak point is presented by the vast amount of information that a human operator needs to process, which makes the assessment of an emergency situation difficult, especially given the limited time frame for appropriate reaction. (Venkatasubramanian, 2011)

Modern industrial plants are equipped with a large number of sensors and control systems that can generate alarms to alert the human operator to a possible failure. With the increasing number of distributed control systems the number of alarms in a plant increases as well. While statistics driven analysis of plant data can aid auditing processes and filtering of excess alarms, the propagation of abnormal states due to the interconnections of components in the plant leads to alarm floods that overload the operator and make it hard to identify the origin of the failure. While alarm floods have been investigated by different groups, there is no established applicable solution to deal with alarm floods. Identifying related alarms that lead to alarm floods requires

a system with knowledge about the interactions of different components in the plant. (Wang et al., 2016a)

Incorporating the expert knowledge of human operators and plant designers into a support system can aid the decision making process by identifying root causes and planning appropriate reactions. To implement this approach rule based expert systems have been developed. These system can analyse a situation and infer procedures based on rules that are deducted from the expert knowledge of operators. A drawback of such systems is the domain specific nature of the expert knowledge and thus the narrow applicability of one such support system. (Cholewa, 2004) A more generic approach to communicating and evaluating expert knowledge is presented by functional modelling (FM). FM frameworks commonly abstract a system as a set of desired behaviours of the whole system or as a combination of functions of its components. This abstraction provides a general overview of the entire system by interconnecting different knowledge domains. The schematic representation of FM can facilitate computer reasoning about the entire system. (Erden et al., 2008)

The functional modelling language of MFM was originally developed as analysis tool to assist human operators in identifying and handling unknown operation situations (Burns and Vicente, 2001) and to support the design of human-machine interfaces (Lind, 2011). MFM has been demonstrated as a valuable tool to represent operational knowledge about processing plants in a range of technological areas in a machine readable form. Among others MFM is used for operator support scenarios in research projects such as the OECD Halden Reactor Project(Zhang, 2015), where the MFM framework is applied for cause and consequence reasoning about abnormal states of the plant. This kind of reasoning allows the operator to relate connected alarms and react more efficiently with a focus on the root causes and preserving essential system functions.(Zhang, 2015) Another branch of MFM has focused on deriving fault trees and failure mode analyses(Gofuku et al., 2006) and possible counter actions(Inoue et al., 2015) based on MFM.

G.2.1 Multilevel Flow Modelling

A MFM model is a hierarchical decomposition of goals to be achieved by certain functions of the system, as well as a part-whole decomposition of a system function into basic material and energy flow functions. MFM provides a graphical modelling language with symbolic representations of these basic flow functions and the relation between functions and objectives of the system. Figure G.1 shows the defined MFM primitives. In a MFM model the flow function primitives are usually used in several flow structures. The functions are connected by influence relations inside a flow structure and by means-end relations across decomposition levels representing the contribution to another function or an objective.

Besides the analysis and representation of function, MFM can be used to reason about the system performance. The reasoning is established in terms of qualitative performance of each function and the propagation of abnormal performance states by

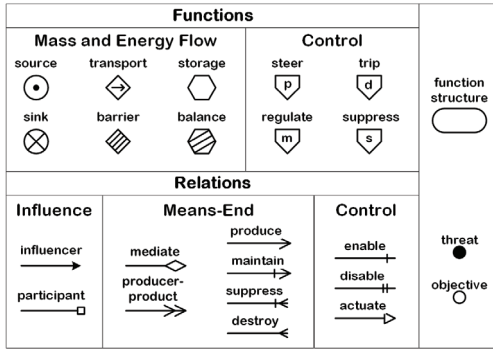


Figure G.1: Basic MFM concepts(Zhang, 2015)

Table G.1: MFM control functions(Lind, 2005)

Intention	action	symbol
produce	$[\neg pTpI\neg p]$	
maintain	$[pTpI\neg p]$	
destroy	$[pT\neg pIp]$	
suppress	$[\neg pT\neg pIp]$	

the relations of one function to another. Table G.2 shows the underlying equations and the qualitative states that form the basis for the reasoning system. Cause and consequence inference rules for the possible combinations of flow functions and relations have been detailed by Petersen(Petersen, 2000) and most recently elaborated on by Zhang et al. (Zhang, 2015; Zhang et al., 2013). The reasoning considers the abnormal states of functions in the way alarm systems commonly represent abnormal sensor readings as high (high-high) or low (low-low). Based on the interactions of the function primitives the propagation of abnormal states can be inferred in both forward (consequence) and backward (cause) direction.

Table G.2: Underlying equations, constraints and failure states of MFM flow functions(Petersen, 2000; Zhang, 2015)

Flow function	Balance equation	State Constraints	Abnormal states
Transport	$F_{in} = F_{out} = F$	$F_{low} \leq F \leq F_{high}$	low low, low, high, high high
Storage	$\Sigma F_{out} = \Sigma F_{in} + dV/dt$	$V_{low} \leq V \leq V_{high}$	low low, low, high, high high
Source	$\Sigma F_{out} = F_{unknown} + dV/dt$	$V_{low} \leq V \leq V_{high}$	low low, low, high, high high
Sink	$\Sigma F_{in} = F_{unknown} + dV/dt$	$V_{low} \leq V \leq V_{high}$	low low, low, high, high high
Balance	$\Sigma F_{out} = \Sigma F_{in}$		sourcing, leak, block
Barrier	$F_{in} = F_{out} = F$	$F = 0$	leak

In addition to the fundamental flow functions and objectives, MFM allows the modelling of control functions as means of intervention. Lind (Lind, 2005) introduced the action notation for the control functions in table G.1. This notation uses

the temporal operator T and an operator I , where the state after T is achieved by the control intervention instead of the state after I which the system would move to without intervention. These control functions are concerned with the functional meaning or intention of the control design e.g. to keep a flow in an heat exchanger steady rather than a specific realization of the controller. A control function is normally connected to an objective that represents the target function to be controlled. The actuate relation connects the control function to the functions that controlled to achieve that target, e.g. a pump is actuated in order to maintain a certain water level in a tank. Zhang et al. (Zhang et al., 2014) point out that the purpose of a control action can be modelled for automated as well as manual intervention. Control functions in MFM are thus a way of extending the model with expert knowledge about the way a plant is operated.

G.2.2 Operational Modes

In the context of diagnosis and faults-tolerant control a similar concept to operational modes is reflected in the hybrid nature of fault-tolerant systems. Fault-tolerant control reconfigures the structure and parameters based on logic as reactions to discrete events, such as faults in the system. The different configurations can be represented as distinct states, exposing a specific behaviours based on the configuration of the control. The evolution of these states can be described e.g. by Petri nets or similar representations. The combination of continuous model for each state and the discrete events limiting the validity of a state is represented in a hybrid systems model. In the hybrid model the discrete events limiting a state are described as constraints on specific system variables. (Blanke et al., 2016)

Operational modes as such have been investigated by Lind et al. (Lind et al., 2012) with regards to their representation in MFM. Operational modes can occur on two different levels of abstraction: as a relation of objective to function, or as a relation of function to physical structure. On either abstraction level modes can be defined both ways, as a selection of means to achieve a constant end, or as a selection of ends that can be achieved by the same means. Zhang (Zhang, 2015) elaborates that this classification is relevant to assess the operability of plant as indication for necessary mode shifts, e.g. configuration changes. Such a configuration change could be between redundant systems, changing the physical structure (means) to achieve the same function (end). Equally a configuration change could be opening a valve (means) thus changing its function (end) from blocking to transporting. Analogously a certain set of functions could serve different objectives, depending on the mode, or vice versa. However, the boundaries of one operational mode are at present not represented in the MFM models. Inoue et al. (Inoue et al., 2015) use MFM models to generate plausible operation procedures in unknown emergency situations. In order to generate these procedures knowledge about possible, and possibly undesired alternative functions of physical components has to be considered. This kind of situation relates directly to the function to structure mode definition by Lind (Lind

et al., 2012). Gofuku et al. (Gofuku et al., 2006) introduced operational information in addition to the MFM models to include the required knowledge. For the context of operational modes the most relevant aspects of operational information are the component behaviour and operation knowledge.

