Nuclear Engineering and Technology 55 (2023) 4408–4425

Contents lists available at ScienceDirect

Nuclear Engineering and Technology

journal homepage: www.elsevier.com/locate/net

Original article

MFM-based alarm root-cause analysis and ranking for nuclear power plants

Mengchu Song [a,*], Christopher Reinartz [a,b], Xinxin Zhang [a], Harald P.-J. Thunem [c], Robert McDonald [c]

[a] Department of Electrical and Photonics Engineering, Technical University of Denmark, 2800 Kongens Lyngby, Denmark
[b] Freiheit Technologies, 20359 Hamburg, Germany
[c] Institute for Energy Technology, 1777 Halden, Norway

ARTICLE INFO

Keywords:
Alarm analysis
Root-cause analysis
Root-cause ranking
Causal reasoning
MFM
Nuclear power plant

ABSTRACT

Alarm flood due to abnormality propagation is the most difficult alarm overloading problem in nuclear power plants (NPPs). Root-cause analysis is suggested to help operators in understand emergency events and plant status. Multilevel Flow Modeling (MFM) has been extensively applied in alarm management by virtue of the capability of explaining causal dependencies among alarms. However, there has never been a technique that can identify the actual root cause for complex alarm situations. This paper presents an automated root-cause analysis system based on MFM. The causal reasoning algorithm is first applied to identify several possible root causes that can lead to massive alarms. A novel root-cause ranking algorithm can subsequently be used to isolate the most likely faults from the other root-cause candidates. The proposed method is validated on a pressurized water reactor (PWR) simulator at HAMMLAB. The results show that the actual root cause is accurately identified for every tested operating scenario. The automation of root-cause identification and ranking affords the opportunity of real-time alarm analysis. It is believed that the study can further improve the situation awareness of operators in the alarm flooding situation.

1. Introduction

Alarm system is the main method to assist operators in the control room of nuclear power plant (NPP) in detecting process anomalies. However, if the plant is experiencing a major transient, when hundreds of alarms can occur at the same time, alarms may become distractions to operators and thus reduce operators' ability to cope with plant abnormalities [1,2]. In other words, the alarm overloading problem can make operators lose their situation awareness during off-normal conditions, which will further contribute to the risk of human errors and plant accidents. The recent digitization of control room may make the alarm overloading situation even worse since more alarms are easily configured [3]. The causes of alarm overloading include: i. frequently chattering due to noise; ii. incorrectly configured alarm variables; iii. process variables exceeding normal setpoints during mode changes that cause momentary alarms; iv. maintenance-related alarms that may not require actions by operators; v. abnormality propagation owing to physical connections, i.e. cause-consequence alarms [1,4]. The last cause is also known as alarm flood, or alarm cascade, which can be seen as the most difficult alarm problem. Not like the other alarm overload factors mentioned above, which are the main reasons of false alarms, in alarm cascade all occurred alarms should be considered as correctly tuned alarms, thus it cannot be solved by alarm reconfiguration or redesign. In other words, all alarms are relevant to the current plant condition, which require operators to understand and respond [5].

Therefore, it is necessary to develop advanced alarm processing techniques to improve usability of the alarm information and accordingly situation awareness of operators during alarm floods [1,6]. Particularly, in order to eliminate impacts from abnormality propagation, using predetermined static alarm priority levels as many of industries are adopting, may fail to provide operators optimal results, because the alarm priority is commonly not adapted to a specific alarm situation [1,7]. Instead, identification of root cause(s) of alarm flood and suppression of consequential alarms have been more recommended for alarm management [4], which is also useful in design of optimized alarm systems [8]. Simeu-Abazi et al. [9] extend the dynamic fault tree (DFT) with novel logic gates to filter and locate a failure from alarms. Abele et al. [10], Cai et al. [11], Hu and Yi [12] establish alarm models by using Bayesian networks (BNs) for the diagnosis of alarm root causes. Wen and Chang [13] develop an alarm processing method based on Tabu-Search (TS), which is a neighborhood search technique, to estimate possible events that causes a specific set of alarms. Lai and Chen [14] aim to reveal the correlations between alarms through pattern mining, so that the optimal alignment of alarm

* Corresponding author.
E-mail address: menso@dtu.dk (M. Song).

https://doi.org/10.1016/j.net.2023.07.034
Received 20 May 2023; Received in revised form 2 July 2023; Accepted 24 July 2023
Available online 1 August 2023

sequences can be identified, which is useful to find the root causes. Schleburg et al. [15] claim that the number of alarms presented to the operator can be reduced by grouping related alarms as one problem. Many researches have proven that the knowledge of plant connectivity is important for the alarm causal analysis [16–18]. In addition, it has been also claimed that historical data are useful for the identification of alarm root causes. For example, Parvez et al. [19] match the real-time alarm sequence with patterns in the alarm logs in order to early predict the incoming alarms. Similar works that explore the historical data can be found in Folmer et al. [20], Meng et al. [21], Zhou et al. [22]. Temporal information of alarms has also been suggested for the alarm root-cause analysis [23]. Laberge et al. [3] address alarm flooding by separating alarms in a time series. Similarly, Bauer and Thornhill [24] identify the propagation path via time delays between variables. However, EPRI [25] indicates that the temporal information of alarms could be probably invalidated for alarm root-cause analysis, considering many alarms receive their time stamps not at the time of occurrence but the time of arriving in the control room.

In this paper, alarm flood resulting from the abnormality propagation is addressed by using Multilevel Flow Modeling (MFM), which is a functional modeling methodology focusing on representing functions and goals [26]. MFM provides formalized methods for creating of qualitative representations of system objectives and functionalities, which potentially enable the qualitative reasoning about the causal dependencies among alarms. MFM has been extensively applied in alarm management. The earliest alarm root-cause analysis based on MFM is Larsson [27], who has included the method into a integrative model-based diagnostic reasoning strategy with the abilities of measurement validation and fault diagnosis [28]. The basic idea is using some generic rules implied in the connections of functions to match with an alarm situation, i.e. to decide which of alarms are primary, and which ones are perhaps consequential effects of the primary alarms. Compared to the above reviewed alarm analysis methods, MFM is in particular appropriate for the alarm problem in NPP as well as the other complex industrial systems, considering the following advantages:

i. MFM uses high level of abstraction makes plant knowledge acquisition, representation and validation relatively easier thus more viable for a complex system Larsson et al. [5].
ii. MFM is normative that describes how a system works normally, and alarms are those states deviated from the intended model. Because MFM is not descriptive for past occurred alarm events, it in fact can ensure the completeness within the modeling boundary that every occurred alarm can be captured in real-time [27].
iii. Unlike many data-driven approaches, which are usually lack of explainability for the obtained results due to the black-box nature [29], MFM offers explicit representation that helps operators understand not only the alarm problem but also the rationale of the root-cause analysis.

The original root-cause algorithm [27] has been extensively applied in various industries where there is the issue of alarm flooding. For instance, EPRI [25] develops an operator support system that can read alarm data from power grid control systems and present the initiating events and their causal alarm chains, which can help operators timely recover from major blackouts. Ouyang et al. [30] combine results from the MFM-based root cause analysis with the concept of ecological interface design (EID) to develop a supervisory system, which has been applied on a Japanese simulator of pressurized water reactor (PWR). A pilot MFM-based alarm system has been developed within a boiling water reactor (BWR) simulator at HAMMLAB (Halden man-machine laboratory) in Norway [31,32].