Component behaviour knowledge refers to the plausible behaviours of a physical structure and their functional representation. Operation knowledge represents the possible interventions and the functional influence of the intervention. Operation knowledge is essentially part of MFM models by including manual as well as automatic interventions in the modelled control functions (Zhang et al., 2014). The other aspects of operation information are not as clearly defined in the MFM Framework as the additional information Gofuku et al. (Gofuku et al., 2006) describe. Alternative behaviours of components have been widely disregarded by the MFM framework, since a MFM model is used to represent intended behaviour (Zhang, 2015).

G.3 Operational modes and control

In order to illustrate the concepts to consistently represent operational modes in MFM the discrete models of two modes in a start-up procedure of a generic power plant are analysed.

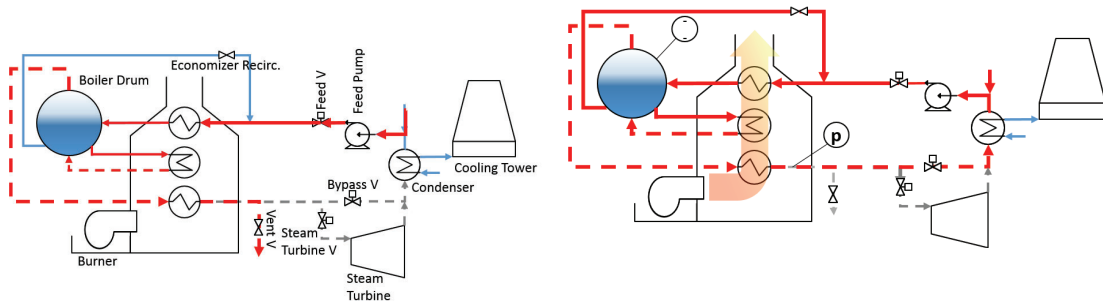


Figure G.2: Filling of a generic power plant

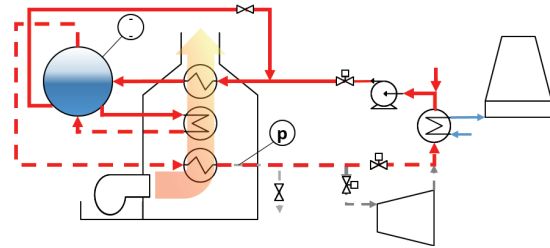


Figure G.3: Pressurizing of a generic power plant

To get the power plant up to operation the first step is filling the boiler drum and piping with water. Figure G.2 shows the active material flow path during this stage of the start-up procedure: Water is pumped from a reservoir into the boiler drum and the ventilation valve is left open to allow air to escape from the steam piping at the output of the boiler. The goal of this stage is to fill the drum to the required water level before the steam production can be initiated. The MFM model shown in figure G.4 reflects the described material flow with the function primitives of MFM. In addition to the intended material flow, the closed off parts of the system, namely the economizer recirculation and the recirculation of steam through the condenser are included as barriers in the MFM model. Including these barriers allows to consider

faulty valve behaviours in the reasoning. The control of the water level in the drum by actuating the feed pump is reflected by the control flow structure, specifying the intention of the mode as producing the required water level.

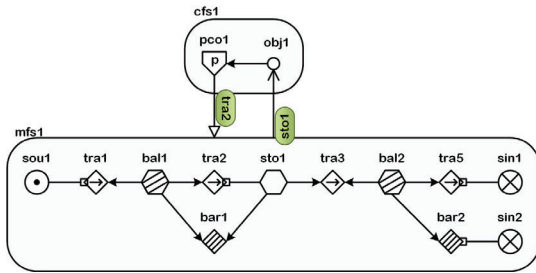


Figure G.4: MFM model of filling

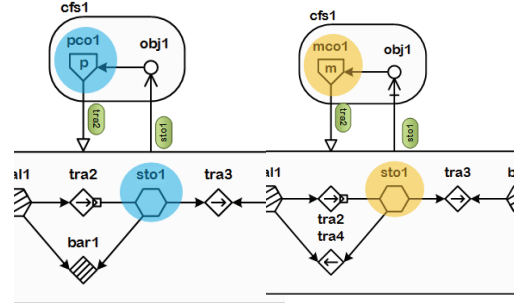


Figure G.5: Mode change indicators and corresponding control function in subsequent modes

Before the steam turbine can be operated to produce energy, pressurized steam has to be generated in order to provide enough energy to convert in the turbine. After steam has been generated is superheated and recirculated to raise the pressure in the system. The MFM model shown in figure G.6 is developed according to the active components in figure G.3. The mass flow structure of fuel and air in the burner is included in the model. The heat from the burner enables the introduction of the energy into the system as represented by the energy flow structure. In this mode the control of the temperature and pressure in the steam piping actuates the flow of steam through the condenser to raise the thermal energy in the steam piping. The overall goal of this stage of the operation procedure is to generate the necessary steam pressure to be able to spin up the turbine.

Comparing the functional representation of corresponding elements across the two modes reveals that the goal and end-point of these modes are related to the control actions, more specifically the end point of produce action. While the control function remains present in subsequent modes, the associated control action changes e.g. from produce to maintain as shown in figure G.5. The representation of valves that can either be closed or opened to change the physical configuration of the system is either a barrier or as a transport function.

Besides valves as a means of changing the system configuration, specific components, like the burner, can be enabled or disabled. This behaviour is reflected in the way that a disabled components function does not contribute to the function of the system and is thus not considered in the model.

The concepts described in the further part of this section have been developed based on these findings and present a pre-processing of a designed MFM model into an interpreted model. The interpreted models are intended to work with the established cause and consequence reasoning based on MFM.

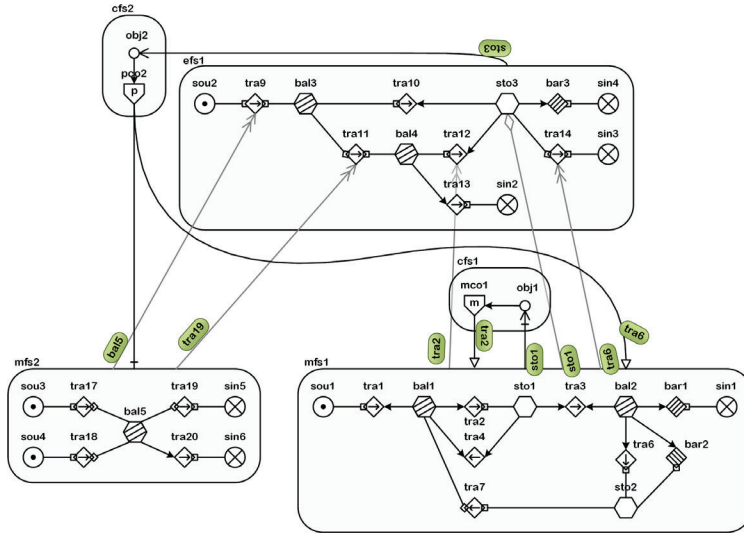
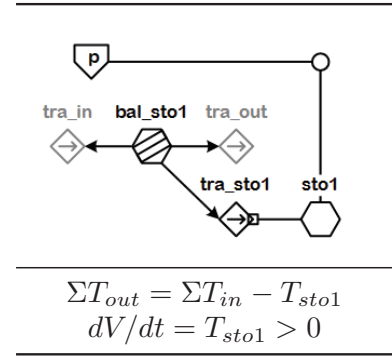


Figure G.6: MFM model of pressurizing

Table G.3: Intended behaviour of a storage with produce controller



G.3.1 Control sequence

The distinct models for the start-up modes have shown, that the end-point associated with the overall objective of one mode is linked to producing a certain storage level in order to prepare the system for the next operational mode. This objective of producing, thus increasing, a certain storage level can not be inferred from the underlying definition of a storage shown in table G.2. According to the definition of a storage the only constraint is on the level of the storage.

In order to reflect the intended aspect of a defined inflow in order to increase the storage level, the function of the storage has to be considered to act like a sink. The interpretation of a storage to be produced with a defined inflow is shown in table G.3. By explicitly including a transport function the inflow can be constraint with relation to the quantitative values in the physical system. Figure G.9 shows the cause reasoning output for a high level in sto1 of the fill mode represented in figure G.4. Considering the states as indicators rather than faults this can be interpreted as possible paths to achieving the goal of this mode: The storage level can be raised by generating a high inflow into the system as well as by keeping the outflow low (no water should leave through the vent valve).

Similar constraints can be introduced for all control functions in MFM. Based on the action schemes defined for the MFM control functions there is an inherent sequence of the control functions in relation to operational modes, as shown in figure G.7. Reasoning about the validity of one mode can be based on the quantitative constraints linked to the intended behaviour of a controlled storage. A breach of either of the constraints on the storage level or the timely change thereof, represented by the in or outflow, indicates a necessary configuration change or a failure of the controller.

Analogous to the concepts of fault-tolerant control the breach of the constraints can be interpreted as a discrete event that leads to a change in the system configuration. The discrete states of a fault-tolerant system as well as the designed operation modes of the overall plant can be defined and reasoned about as exemplified in figure G.8. An important difference, however, is the fact that the model for each mode expresses the boundaries it is designed for explicitly, as opposed to the continuous models in hybrid systems where assumptions for a mathematical model are implicit in the model and expressed explicitly only by the separately defined events.

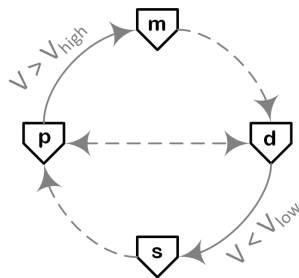


Figure G.7: Implicit (solid) and deliberate (dashed) control function sequence throughout modes

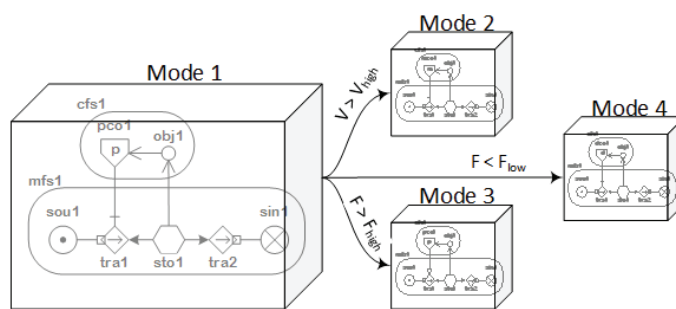


Figure G.8: Possible mode transition events with a single produced storage

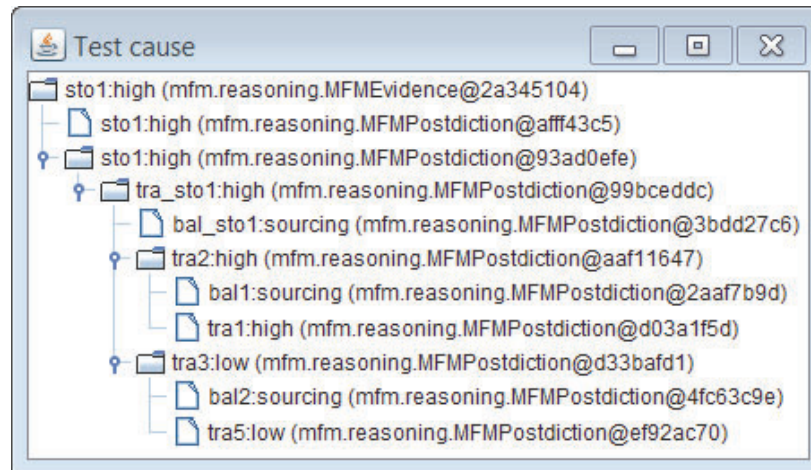


Figure G.9: MFM based cause tree for high water level in the generic powerplant during filling

G.3.2 Modelling configuration

The difference in the configurations considered in the distinct models of the power plant start-up can be represented by transport or barrier functions reflecting the state of a valve. As described in the previous section the reasoning about operational modes is closely related to the intention represented by control functions. Consequently, the control relations are explored as a means of representing configuration in MFM.

MFM defines three control relations, where the enable and disable relation directly correspond to the two complementary states a configuration valve can have (open or close). However, the functional representation of a configuration valve can coincide with the function of a continuously actuated control valve. The enable and actuate relations are thus interpreted to reflect the normal function, while the disable relation is interpreted in the preprocessing to reflect a closed valve or deactivated component.

The analysis of the start-up procedure revealed that a closed valve does not only affect the MFM function representing it, but also propagates through the means-end relations, specifically the mediate and producer-product relations. In effect, if a transport function as producer is interpreted as barrier, the transport function, that is the product, also has to be interpreted as a barrier.

Using the concepts for interpretation proposed here, allows for a mode to be represented by a set of control functions that reflect the objective of the mode and a set of control relations to reflect the configuration. This set can be defined as additional information for a MFM model as reference, thus adding a two step process to generate the interpreted models for the MFM based cause and consequence reasoning: The first step is model generation, that adapts the reference model to contain the control functions and relations for the respective mode. The second step is the interpretation of the control relations and the functions to represent the intended behaviour of the mode in an interpreted model that is subsequently used for the reasoning.

G.4 Generating models

By joining the MFM model of all modes a reference model to accommodate all modes can be derived. This model essentially reflects all possible flows as they could for example be found in a piping and instrumentation diagram. An exemplary combined model for the generic thermo-electric power plant is shown in figure G.10

Using the proposed interpretation concepts the MFM representation for a specific mode can be generated. For the sake of readability the explicit modelling of the intention of control has been left out of the interpreted model for filling shown in figure G.11. Comparing this model to the discretely modelled representation of the same mode in figure G.4 there are some obvious differences.