The MFM-based alarm analysis method has been constantly refined since developed. Dahlstrand [33] designs causal dependency graphs (CDGs) to represent causal relationships between functions, which are intuitive to define the causal reasoning rules. By using this new algorithm, MFM can not only be used to separate primary alarms from consequential alarms, but also to diagnose the root-cause faults that may not be observed as alarms. Dahlstrand [34] introduces fuzzy logic into MFM in order to cope with the uncertainty of root-cause reasoning due to noisy signals. Kirchhübel et al. [35] identify causality from alarm observations by using the new set of MFM rules developed by Zhang [36]. An alarm root-cause analysis paradigm that is distinct from [27] has been proposed, which is based on mutual reasoning of both cause and consequence for every single alarm. Note that both Dahlstrand [33] with the old algorithm and Kirchhübel et al. [35] with the new algorithm emphasize that multiple analyses should be combined to identify the root cause(s) that result in alarm floods. However, the existing MFM-based alarm analysis methods have several deficiencies, which hinder their use in real-time situations,

i. Matching patterns in alarms with predefined causal dependency graphs is difficult to handle with complex alarm problems.
ii. Applying rules in causal reasoning does not consider reasoning conflicts, which are easy to cause confused results.
iii. Due to the common feature of functional modeling that it normally does not give a certain result but provide many possibilities [37], the root-cause analysis cannot give the accurate root cause that actually leads to alarm flood.

This paper resents an automated root-cause analysis system based on the cause-consequence reasoning for every single alarm. The root causes identified by the system can either be an occurred alarm or a fault that may not be evidenced as an observation. An algorithm is proposed to address reasoning conflicts during the root-cause identification. In order to further improve the situation awareness of operators during alarm flooding situations given several possible root causes have been provided, it is also proposed a root-cause ranking algorithm, which can be used to isolate the actual root cause of alarm flood from all possible candidates found by causal reasoning on the MFM model. The validity of the novel algorithms presented in the paper is proven by several testing operating scenarios on one of simulators constructed at HAMMLAB called generic PWR (gPWR) simulator, which is referenced on an operating U.S. commercial NPP [38].

The remaining parts of the paper are organized as follows: Section 2 presents the MFM-based alarm analysis method, including a brief introduction of the MFM methodology, the method of alarm root-cause identification, and the method for ranking possible root causes. The possibility of real-time alarm analysis offered by the study will also be discussed. The proposed alarm analysis method has been validated on a PWR simulator. Section 3 introduces the functional modeling process for a PWR. The case study including the experimental setups and operating scenarios, as well as the application of alarm root-cause identification and ranking to the resulting MFM model of PWR are described in Section 4. A discussion of the obtained results is included in Section 5. A summary of the paper as well as a general evaluation of the findings is made in Section 6.

2. MFM-based alarm analysis method

This section presents the current MFM-based technique of alarm root-cause identification, which has been applied in the real-time alarm analysis. A set of ranking criteria are proposed, which can be used to evaluate root-cause candidates and determine the true one that results in the alarm flooding situation. A brief introduction to MFM will also be provided.

2.1. Multilevel flow modeling

Multilevel Flow Modeling (MFM) is a functional modeling approach that has been shown to be capable of capturing causal-dependencies of complex-processes like NPP, and has been enabled automated reasoning [39,40], fault diagnosis [41,42] and planning for severe accident

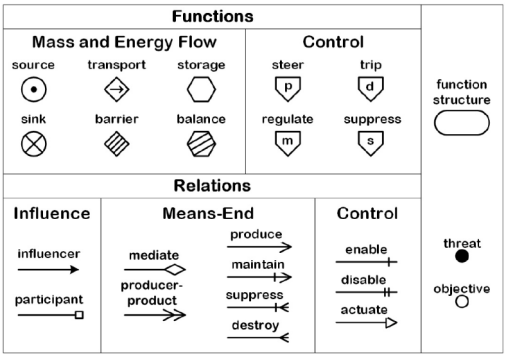


Fig. 1. MFM Flow functions and relations.

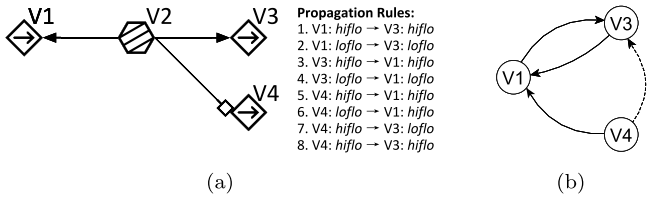


Fig. 2. Conversion from MFM to SDG: (a) Multilevel Flow Model of a hypothetical process with branching causal relations and the involved propagation rules (*hiflo*: high flow, *loflo*: low flow); (b) equivalent Signed Directed Graph representation excluding "V2". (In SDG, solid lines indicate positive effects, e.g. high → high, while dotted lines indicate negative effects, e.g. high → low).

management [43,44]. MFM describes the function-objective structure of a system in terms of a set of casually dependent mass and energy flows on different levels of abstraction. MFM models provide normative representations of a system, which describe system intentions, i.e. what the system should do, in contrast to a quantitative descriptive model which describes what the actual behavior of the systems is [45]. MFM is based on a set of generic flow functions and relations that are organized into mass, energy and control flow structures. The basic functions and relations of the MFM methodology are shown in Fig. 1.

Causal relations between functions and objectives, which are the most important aspects for the alarm analysis, can be separated into three groups:

(1) Influence relations that describe dependencies between functions within one flow structure.
(2) Means-end relations that describe cause–effect from functions to objectives or across flow structures.
(3) Conditional relations that describe conditional causal dependencies that are contingent on the achievement of process objectives.

The causality encoded in MFM relations is defined generically and independently of any particular domain of application. That means that as long as the system can be modeled using MFM, it can be evaluated using a generic set of inference rules that is tailored to the MFM modeling language. The set of generic inference rules used in MFM is separated into a rule-set for cause-reasoning, enabling root-cause analysis for a given set of symptoms, and consequence-reasoning, enabling prediction of potential effects of an observed scenario. The specific inference results that describe a causal propagation of process states in the system depend on three factors; the involved MFM functions, relations and the direction of flow of the system (upstream or downstream). A complete summary of inference rules is provided by [36]. The application of an inference rule between two MFM functions implies a propagation of a function state from a "source function" to a "target function". Valid states for the majority of the functions presented in Fig. 1, i.e. *source*, *sink*, *transport*, and *storage*, are *high*, *low* and *normal*. The *barrier* and *balance* functions are exceptions to this convention as the *barrier* can assume the states *normal* and *breach*, while *balance* is usually assumed that only has a *normal* state. A function may correspond to a process variable set with alarm threshold(s), exceeding of which, either in high (including breach) or low function state, can be considered as an alarm.