The central mass flow structure of the discretely developed model for the filling mode did not contain the recirculation of water through the condenser, since that part of the system is not part of the functional scope of the mode. However, the functional

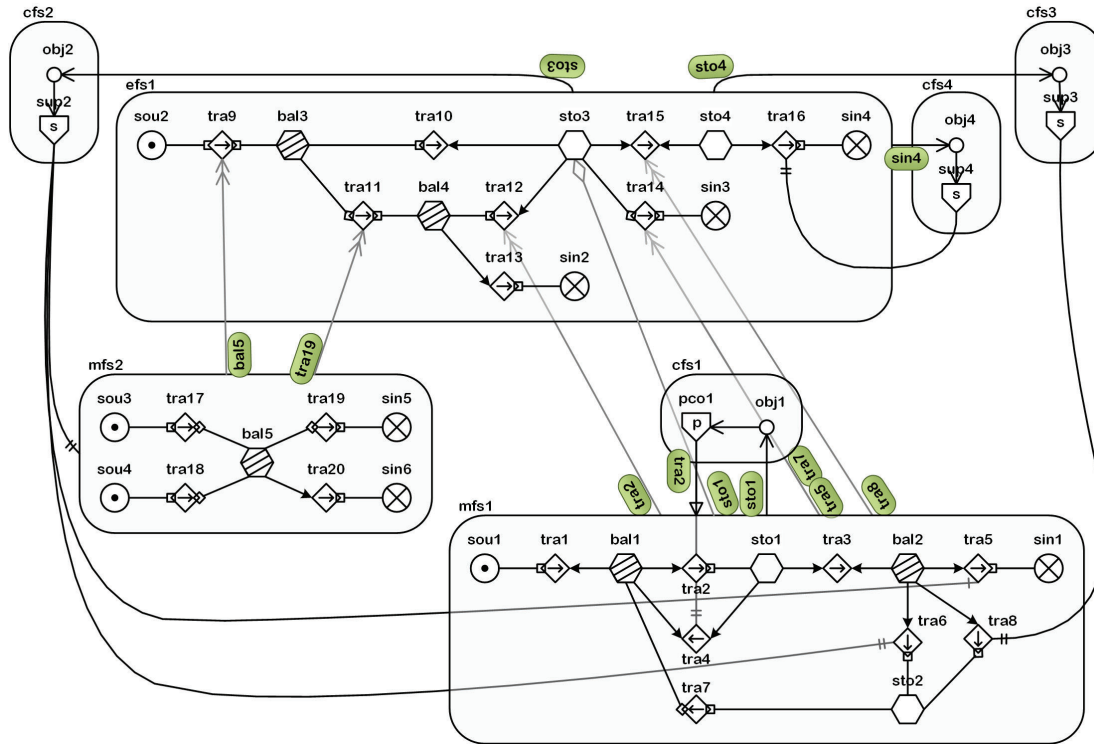
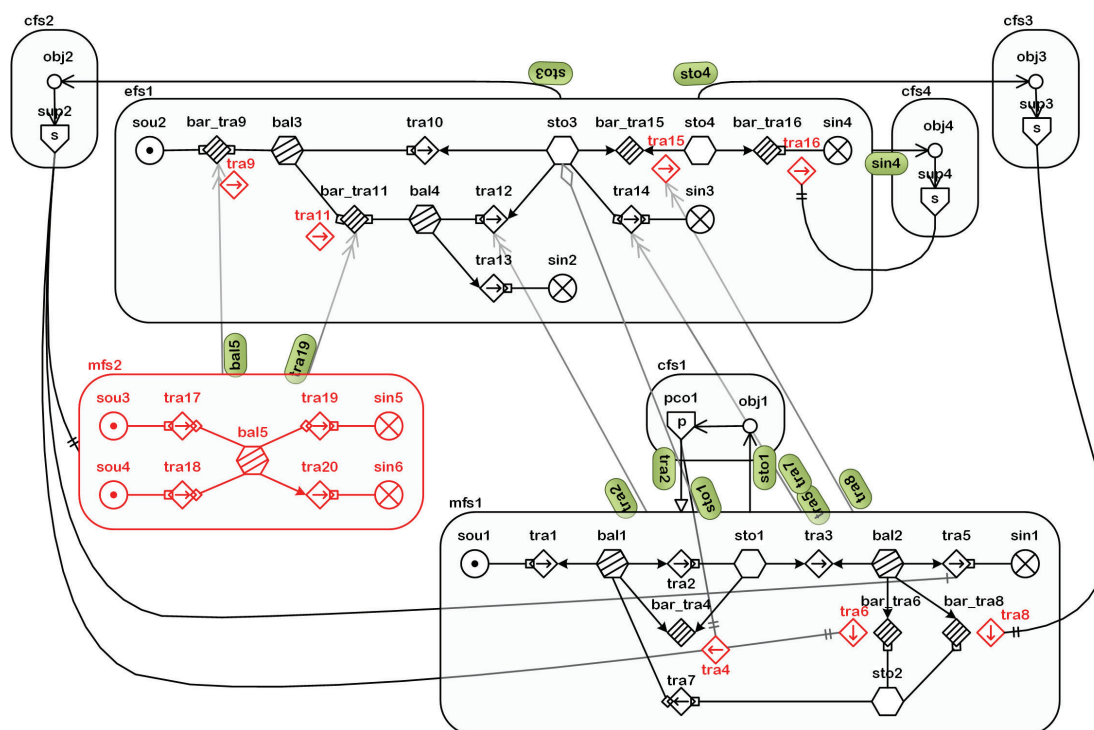


Figure G.10: Reference model (fill mode) for a generic thermo-electric power plant

information is fundamentally the same, given that the barrier at the output of the discrete model is reflected more in detail by the two barriers in the recirculation.

The energy flow, while not relevant to the function of this mode, is represented in the interpreted model. A detailed inspection of that energy flow reveals, that there is no intended energy flow: From the source in the energy flow only barriers connect to the other functions and the two storages are connected to suppress control functions, thus acting as balances.



G.5 Conclusion

Functional modelling can provide the framework for representing process knowledge for a range of engineering applications. Incorporating such knowledge in a reasoning systems enables the creation of operator support tools that can help diagnosing the current state of a complex plant and provide guidance for reasonable reactions. This kind of plant-wide diagnosis is believed to be capable of relieving the load of alarms operators are confronted with by identifying connected alarms and thus yielding more meaningful information. MFM presents such a functional modelling framework with an established reasoning system for cause and consequence diagnostics.

In this work the distinctive elements of operational modes and the boundaries of each mode have been investigated with regards to functional modelling. Based on the analysis of distinct functional models for a start-up procedure concepts for the consistent representation of operational modes in MFM are proposed. These concepts facilitate the modelling of operational modes in MFM by using a common reference model as basis for the generation of specific mode models. The common model can be derived from existing engineering documentation, such as piping and instrumentation diagrams, and thus simplifies the modelling process in MFM.

G.6 Future work

The proposed interpretation of the intention of each mode provides the means of linking the models for specific configurations to the real-time environment equipped with sensors. In the development process for a complete real-time diagnostics system based on MFM these concepts will serve as the basis for dynamically adapting the functional model to the actual state of the plant. An important element of this is to provide the rule system with knowledge about manual and automatic intervention points in the plant and their possible functions, as it is realized through the control relations in the proposed concepts.

The future effort in this field will be directed to the application of machine learning approaches to facilitate the identification of failure states from real-time sensor data, as well as the consolidation of the MFM based reasoning to enable on-line adaptation of the model to reflect the current state of the system. The goal for this project is to provide a novel kind of diagnostics system, that incorporates the operational knowledge and enables an organised representation of the state of a plant by identifying relevant failure paths from the on-line data.

Bibliography

- Abdallah, I., A. L. Gehin, and B. Ould Bouamama (2018). “On-line robust graphical diagnoser for hybrid dynamical systems”. In: *Engineering Applications of Artificial Intelligent* 69, pp. 36–49. DOI: 10.1016/j.engappai.2017.12.002.
- Abele, L., M. Anic, T. Gutmann, J. Folmer, M. Kleinstember, and B. Vogel-Heuser (2013). “Combining Knowledge Modeling and Machine Learning for Alarm Root Cause Analysis”. In: *IFAC Proceedings Volumes* 46.9, pp. 1843–1848. DOI: 10.3182/20130619--3-RU-3018.00057.
- Antonucci, A. (2011). “The Imprecise Noisy-OR Gate”. In: *Information Fusion*, pp. 709–715.
- Arroyo Esquivel, E. (2017). “Capturing and Exploiting Plant Topology and Process Information as a Basis to Support Engineering and Operational Activities in Process Plants”. PhD. Helmut-Schmidt-Universität, Hamburg.
- Azam, M., S. Ghoshal, S. Deb, K. Pattipati, D. Haste, S. Mandal, and D. Kleinman (Mar. 2014). “Integrated diagnostics and time to maintenance estimation for complex engineering systems”. In: *2014 IEEE Aerospace Conference*. IEEE, pp. 1–10. DOI: 10.1109/AERO.2014.6836478.
- Basu, C., K. Das, J. Hazra, and D. P. Seetharam (Oct. 2013). “Enhancing wide-area monitoring and control with intelligent alarm handling”. In: *IEEE PES ISGT Europe 2013*. IEEE, pp. 1–5. DOI: 10.1109/ISGTEurope.2013.6695338.
- Beebe, D., S. Ferrer, and D. Logerot (Mar. 2013). “The Connection of peak alarm rates to plant incidents and what you can do to minimize”. In: *Process Safety Progress* 32.1, pp. 72–77. DOI: 10.1002/prs.11539.
- Blanke, M., M. Kinnaert, J. Lunze, and M. Staroswiecki (2016). *Diagnosis and Fault-Tolerant Control*. Berlin, Heidelberg: Springer Berlin Heidelberg. DOI: 10.1007/978-3-662-47943-8.
- Bobbio, A., L. Portinale, M. Minichino, and E. Ciancamerla (1999). “Comparing Fault Trees and Bayesian Networks for Dependability Analysis”. In: *Computer Safety, Reliability and Security. SAFECOMP 1999*. Springer, Berlin, Heidelberg, pp. 310–322. DOI: 10.1007/3-540-48249-0_27.
- Bondy, J. A. and U. S. R. Murty (1976). *Graph Theory with Applications*. New York: Elsevier.
- Borutzky, W. (2010). *Bond Graph Methodology*. Springer London.
- Burns, C. and K. Vicente (2001). “Model-based approaches for analyzing cognitive work: A comparison of abstraction hierarchy, multilevel flow modeling, and decision ladder modeling”. In: *International Journal of Cognitive Ergonomics* 5.3, pp. 357–366.
- Cai, B., H. Liu, and M. Xie (2016). “A real-time fault diagnosis methodology of complex systems using object-oriented Bayesian networks”. In: *Mechanical Systems and Signal Processing* 80, pp. 31–44. DOI: 10.1016/j.ymssp.2016.04.019.

- Charbonnier, S., N. Bouchair, and P. Gayet (Oct. 2014). "Analysis of fault diagnosability from SCADA alarms signatures using relevance indices". In: *2014 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, pp. 2739–2744. DOI: 10.1109/SMC.2014.6974342.
- Cholewa, W. (2004). "Expert Systems in Technical Diagnostics". In: *Fault Diagnosis*. Berlin: Springer, pp. 591–631.
- Crowl, D. A. and J. F. Louvar (2011). *Chemical Process Safety - Fundamentals with Applications*. Third. Paul Boger, Prentice Hall.
- D'Angelo, M., R. M. Palhares, M. C. Camargos Filho, R. D. Maia, J. B. Mendes, and P. Y. Ekel (2016). "A New Fault Classification Approach Applied to Tennessee Eastman Benchmark Process". In: *Applied Soft Computing* 49.1568-4946, pp. 676–686.
- Dahlstrand, F. (1998). "Alarm analysis with fuzzy logic and multilevel flow models". In: *Proceedings of the 18th Annual International Conference of the British Computer Society Special Group on Expert Systems*, pp. 173–188.
- (2002). "Consequence analysis theory for alarm analysis". In: *Knowledge-Based Systems* 15.1-2, pp. 27–36.
- Dong, G., W. Chongguang, Z. Beike, and M. Xin (2010). "Signed Directed Graph and Qualitative Trend Analysis Based Fault Diagnosis in Chemical Industry". In: *Chinese Journal of Chemical Engineering* 18.2, pp. 265–276. DOI: 10.1016/S1004-9541(08)60352-3.
- Downs, J. J. and E. F. Vogel (1993). "A plant-wide industrial process control problem". In: *Computers & Chemical Engineering*. DOI: 10.1016/0098-1354(93)80018-I.
- Duan, P., T. Chen, S. L. Shah, and F. Yang (June 2014). "Methods for root cause diagnosis of plant-wide oscillations". In: *AIChE Journal* 60.6, pp. 2019–2034. DOI: 10.1002/aic.14391.
- Dubois, L., J.-M. Forêt, P. Mack, and L. Ryckaert (2010). "Advanced logic for alarm and event processing: Methods to reduce cognitive load for control room operators". In: *IFAC Proceedings Volumes* 43.13, pp. 158–163. DOI: 10.3182/20100831-4-FR-2021.00029.
- EEMUA, E. (2013). "Alarm systems - A guide to design, management and procurement". In: *EEMUA Publication 191*.
- EPA, ((2007). "Guidance for Preparing Standard Operating Procedures". In: *EPA QA/G-6* April.
- Erden, M., H. Komoto, T. van Beek, V. D'Amelio, E. Echavarria, and T. Tomiyama (2008). "A review of function modeling: Approaches and applications". In: *Artificial Intelligence in Engineering Design and Manufacture* 22.02, pp. 147–169. DOI: 10.1017/S0890060408000103.
- Fang, M. (1994). *MFM Modelling Method and Application*. Tech. rep. 94-D-713. Lyngby, Denmark: Technical University of Denmark.
- Fang, M. and M. Lind (1995). "Model Based Reasoning Using MFM". In: *Proc. Pacific-Asian Conf. Expert Syst.* Huangshan, China.

- Folmer, J., F. Schuricht, and B. Vogel-Heuser (2014). "Detection of Temporal Dependencies in Alarm Time Series of Industrial Plants". In: *IFAC Proceedings Volumes* 47.3, pp. 1802–1807. DOI: 10.3182/20140824--6-ZA-1003.01897.
- Georges, A., D. Buytaert, and L. Eeckhout (2007). "Statistically rigorous java performance evaluation". In: *ACM SIGPLAN Notices* 42.10, p. 57. DOI: 10.1145/1297105.1297033.
- Goel, P., A. Datta, and M. S. Mannan (2017). "Industrial alarm systems: Challenges and opportunities". In: *Journal of Loss Prevention in the Process Industries* 50, pp. 23–36. DOI: 10.1016/j.jlp.2017.09.001.
- Gofuku, A. (2011). "Applications of MFM to intelligent systems for supporting plant operators and designers: function-based inference techniques". In: *Nuclear Safety and Simulation* 2.3, pp. 235–246.
- Gofuku, A., S. Koide, and N. Shimada (2006). "Fault Tree Analysis and Failure Mode Effects Analysis Based on Multi-level Flow Modeling and Causality Estimation". In: *SICE-ICASE International Joint Conference*. Tokyo, Japan, pp. 497–500.
- Gofuku, A. and Y. Tanaka (Aug. 1997). "A Combination of Qualitative Reasoning and Numerical Simulation to Support Operator Decisions in Anomalous Situations". In: *Proc. 3rd IJCAI Work. Eng. Probl. Qual. Reason.* Pp. 19–27.
- Guo, C., W. Hu, S. Lai, F. Yang, and T. Chen (Sept. 2017). "An accelerated alignment method for analyzing time sequences of industrial alarm floods". In: *Journal of Process Control* 57, pp. 102–115. DOI: 10.1016/J.JPROCONT.2017.06.019.
- Guo, W., F. Wen, Z. Liao, L. Wei, and J. Xin (Oct. 2010). "An Analytic Model-Based Approach for Power System Alarm Processing Employing Temporal Constraint Network". In: *IEEE Transactions on Power Delivery* 25.4, pp. 2435–2447. DOI: 10.1109/TPWRD.2009.2032054.
- Haasl, D., N. Roberts, W. Vesely, and F. Goldberg (1981). *Fault tree handbook*. Tech. rep. Nuclear Regulatory Commission, Washington, DC (USA). Office of Nuclear Regulatory Research.
- Hallgrímsson, Á. D., H. H. Niemann, and M. Lind (Aug. 2019). "Autoencoder Based Residual Generation for Fault Detection of Quadruple Tank System". In: *2019 IEEE Conference on Control Technology and Applications (CCTA)*, pp. 994–999. DOI: 10.1109/CCTA.2019.8920588.
- Heussen, K. and M. Lind (Aug. 2012). "On support functions for the development of MFM models". In: *Proceedings of the First International Symposium on Socially and Technically Symbiotic System*. Okayama, Japan.
- Hollifield, B. R. and E. Habibi (2006). *The Alarm Management Handbook: A Comprehensive Guide*. PAS.
- Hollnagel, E. (2002). "Time and time again". In: *Theoretical Issues in Ergonomics Science* 3.2, pp. 143–158. DOI: 10.1080/14639220210124111.
- Hu, J. and Y. Yi (2016). "A two-level intelligent alarm management framework for process safety". In: *Safety Science* 82, pp. 432–444. DOI: 10.1016/j.ssci.2015.10.005.
- Hu, J., L. Zhang, Z. Cai, and Y. Wang (2015). "An intelligent fault diagnosis system for process plant using a functional HAZOP and DBN integrated methodology".