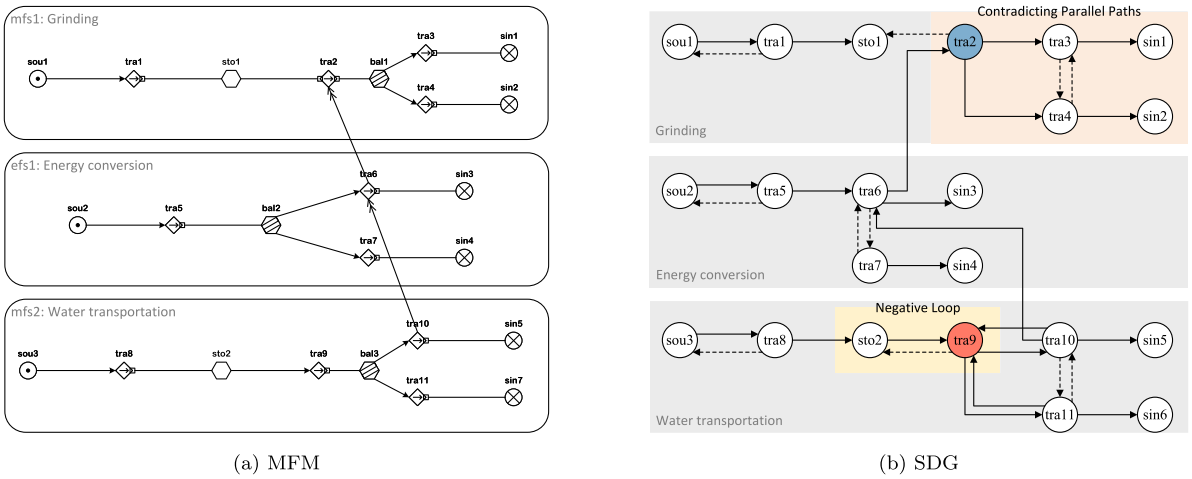
2.2. Converting MFM into SDG

As mentioned above, MFM encodes the causal relations between functions using specific propagation rules. As a result, it is possible to tailor the graphical modeling language to facilitate modeling of specific systems, while maintaining valid causal reasoning through a specifically designed set of propagation rules. Conversely, this means that the determination of cause–effect relations in MFM models requires specific knowledge of those MFM propagation rules, which can be inconvenient for users who are not familiar with the modeling methodology. In order to facilitate the causal analysis based on MFM, e.g. alarm root-cause identification, an MFM model can be converted into a causal model that can more explicitly describe the causal relations, i.e. signed directed graph (SDG) [46].

SDGs can be used to describe qualitative causal models generically [47]. Mathematically, SDGs can be defined as $G = (V, E, \phi, \psi)$ as defined by [48], where V and E represent the graphs nodes and edges, respectively. The nodes represent system variables (or functions) and the edges causal dependency. A qualitative state $\psi : V \rightarrow \{+, 0, -\}$ is assigned to each node and each directed edge has either a positive or a negative sign $\phi : E \rightarrow \{+, -\}$, determining the direction of effect between the connected nodes. The direction of the edge represents the direction from cause to effect. SDGs can thus be viewed as causal models with a very simple rule-base, namely that positive edges preserve the state from cause to effect, while negative edges invert it, whereas the 0 state propagates to another 0 state in either case. Reinartz et al. [46] show that an MFM model can be converted to an SDG, as long as the causal relations between all functions are either strictly proportional or strictly inverse. Representing the underlying causal information in the form of SDG enables the utilization of graph analysis methods during model analysis. Apart from analysis, the graph representation can be used to simplify the model in order to focus on specific functions of interest in a diagnostic application, as the SDG representation is not bound by MFM modeling syntax that prohibits arbitrary function removal. Fig. 2 shows a simple example of conversion from MFM to SDG. Since *balance* function is assumed as normal state, there are eight propagation rules between three *transport* functions that may be associated with process variables, which can be converted into the equivalent SDG model. The remainder of the paper will feature both MFM during the explanation of the modeling procedure as well as SDG in the alarm analysis section.

2.3. Identification of alarm root causes

During alarm flooding when multiple alarms occur, identifying the root cause(s) is about to find one or more minimal cause-consequence trees, which can use one or a combination of as few root causes as possible to explain as many observations as possible. To this end, combining analysis of causes as well as consequences of individual alarm is necessary [35]. Performing multiple alarm analyses on the same function model, it is inevitable to encounter conflicts, e.g. distinct states being concluded for the identical function. This section presents how the reasoning conflicts are addressed and how all possible root cause(s) for an alarm flooding situation can be automatically identified.



(a) MFM (b) SDG

Fig. 3. Converting the MFM model of a watermill described in Lind [49] to SDG graph. Blue and red nodes mark low (-) and high (+) states, respectively. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

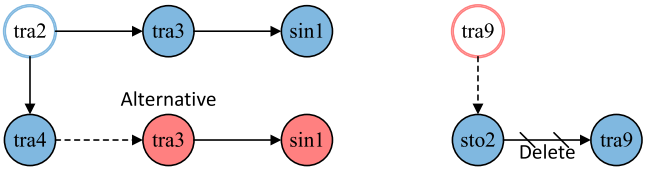
2.3.1. Propagation and conflict-resolution

A modeling example of watermill from Lind [49] is used to demonstrate how propagation and conflict are addressed in the alarm root-cause analysis. As shown in Fig. 3(a), the MFM model describes how the watermill can be divided into three functional levels. These functional structures are organized in terms of means-end. For instance, grinding of the grain is achieved by conversion of potential energy of water to kinetic energy, which in turn requires to transport and delivery water. As shown in Fig. 3(b), the MFM model is converted into SDG graph by considering the propagation rules that are introduced in Section 2.1. Converting from MFM to SDG is useful to visualize representations of causal dependencies between modeled functions. The general principles of the propagation rules are:

1. The relation *influencer* allows propagation in both directions, i.e. positive effect from upstream to downstream while negative effect in the opposite direction, e.g. *sou*1–*tra*1.
2. The relation *participant* only allows unidirectional propagation from *transport* to non-*transport*, with either positive or negative effect depending on whether the influence source is from upstream or downstream, e.g. *tra*1–*sto*1, and *sto*1–*tra*2.
3. In relation to the function *balance*, assume that all *transports* are connected to *balance* with *influencer*, *transports* in the same side of *balance* have negative effect, e.g. *tra*3–*tra*4, while *transports* in the same side of *balance* have positive effect, e.g. *tra*9–*tra*11.
4. Means-end relations only allow unidirectional propagation from means to end with only positive effect, e.g. *tra*6–*tra*2.

Propagating a state from one function to another typically requires a simple application of the inference rules between both functions. For a causal model like Fig. 3, a simple application of these reasoning rules can lead to two different kinds of reasoning conflicts, i.e. contradictions on parallel reasoning paths and contradictions in negative loops [50]. Handling such reasoning conflicts is one of the main challenges in the application of causal process models, because there exists no generic solution to resolve the conflicts. Instead, the reasoning engine must be tailored for the specific modeling methodology it is applied to, taking into account whether the causal model encodes precedence of certain relations over others or not. The case of Fig. 3 has both reasoning conflicts, which have been extracted to illustrate how the propagation in conflicts is addressed, as shown in Fig. 4.