- In: *Engineering Applications of Artificial Intelligence* 45, pp. 119–135. DOI: 10.1016/j.engappai.2015.06.010.
- Hu, W., T. Chen, and S. L. Shah (2017a). “Discovering Association Rules of Mode-Dependent Alarms From Alarm and Event Logs”. In: *IEEE Trans. Control Syst. Technol.*, pp. 1–13. DOI: 10.1109/TCST.2017.2695169.
- Hu, W., S. L. Shah, and T. Chen (Oct. 2017b). “Framework for a smart data analytics platform towards process monitoring and alarm management”. In: *Computers & Chemical Engineering*. DOI: 10.1016/J.COMPCHENG.2017.10.010.
- Hu, W., J. Wang, T. Chen, and S. L. Shah (July 2017c). “Cause-effect analysis of industrial alarm variables using transfer entropies”. In: *Control Engineering Practice* 64, pp. 205–214. DOI: 10.1016/j.conengprac.2017.04.012.
- IEC, I. (2014). “Management of alarms systems for the process industries”. In: *IEC 62682* Edition 1. P. 13.
- Inoue, T. and A. Gofuku (2016). “A technique to prioritize plausible counter operation procedures in an accidental situation of plants”. In: *Nuclear Safety and Simulation* 7.2.
- Inoue, T., A. Gofuku, and T. Sugihara (2015). “A technique to generate plausible operating procedure for an emergency situation based on a functional model”. In: *Proceedings of STSS/ISSNP 2015*. Kyoto, Japan.
- ISA, I. (2009). “Management of Alarm Systems for the Process Industries”. In: *ANSI/ISA-18.2-2009*.
- Jensen, F. V. (1996). *An introduction to Bayesian networks*. UCL Press, p. 178.
- Kim, S. G. and P. H. Seong (2018). “Enhanced reasoning with multilevel flow modeling based on time-to-detect and time-to-effect concepts”. In: *Nuclear Engineering and Technology* 50.4, pp. 553–561. DOI: 10.1016/j.net.2018.03.008.
- Kirchhübel, D. and T. M. Jørgensen (2019). “Generating Diagnostic Bayesian Networks from Qualitative Causal Models”. In: *24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*. IEEE, pp. 1239–1242.
- Kirchhübel, D., M. Lind, and O. Ravn (Jan. 2017a). “Representing Operational Modes for Situation Awareness”. In: *13th European Workshop on Advanced Control and Diagnosis (ACD 2016)*. Vol. 783. 012055. Journal of Physics: Conference Series. DOI: 10.1088/1742-6596/783/1/012055.
- (2019a). “Combining Operations Documentation and Data to Diagnose Procedure Execution”. *Computers and Chemical Engineering*. accepted in Nov 2019 pending revision.
- (2019b). “Dynamic Reasoning in Functional Models for Multiple Fault Diagnosis”. *Computers and Chemical Engineering*. submitted in April 2019.
- (2019c). “Toward Comprehensive Decision Support Using Multilevel Flow Modeling”. In: *5th IFAC Conference on Intelligent Control and Automation Sciences*. Belfast, UK: IFAC-PapersOnLine.
- Kirchhübel, D., X. Zhang, M. Lind, and O. Ravn (2017b). “Identifying causality from alarm observations”. In: *International Symposium on Future I&C for Nuclear Power Plants (ISOFIG) 2017*. Gyeongju, Korea, pp. 1–6.

- Kjærulff, U. B. and A. L. Madsen (2013). *Bayesian Networks and Influence Diagrams: A Guide to Construction and Analysis*. Vol. 22. Information Science and Statistics. New York, NY: Springer New York. DOI: 10.1007/978-1-4614-5104-4.
- Koffskey, C., L. H. Ikuma, and C. Harvey (2013). “Performance Metrics for Evaluating Petro-chemical Control Room Displays”. In: *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* 57.1, pp. 1717–1721. DOI: 10.1177/1541931213571383.
- Kramer, M. A. and B. L. Palowitch (1987). “A rule-based approach to fault diagnosis using the signed directed graph”. In: *AIChE Journal* 33.7, pp. 1067–1078. DOI: 10.1002/aic.690330703.
- Laberge, J. C., P. Bullemer, M. Tolsma, D. Vernon, and C. Reising (2014). “Addressing alarm flood situations in the process industries through alarm summary display design and alarm response strategy”. In: *International Journal of Industrial Ergonomics* 44, pp. 395–406. DOI: 10.1016/j.ergon.2013.11.008.
- Lai, S. and T. Chen (Jan. 2017). “A method for pattern mining in multiple alarm flood sequences”. In: *Chemical Engineering Research and Design* 117, pp. 831–839. DOI: 10.1016/J.CHERD.2015.06.019.
- Lampis, M. (2010). “Application of Bayesian Belief Networks to System Fault Diagnostics”. PhD thesis. Loughborough University, p. 184.
- Landman, R., J. Kortela, Q. Sun, and S.-L. Jämsä-Jounela (2014). “Fault propagation analysis of oscillations in control loops using data-driven causality and plant connectivity”. In: *Computers and Chemical Engineering* 71, pp. 446–456. DOI: 10.1016/j.compchemeng.2014.09.017.
- Larsson, J. E. (1996). “Diagnosis based on explicit means-end models.” In: *Artificial Intelligence* 80(1), pp. 29–93.
- (2002). “Diagnostic reasoning based on means-end models: Experiences and future prospects”. In: *Knowledge-Based Systems* 15.1-2, pp. 103–110. DOI: 10.1016/S0950-7051(01)00126-5.
- Larsson, J. E., J. Ahnlund, T. Bergquist, F. Dahlstrand, B. Öhman, and L. Spaanenburg (2004). “Improving expressional power and validation for multilevel flow models”. In: *Journal of Intelligent & Fuzzy Systems: Applications in Engineering and Technology* 15.1, pp. 61–73.
- Larsson, J. E., B. Öhman, and A. Calzada (2007). “Real-Time Root Cause Analysis for Power Grids”. In: *Security and Reliability of Electric Power Systems*. Tallinn, Estonia, pp. 1–7.
- Larsson, J. E., B. Öhman, A. Calzada, C. Nihlwing, H. Jokstad, L. I. Kristianssen, J. Kvaem, and M. Lind (2006). “A Revival of the Alarm System: Making the Alarm List Useful During Incidents”. In: *Proceedings of the 5. International Topical Meeting on Nuclear Plant Instrumentation Controls, and Human Machine Interface Technology*, pp. 1–6.
- Lind, M. (1991). “ABSTRACTIONS for Modelling of Diagnostic Strategies”. In: *Proc. IFAC Workshop on Computer Software Structures Integrating AI/KBS Systems in Process Control*. Bergen, Norway.