Contradicting parallel paths. As shown in Fig. 4(a), the propagation is initiated by setting node *tra*2 to a low state, it can be observed that nodes *tra*3 and *sin*1 can assume either negative or positive states, depending on whether the propagation follows the path "*tra*2-*tra*3-*sin*1" or "*tra*2-*tra*4-*tra*3-*sin*1".



(a) Contradicting parallel paths (b) Negative loops

Fig. 4. Propagation results of reasoning conflicts. Using color-framed nodes to indicate observed states and using color-filled nodes to indicate propagated states. Blue and red nodes mark low (-) and high (+) states, respectively. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

In MFM, causal connections between variables represent potential cause–effect relations that can, but must not necessarily take effect. This means that the employed reasoning system must consider conflicting parallel paths like in Fig. 4(a) as equally likely alternatives until specific evidence confirms one or the other.

Negative loop. As shown in Fig. 4(b), the example is set for node *tra*9 in a high state. It becomes apparent that a propagation through the loop would lead to a contradiction of the state of node *tra*9, because the propagation "*tra*9-*sto*2-*tra*9" results in a low state for *tra*9, while its observed state is a high state.

In the case of loops, positive and negative loop have to be treated distinctly. Positive loops do not require conflict resolution, because all evidences inside the loop are by definition consistent, whereas negative loop introduces a conflict at the loop initiation point. Since this introduces a conflict on a single propagation path, it cannot be considered as an alternative propagation, as it the path contradicts itself. In order to resolve conflicts in negative loops, the reasoning system assumes that the initial effect on the loop initiation point takes precedent, meaning that the propagation in the loop is only executed up to the point where the conflict occurs.

The extended causal reasoning algorithm that results in a tree structure of reasoning results for propagation from a single root-node is outlined in the flow-chart in Fig. 5. It should further be noticed that it is necessary to track all node-state pairs during propagation in order to avoid infinite propagation in case of positive loops and merging non-conflicting parallel paths.

2.3.2. Automated root-cause identification

By using the above algorithm to handle reasoning conflicts, the current MFM-based causal reasoning system is able to automatically

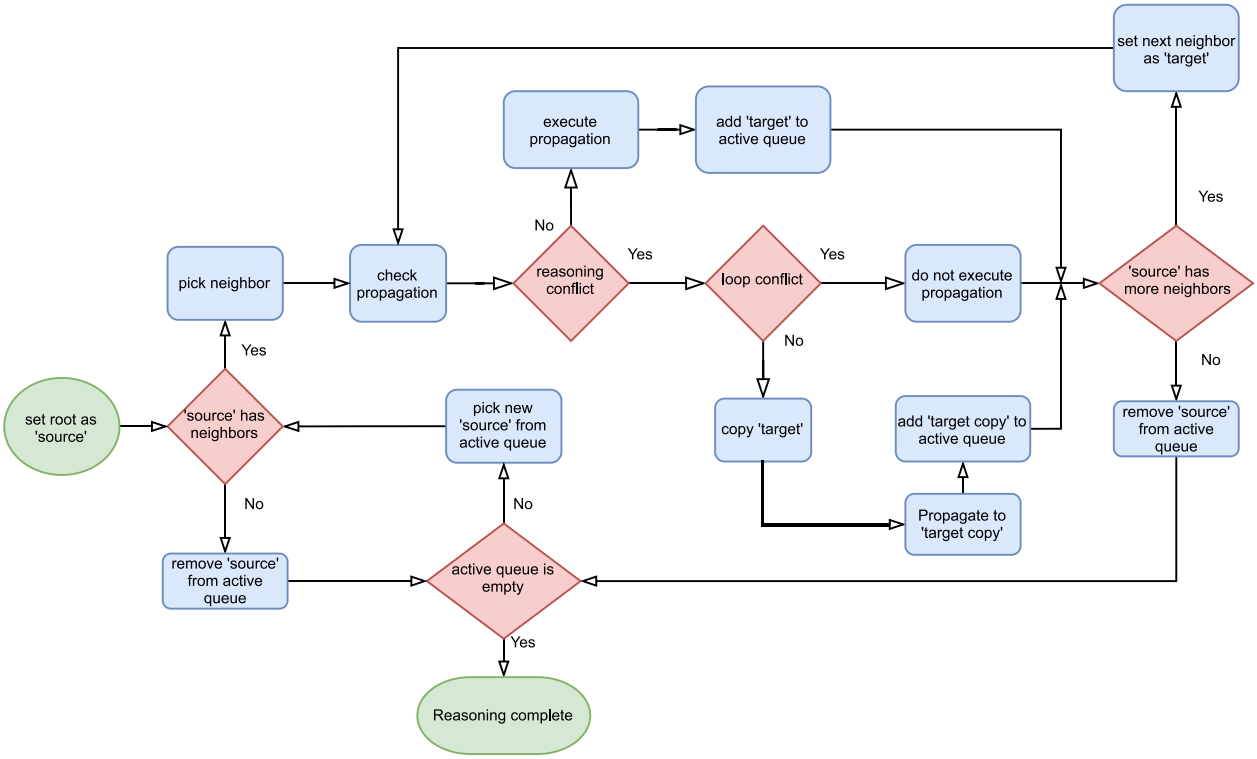


Fig. 5. Causal reasoning algorithm that extends the basic rule-based propagation by accounting for potential reasoning conflicts.

identify the possible root cause that can explain all occurring alarms. Unlike the Larsson Algorithm [27] which implies that the root cause can only be one of alarms as the root-cause alarm, it is assumed that the root cause can be either an observation or a function state that is not associated to any process variable. In other words, the proposed causal reasoning algorithm is capable to identify the root cause as either an alarm event or a specific fault that may not be set with alarms. Fig. 6 shows the analysis result for the case shown in Fig. 3. Among all the cause or consequence nodes deducted from two alarms, only the tree oriented from node *tra*11 in positive state can explain both alarms simultaneously. However, given an alarm flooding situation when a large amount number of alarms are generated, the algorithm usually generates more than one root cause in order to account all occurred alarms even though the true root cause may be unique. Therefore, it is necessary to determine the true root cause from all the identified candidates.

2.4. Root-cause ranking with reward function

A reward function is applied to the trees that result from the application of the propagation algorithm for different alarm root causes in order to determine the root cause that describes the current set of alarms best. Loosely defined, a probable root cause will, based on the causal model, explain many evidences, i.e. alarms while contradiction few or none. More precisely, the reward function needs to calculate a quantitative value based on a given causal process model and a set of evidences. Trying to emulate a human operators approach to causal root-cause analysis, the reward function is based on three factors that are typical to human reasoning about potential plant scenarios. The human reasoning approach about root causes involves answering the following questions:

Rule 1. How many, and which of the evidences that are observed can be explained by the given root cause?

Rule 2. How many, and which of the evidences that are observed cannot be explained by the given root cause?

Rule 3. How many, and which evidences would be expected to appear but not be observed, if the given root-cause was present?

The ranking algorithm extracts this information from the propagation trees by traversing all paths from the root-node to the various leaf-nodes of the tree. Each path is only traversed until the leaf is reached or until a node that contradicts an evidence is encountered. The procedure is explained in Fig. 7 for the evaluation of a hypothetical root cause "α (-)". The propagation tree supports the alarm evidences "α (-), β (-), γ (+), ϵ (-) and θ (-)", namely it results in five evidences that can potentially be explained by the chosen root cause candidate. Of the remaining sensors in the model, while the sensors "δ, κ, ζ λ, η, ι" would be expected to show responses. The evidences "ϵ, θ" are also expected to show the states that contradict the evidences based on the causal model, thus are omitted.

The following analysis section shows results for two reward functions are proposed to assess the potential alarm root causes. The first evaluates the scenarios purely based on how many evidences can be explained by the specific root-cause candidates, i.e. Rule 1, which can be simply be written as,

$$R_1 = n_{explained}. \tag{1}$$

While the second penalizes the reward for additional expected evidences that are not supported by measurements, i.e. $n_{observed}$ (Rule 2). This reward function takes the form,

$$R_2 = \frac{n_{explained}^2}{n_{observed} \cdot n_{expected}}.(n_{observed} \neq 0) \tag{2}$$

Where $n_{expected}$ represents Rule 3, i.e. the number of evidences that would be expected to appear if the given root cause is presented. Note that $n_{expected}$ would never be zero in reality because each root cause is identified by backward reasoning from one of observed alarms

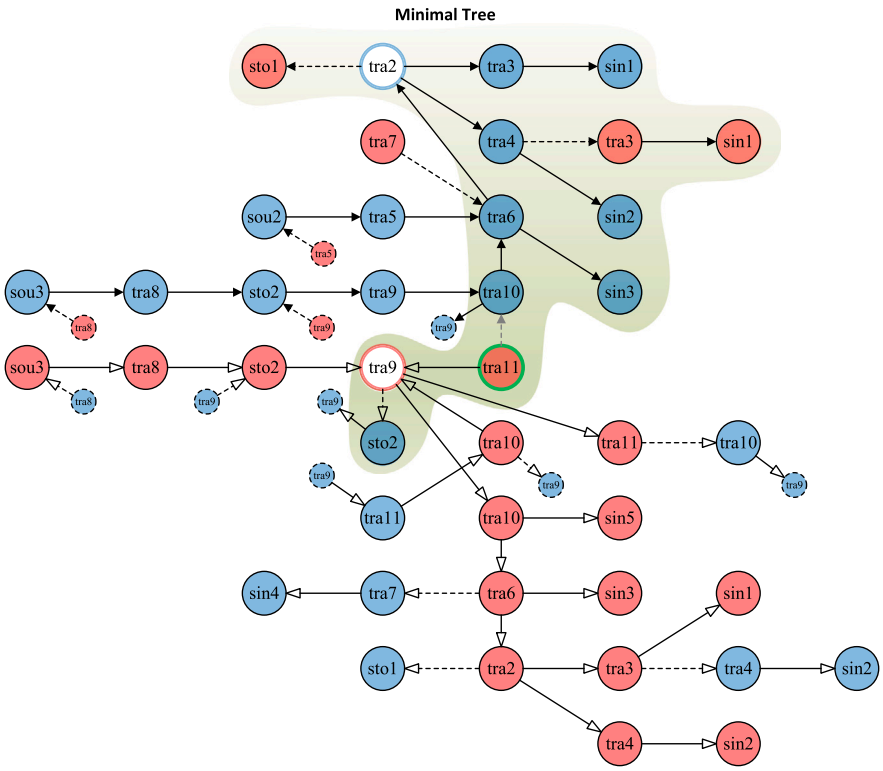


Fig. 6. Identifying the minimal cause-consequence tree and root cause that can explain two alarm observations, [*tra*2,low] and [*tra*9,high]. Using color-framed nodes to indicate observed states and using color-filled nodes to indicate propagated states. Blue and red nodes mark low (+) and high (+) states, respectively. Dotted nodes are removed by the reasoning system due to negative loops. Solid arrows indicate the causal reasoning for node *tra*2, and hollow arrows are for the causal reasoning for node *tra*9. Green-framed node (i.e. [*tra*11,high]) is the identified root cause. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

along the relations in the MFM model, there is at least one evidence is expected for this root cause, i.e. $n_{expected} \geqslant 1$. However, $n_{observed}$ could be zero in the rare situation that an identified root cause can explain all occurred alarms and there is no alarm that is expected to be triggered by this root cause but not observed. It will lead to $R_2 \rightarrow \infty$, which in fact makes sense because the high ranking rate indicates that the identified root cause is exactly the one resulting in the current alarm situation. For the mathematical reason, it is assumed that $n_{observed} \neq 0$. In the example shown in Fig. 7, it is shown that of the total six evidences could be explained by the hypothetical root cause "α (-)", but that a total of eleven variables were expected to show a deviation, resulting in $n_{explained} = 5$, $n_{observed} = 6$ and $n_{expected} = 11$. The corresponding reward values for this root cause are thus $R_1 = 5$ and $R_2 = 5 * 5/(6 * 11) = 0.38$.

2.5. Real-time alarm analysis

Combining root-cause identification with the reasoning algorithm and root-cause optimization with the ranking algorithm, it is able to provide a real-time alarm analysis. Fig. 8 demonstrates how the developed alarm analysis system can be applied in a real-time alarm flooding situation. Initially, root causes are identified for individual alarm. Note that there could be cases that multiple alarms would be traced back to the same root cause. All of identified root causes are evaluated against the consequence reasoning results for each root cause. The ranking algorithm can be used to provide the rate of root-cause candidates. As the alarm list changes in a dynamic operational scenario, different set of root causes will be found, and correspondingly a different ranking will be given. This could imply that an additional failure may have occurred if the fault leading to the alarm flood at the previous moment has not been eliminated.

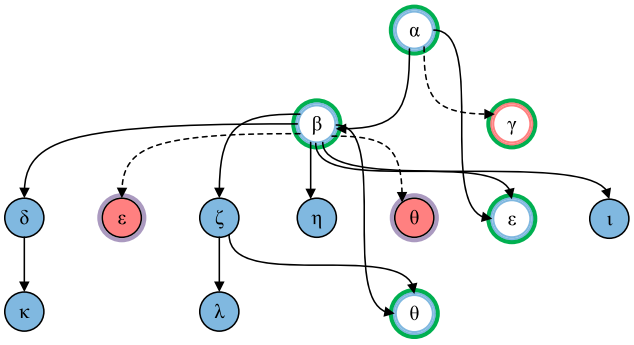


Fig. 7. Propagation result tree for a hypothetical root cause "α (-)". Using color-framed nodes to indicate observed states and using color-filled nodes to indicate propagated states. States that explain the alarm evidences are outlined in green. Contradictions to explained states are outlined in purple. Blue and red indicate (-) and (+), respectively. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

3. Functional modeling of PWR

The alarm analysis method described in Section 2 is applied to a Westinghouse-type PWR. This section presents how MFM is used for the functional modeling of PWR. The resulting MFM model will latter be used in the alarm root-cause analysis and ranking.