- Lind, M. (1992). "A Categorization of Models and Its Application for the Analysis of Planning Knowledge". In: *Post ANP'92 Conference Seminar at Human Cognitive and Cooperative Activities in Advanced Technological Systems*. Kyoto, Japan.
- (2005). *Modeling goals and functions of control and safety systems - theoretical foundations and extensions of MFM*. Nordic nuclear safety research.
- (2011). "An introduction to multilevel flow modeling". In: *Nuclear Safety and Simulation*.
- (Sept. 2013). "An overview of multilevel flow modeling". In: *Nuclear Safety and Simulation* 4.3, pp. 186–191.
- (2017). "Knowledge Acquisition and Strategies for Multilevel Flow Modelling". In: *International Symposium on Future I&C for Nuclear Power Plants (ISOFIG) 2017*. Gyeongju, Korea, pp. 1–8.
- Lind, M., H. Yoshikawa, S. B. Jørgensen, and M. Yang (2012). "Modeling Operating Modes for the Monju Nuclear Power Plant". In: *International Journal of Nuclear Safety and Simulation* 3.4, pp. 314–324.
- Lunze, J. (Sept. 2004). "Complexity reduction in state observation of stochastic automata". In: *IFAC Proceedings Volumes* 37.18, pp. 339–344. DOI: 10.1016/S1474--6670(17)30769--3.
- Lv, N. and X. Wang (Feb. 2007). "SDG-based hazop and fault diagnosis analysis to the inversion of synthetic ammonia". In: *Tsinghua Science and Technology* 12.1, pp. 30–37. DOI: 10.1016/S1007-0214(07)70005-6.
- Ma, X. and D. Li (2017). "A Hybrid Fault Diagnosis Method Based on Fuzzy Signed Directed Graph and Neighborhood Rough Set". In: *2017 IEEE 6th Data Driven Control and Learning Systems Conference*. Chongqing, China, pp. 253–259.
- Maurya, M. R., R. Rengaswamy, and V. Venkatasubramanian (2007). "a Signed Directed Graph and Qualitative Trend Analysis-Based Framework for Incipient Fault". In: *Chemical Engineering Research and Design* 85.10, pp. 1407–1422. DOI: 10.1205/cherd06198.
- Milford, K. P. R. (2006). "An Efficient Framework for the Conversion of Fault Trees to Diagnostic Bayesian Network Models". In: *2006 IEEE Aerospace Conference*. IEEE, pp. 1–14. DOI: 10.1109/AERO.2006.1656103.
- Natarajan, S. and R. Srinivasan (2014). "Implementation of multi agents based system for process supervision in large-scale chemical plants". In: *Computers & Chemical Engineering* 60, pp. 182–196. DOI: 10.1016/j.compchemeng.2013.08.012.
- Nguyen, D. T., Q. B. Duong, E. Zamai, and M. K. Shahzad (Apr. 2016). "Fault diagnosis for the complex manufacturing system". In: *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability* 230.2, pp. 178–194. DOI: 10.1177/1748006X15623089.
- Nielsen, E. K., M. V. Bram, J. Frutiger, G. Sin, and M. Lind (Mar. 2018a). "A water treatment case study for quantifying model performance with multilevel flow modeling". In: *Nuclear Engineering and Technology*.
- Nielsen, E., S. Jespersen, X. Zhang, O. Ravn, and M. Lind (Jan. 2018b). "On-line Fault Diagnosis of Produced Water Treatment with Multilevel Flow Modeling".

- English. In: *IFAC-PapersOnLine* 51.8, pp. 225–232. DOI: 10.1016/j.ifacol.2018.06.381.
- Ouyang, J., M. Yang, H. Yoshikawa, and Y. Zhou (2005). “Modeling of power plant by multilevel flow model and its application in fault diagnosis”. In: *Journal of Nuclear Science and Technology* 42.8, pp. 695–705.
- Parasuraman, R., T. B. Sheridan, and C. D. Wickens (2000). “A model for types and levels of human interaction with automation - Systems, Man and Cybernetics, Part A, IEEE Transactions on”. In: *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans* 30.3, pp. 1–12.
- Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann Publishers, p. 552.
- Peng, D., Z. Geng, and Q. Zhu (June 2014). “A Multilogic Probabilistic Signed Directed Graph Fault Diagnosis Approach Based on Bayesian Inference”. In: *Industrial & Engineering Chemistry Research* 53.23, pp. 9792–9804. DOI: 10.1021/ie403608a.
- Petersen, J. (2000). “Causal reasoning based on MFM”. In: *Cognitive Systems Engineering in Process Control*. Taejon, Korea, pp. 36–43.
- Quiñones-Grueiro, M., A. Prieto-Moreno, C. Verde, and O. Llanes-Santiago (2019). “Data-driven monitoring of multimode continuous processes: A review”. In: *Chemo-metrics and Intelligent Laboratory Systems* 189. December 2018, pp. 56–71. DOI: 10.1016/j.chemolab.2019.03.012.
- Rangaiah, G. P. and V. Kariwala (2012). *Plantwide Control : Recent Developments and Applications*. Wiley, p. 478.
- Reinartz, C. C., D. Kirchhübel, O. Ravn, and M. Lind (2019). “Generation of Signed Directed Graphs Using Functional Models”. In: *5th IFAC Conference on Intelligent Control and Automation Sciences*. Belfast, UK: IFAC-PapersOnLine.
- Ricker, N. L. and J. H. Lee (1995). “Nonlinear modeling and state estimation for the Tennessee Eastman challenge process”. In: *Computers and Chemical Engineering* 19.9, pp. 983–1005. DOI: 10.1016/0098-1354(94)00113-3.
- Ricker, N. L. (Aug. 1996). “Decentralized control of the Tennessee Eastman Challenge Process”. In: *Journal of Process Control* 6.4, pp. 205–221. DOI: 10.1016/0959-1524(96)00031-5.
- (2019). *Tennessee Eastman Challenge Archive*. URL: <https://depts.washington.edu/control/LARRY/TE/download.html> (visited on 04/03/2019).
- Rodrigo, V., M. Chioua, T. Hagglund, and M. Hollender (2016). “Causal analysis for alarm flood reduction”. In: *IFAC-PapersOnLine* 49.7. DOI: 10.1016/j.ifacol.2016.07.269.
- Rossing, N. L., M. Lind, N. Jensen, and S. B. Jørgensen (2010). “A functional HAZOP methodology”. In: *Computers and Chemical Engineering* 34.2, pp. 244–253. DOI: 10.1016/j.compchemeng.2009.06.028.
- Rothenberg, D. H. (2009). *Alarm Management for Process Control: A Best-practice Guide for Design*. New York: Momentum Press.

- Sassen, J. M. A., P.C.Riedijk, and R. B. M. Jaspers (1991). "Using Multilevel Flow Models for fault diagnosis of industrial processes". In: *Proceedings of the 3rd European Conference on Cognitive Science Approaches to Process Control (CSAPC)*.
- Schleburg, M., L. Christiansen, N. F. Thornhill, and A. Fay (2013). "A combined analysis of plant connectivity and alarm logs to reduce the number of alerts in an automation system". In: *Journal of Process Control* 23.6, pp. 839–851. DOI: 10.1016/j.jprocont.2013.03.010.
- Simeu-Abazi, Z., A. Lefebvre, and J.-P. Derain (2011). "A methodology of alarm filtering using dynamic fault tree". In: *Reliability Engineering & System Safety* 96.2, pp. 257–266. DOI: 10.1016/j.ress.2010.09.005.
- Soares, V. B., J. C. Pinto, and M. B. de Souza (2016). "Alarm management practices in natural gas processing plants". In: *Control Engineering Practice* 55, pp. 185–196. DOI: 10.1016/j.conengprac.2016.07.004.
- Song, M. and A. Gofuku (2017). "Accident Management of the Station Blackout at BWR by Using Multilevel Flow Modeling". In: *International Symposium on Future I&C for Nuclear Power Plants (ISOFC) 2017*. Gyeongju, Korea, pp. 1–8.
- Srinivasan, R., P. Viswanathan, H. Vedam, and A. Nochur (Jan. 2005). "A framework for managing transitions in chemical plants". In: *Computers & Chemical Engineering* 29.2, pp. 305–322. DOI: 10.1016/J.COMPCHENG.2004.09.024.
- Takizawa, Y. and K. Monta (Nov. 1996). "Development of a Plant Diagnosis Method Based on a Human Cognitive Process". In: *Proc. 3rd European Conference on Cognitive Science Approaches to ProcessControl (CSAPC)*. Kyoto, Japan, pp. 99–106.
- Tchamova, A. and J. Dezert (Sept. 2012). "Intelligent alarm classification based on DS_mT". In: *2012 6th IEEE INTERNATIONAL CONFERENCE INTELLIGENT SYSTEMS*. IEEE, pp. 120–125. DOI: 10.1109/IS.2012.6335124.
- Us, T., J. Niels, L. Morten, and J. Sten Bay (2011). "Fundamental Principles of Alarm Design". In: *International Journal of Nuclear Safety and Simulation* 2.1, pp. 44–51.
- Venkatasubramanian, V. (Jan. 2011). "Systemic failures: Challenges and opportunities in risk management in complex systems". In: *AIChE Journal* 57.1, pp. 2–9. DOI: 10.1002/aic.12495.
- Venkatasubramanian, V., R. Rengaswamy, and S. N. Kavuri (2003a). "A review of process fault detection and diagnosis: Part II: Qualitative models and search strategies". In: *Computers & Chemical Engineering* 27.3, pp. 313–326. DOI: 10.1016/S0098-1354(02)00161-8.
- Venkatasubramanian, V., R. Rengaswamy, S. N. Kavuri, and K. Yin (2003b). "A review of process fault detection and diagnosis: Part III: Process history based methods". In: *Computers & Chemical Engineering* 27.3, pp. 327–346. DOI: 10.1016/S0098-1354(02)00162-X.
- Venkatasubramanian, V., R. Rengaswamy, K. Yin, and S. N. Kavuri (Mar. 2003c). "A review of process fault detection and diagnosis: Part I: Quantitative model-based methods". In: *Computers & Chemical Engineering* 27.3, pp. 293–311. DOI: 10.1016/S0098-1354(02)00160-6.

- Wan, Y., F. Yang, N. Lv, H. Xu, H. Ye, W. Li, P. Xu, L. Song, and A. K. Usadi (2013). "Statistical root cause analysis of novel faults based on digraph models". In: *Chemical Engineering Research and Design* 91.1, pp. 87–99. DOI: 10.1016/j.cherd.2012.06.010.
- Wang, J., J. Xu, and D. Zhu (June 2014). "Online root-cause analysis of alarms in discrete Bayesian networks with known structures". In: *Proceeding of the 11th World Congress on Intelligent Control and Automation*. IEEE, pp. 467–472. DOI: 10.1109/WCICA.2014.7052758.
- Wang, J., F. Yang, T. Chen, and S. L. Shah (Apr. 2016a). "An Overview of Industrial Alarm Systems: Main Causes for Alarm Overloading, Research Status, and Open Problems". In: *IEEE Transactions on Automation Science and Engineering* 13.2, pp. 1045–1061. DOI: 10.1109/TASE.2015.2464234.
- Wang, W. and M. Yang (Nov. 2016). "Implementation of an integrated real-time process surveillance and diagnostic system for nuclear power plants". In: *Annals of Nuclear Energy* 97, pp. 7–26. DOI: 10.1016/j.anucene.2016.06.002.
- Wang, W., M. Yang, and H. Seong (2016b). "Development of a rule-based diagnostic platform on an object-oriented expert system shell". In: *Annals of Nuclear Energy* 88, pp. 252–264. DOI: 10.1016/j.anucene.2015.11.008.
- Yang, F., P. Duan, S. L. Shah, and T. Chen (2014). *Capturing Connectivity and Causality in Complex Industrial Processes*. SpringerBriefs in Applied Sciences and Technology. Cham: Springer International Publishing. DOI: 10.1007/978-3-319-05380-6.
- Yang, F., S. Shah, and D. Xiao (2012). "Signed directed graph based modeling and its validation from process knowledge and process data". In: *International Journal of Applied Mathematics and Computer Science* 22.1, pp. 41–53. DOI: 10.2478/v10006-012-0003-z.
- Yin, S., S. X. Ding, A. Haghani, H. Hao, and P. Zhang (2012). "A comparison study of basic data-driven fault diagnosis and process monitoring methods on the benchmark Tennessee Eastman process". In: *Journal of Process Control* 22.9, pp. 1567–1581. DOI: 10.1016/j.jprocont.2012.06.009.
- Zerkaoui, S., F. Druaux, E. Leclercq, and D. Lefebvre (2010). "Indirect neural control for plant-wide systems: Application to the Tennessee Eastman Challenge Process". In: *Computers and Chemical Engineering* 34.2, pp. 232–243. DOI: 10.1016/j.compchemeng.2009.08.003.
- Zhang, X. (2015). "Assessing Operational Situations". PhD thesis. Technical University of Denmark.
- Zhang, X. and M. Lind (2017). "Reasoning about Cause-effect through Control Functions in Multilevel Flow Modelling". In: *International Symposium on Future I&C for Nuclear Power Plants (ISOFC) 2017*. Gyeongju, Korea, pp. 1–8.
- Zhang, X., M. Lind, S. B. Jorgensen, O. Ravn, and N. Jensen (2014). "Representing operational knowledge of pwr plant by using multilevel flow modelling". In: *Proceedings of the ISOFC/ISSNP*. Jeju, Rep. of Korea.

- Zhang, X., M. Lind, and O. Ravn (Aug. 2013). “Consequence Reasoning in Multi-level Flow Modelling”. In: *Analysis, Design, and Evaluation of Human-Machine Systems*. Vol. 12, pp. 187–194.
- Zhang, Z.-Q., C.-G. Wu, B.-K. Zhang, T. Xia, and A.-F. Li (2005). “SDG multiple fault diagnosis by real-time inverse inference”. In: *Reliability Engineering & System Safety* 87.2, pp. 173–189. DOI: 10.1016/j.ress.2004.04.008.
- Zhu, J., C. Wang, C. Li, X. Gao, and J. Zhao (2016). “Dynamic alarm prediction for critical alarms using a probabilistic model”. In: *Chinese Journal of Chemical Engineering* 24.7, pp. 881–885. DOI: 10.1016/j.cjche.2016.04.017.