3.1. PWR

PWR is designed to include one primary cooling circuit and one secondary circuit. The primary system uses light water as coolant and

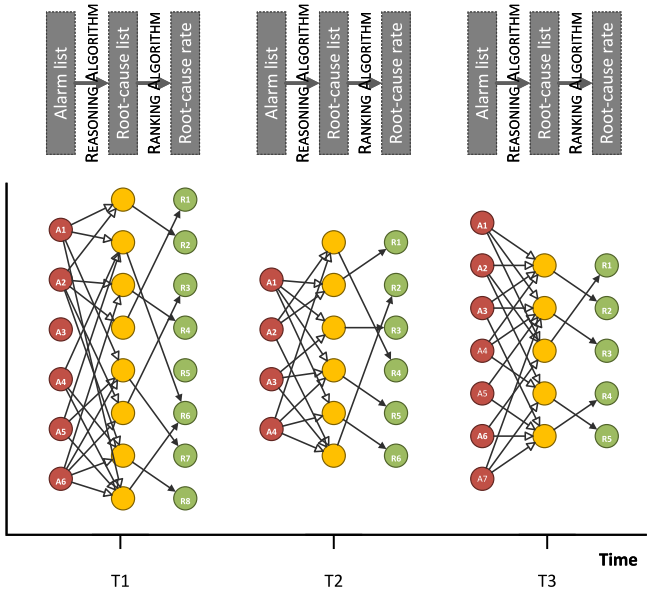


Fig. 8. Real-time alarm analysis.

moderator, transferring thermal energy generated by the reactor to the secondary system. The heat exchange between the primary coolant and the secondary coolant happens in the steam generators, where very high temperature coolant water at the primary side runs through internal tubes inside the steam generators, heating up the secondary coolant water through the tube walls to generate steam. The steam generated in the secondary system is then used to power the steam turbine to generate electricity.

In the primary system of PWR, i.e. reactor coolant system (RCS), a pressurizer (PRZ) is used to maintain the RCS pressure. Saturated steam kept inside the pressurizer maintains the RCS pressure at 2235 psig during normal operation. Heaters and water sprays can be used to increase or decrease the RCS pressure during normal operation. Further over-pressure protection is provided through a set of power operated relief valves (PORVs) that allow steam from the pressurizer to be released into a pressurizer relief tank (PRT). Another function of the pressurizer is to provide a surge volume for reactor coolant expansion and contraction due to density changes that result from varying coolant temperatures. The pressurizer level control is done by manipulating the charging flow into the RCS main loops. The charging and letdown in the RCS are also used to change the boron concentration in the RCS to achieve reactivity control. Additionally, PWR uses banks of neutron absorbing control rods to control the reactivity by automatically or manually inserting or withdrawing the control rods.

For the purpose of illustration, only RCS is considered for the functional modeling.

3.2. MFM modeling procedure

Building MFM models requires system knowledge acquisition and a subsequent conversion of that knowledge into MFM syntax [51]. The knowledge acquisition phase should result in a comprehensive description of the operation of the process, captured in two diagrams. The first is function-stream diagram (FSD), which provides and abstract representation of material, energy, and control information flows in system. The second is objective-function tree (OFT), which captures the relation between high-level system objectives and their low-level realization in the process. FSD helps to identify relevant subsystems and their relations and is subsequently used to determine the mass, energy and control flow structures in MFM that are required to describe that process. Further relations between flow structures based on the information about means-end dependencies expressed in OFT are added once the flow structures have been defined.

FSD is mainly constructed by reference of engineering drawings such as piping and instrumentation diagrams (P&ID) and process flow diagrams (PFDs). Converting from engineering drawings to FSD is relatively easy to perform. While constructing OFT is not straightforward, which needs to learn the operational targets or hazards, and their means-end relationships from diverse plant documentations such as system function descriptions, standard operating procedures (SOPs), and results from safety assessments such as hazard and operability studies (HAZOP) [51]. Fig. 9 shows the resulting OFT of RCS. The primary objectives of the modeled system are the production of thermal energy for power generation, and the leak of produced energy from the reactor to ensure safety. It is further imperative to maintain the integrity of the primary side to stop radioactive material from exiting the system. These primary objectives are achieved through a number of enabling objectives, which include maintaining coolant circulation so the thermal energy can be transported and removed, as well as providing enough coolant for level control in the pressurizer. Further, correct boron concentration and control rod positioning in the RCS loops to control the reactivity have to be ensured. Another enabling objective is to maintain the pressure in the system. As shown in Fig. 9, the enabling objectives are further decomposed until they can be represented by MFM functions.

3.3. MFM model of RCS

Following the above modeling procedure, the MFM model of PWR's RCS is built as shown in Fig. 10. The coolant material flow in the MFM model is represented by three different MFM flow structures. "Coolant level" represents the coolant stream from the charging towards the letdown, while "Coolant loop" summarizes the three main coolant circulations as one mass circulation. The flow structure "Pressurizer mass" represents the coolant in- and output to the pressurizer. The three mass flow structures are connected by means-end relations following Fig. 9, representing the fact that the charging provides conditions for coolant circulation and constant coolant inventory in the system. The main thermal energy flow from the reactor to the secondary system is represented by the energy flow structure "Energy production" in the MFM model. The energy transportation in this flow structure is realized by the means of coolant circulation. The other energy flow structure "Pressurizer energy" in the MFM model represents the pressure balances that is influenced by the mass volume, coolant temperature, and the system integrity. The remaining means-end relations between different flow structures in the model also follow the means-end dependencies in Fig. 9. The flow functions associated with the sensors where alarms can be captured are also highlighted in the model. Table 1 shows the description of these sensors including their operating range and alarm thresholds.

4. Case study

In this section, the developed method is used to perform the root-cause analysis for several emergency plant scenarios of PWR. Data for the operating scenarios are obtained by using a full-scope PWR simulator integrated at HAMMLAB. All presented scenarios were supervised by an experienced control room operator at HAMMLAB who issued process commands to accommodate the current plant state and production objectives.

4.1. Scenarios

Three scenarios are investigated for the case study of alarm root-cause analysis. All scenarios are initiated at the nominal plant operating condition, which is very stable during fault-free operation. The process is stable to the degree that all visually observable signal deviations would be considered as abnormal plant behavior by a control room operator.

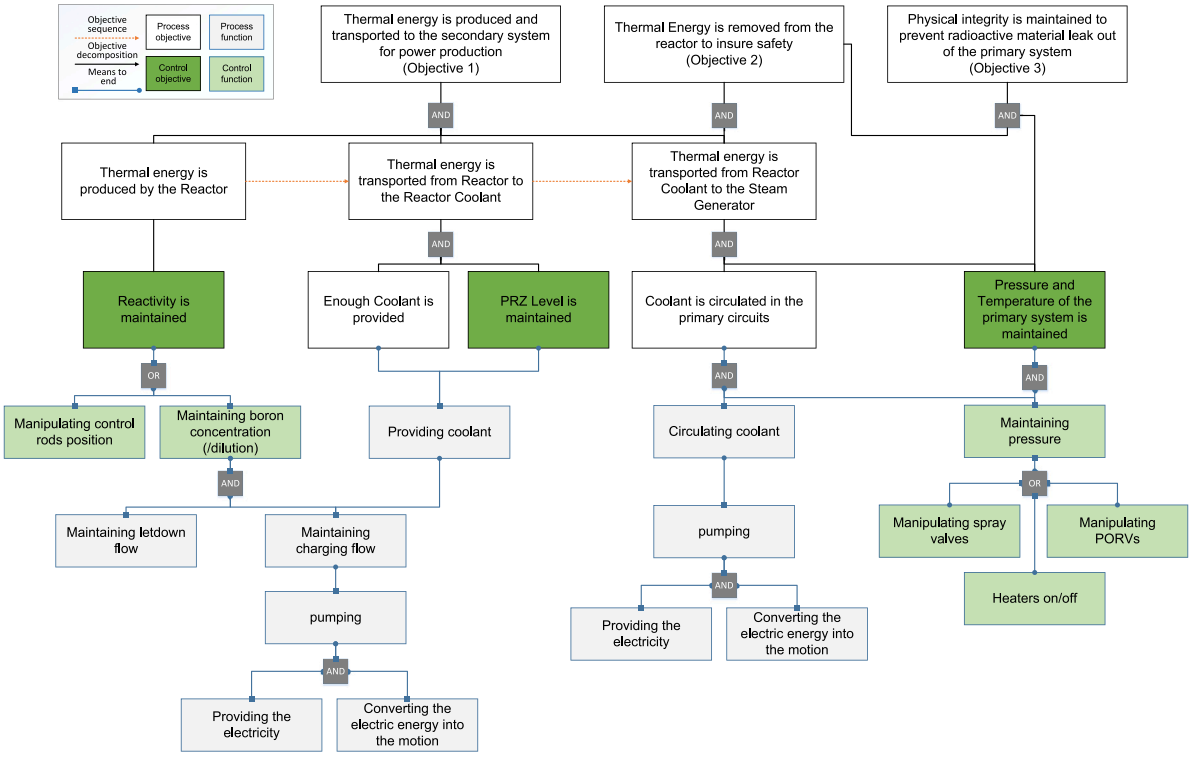


Fig. 9. OFT of RCS.

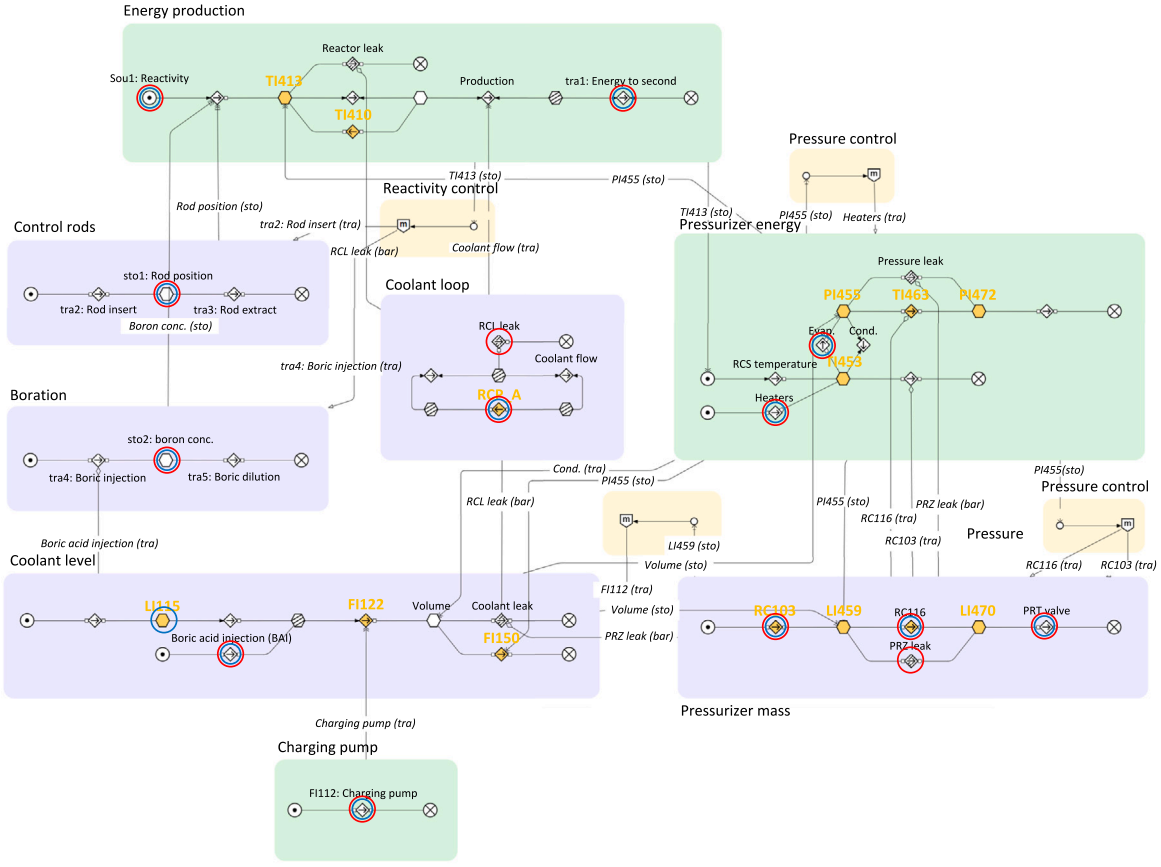


Fig. 10. MFM model of RCS of PWR. Flow functions with yellow fill indicate their associations with sensors. Functions marked with red or blue circles indicate the potential root causes that can be identified by the MFM-based alarm analysis system (see Table 2). Red and blue indicate high (or breach specific to the barrier function) and low state, respectively. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 1
Description of the sensors included in the process model, including their operating range and alarm thresholds.

Sensor Tag	Description	Unit	min	max	Nominal	Low	High
TI413	Reactor coolant system - Hot leg temperature	Deg F	0	700	619.98	350	650
TI410	Reactor coolant system - Cold leg temperature	Deg F	0	700	557.61	300	620
LI115	Volume control tank (VCT) level	%	0	100	30.74	22	34
FI122	Charging pump flow	gpm	0	150	79.82	50	110
FI150	Letdown flow	gpm	0	200	105.40	90	150
RC116	PORV opening percentage	%	0	100	0	–	–
RC103	Spray valve opening percentage	%	0	100	0	–	–
LI459	Pressurizer level	%	0	100	60.00	50	70
LI470	Pressurizer relief tank level	%	0	100	70	–	–
PI455	Pressurizer pressure (channel 1)	psig	1700	2500	–	2050	2250
PI472	Pressurizer relief tank pressure	psig	0	120	2.49	1.00	40.00
TI463	Pressurizer relief valve temperature	Deg F	50	400	112.96	90	250
TI453	Pressurizer temperature (waterspace)	Deg F	100.00	700	652.28	400	680
RCP_A	Reactor coolant pump (A)	%	0	120	100	90.5	–

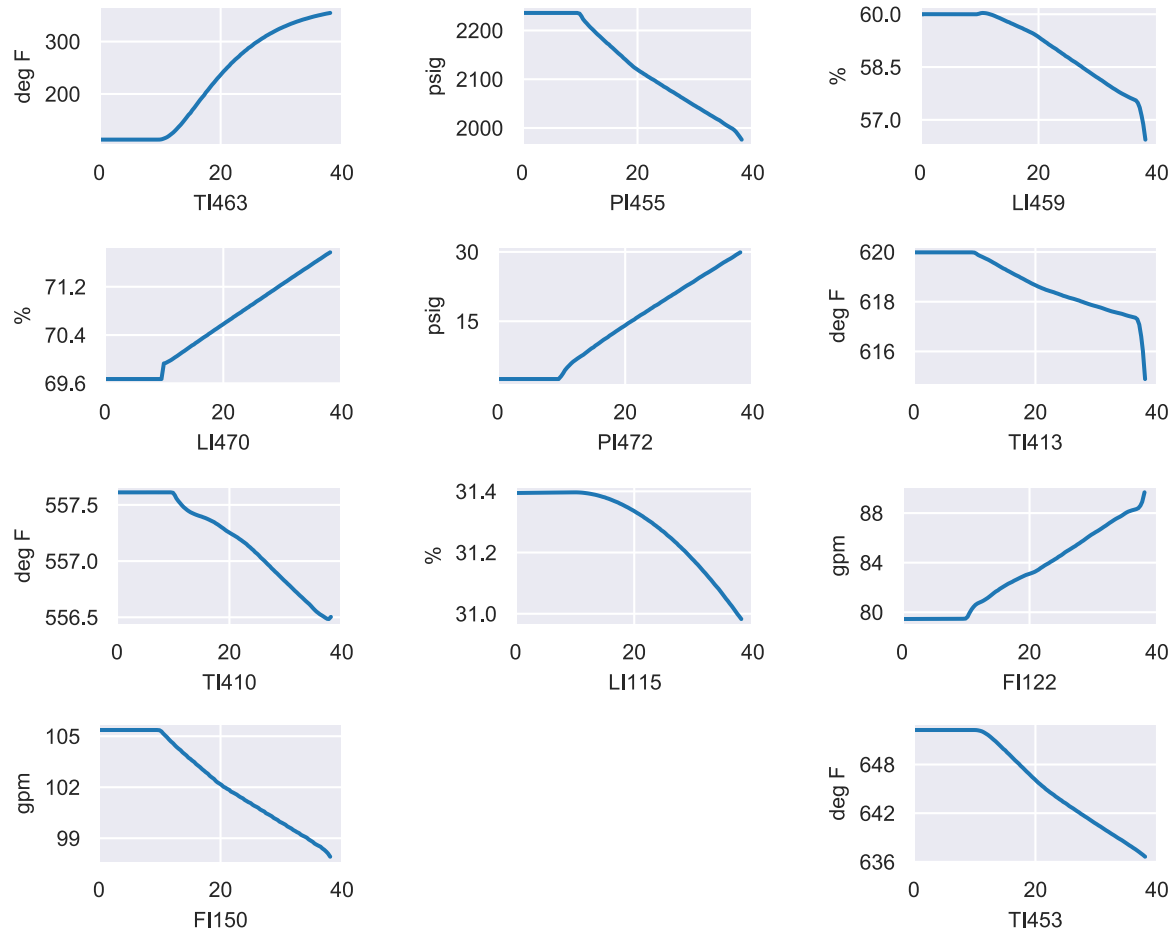


Fig. 11. Plant reaction to a leak in the pressurizer at t = 9.16 s, leading up to a reactor trip. Time in seconds on all x-axes, units of the featured sensors on the *y*-axis.

4.1.1. Pressurizer relief valve leak

The first scenario is regarding pressurizer relief valve leak, which is initiated by introducing a failure, i.e. failed open to the pressurizer relief valve on top of the pressurizer. Fig. 11 shows the plant reaction, i.e. trends of process variables after initiating the scenario. A direct result of the introduced failure is the removal of steam from the pressurizer, which further affects the state of the pressurizer on the one hand and the state of the pressurizer relief tank on the other hand.

Effect on the pressurizer. The leak of steam from the pressurizer leads to a reduction of pressure and temperature in the pressurizer, which subsequently results in a drop of the pressurizer level. As the pressurizer level goes down, the charging system automatically increases charging flow to put more water into the RCS, resulting in the decrease of the charging system Volume Control Tank (VCT) level as a secondary effect. Increased charging further results in a reduction of the pressurizer water space temperature, as more water from the cold leg is fed into the pressurizer. The decrease of the RCS pressure also results in a reduction of the pressure-driven letdown flow (water that exits the system from the RCS cold leg).

Effect on the pressurizer relief tank. A primary indication of the relief valve failure is the pressurizer relief temperature, which increases as a result of the valve failure. The increase is due to the increased throughput of hot steam through the relief valve that causes the local temperature to rise. The increased throughput further results in a increase of the pressurizer relief tank level and pressure, both of which

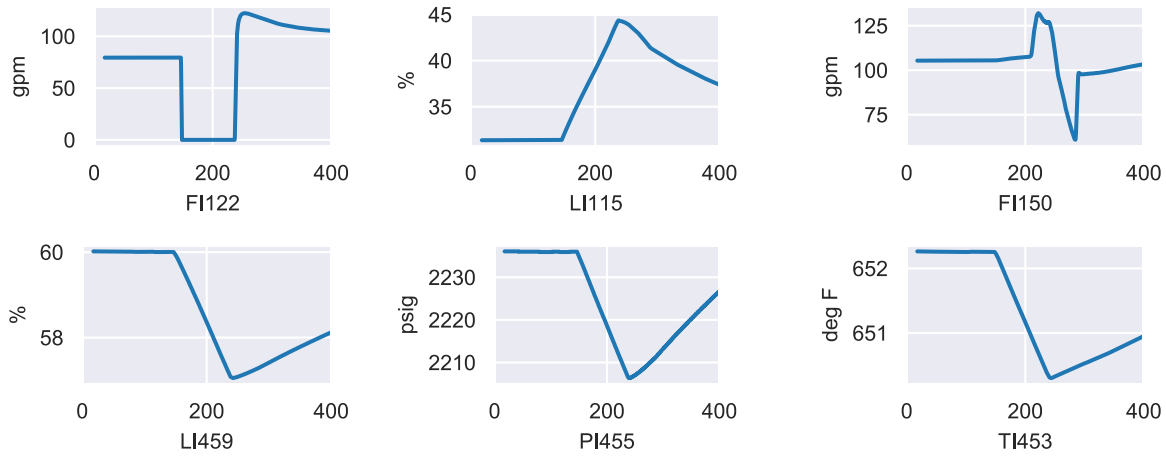


Fig. 12. Plant reaction to a loss of the primary charging pump at t = 145.92 s and subsequent recovery after the activation of the back-up charging pump at 233.12 s. The reaction of FI150 is considered as a deviation which initially increases. Time in seconds on all x-axes, units of the featured sensors on the y-axis.

indicate a leak through the relief valve rather than a leak at another location in the pressurizer.

4.1.2. Loss of the primary charging pump

The second scenario describes the effect of a complete loss of the primary charging pump of the RCS, which can be caused by either a mechanical pump failure or a loss of power to the pump. A summary of the trends of affected process variables is shown in Fig. 12.

The main indicator of the pump breakdown is the complete loss of charging flow (FI122). This loss of flow has immediate effects on the volume control tank and pressurizer levels. The VCT level increases, because letdown is still fed into the VCT, while the charging flow that usually exits the VCT is lost. The pressurizer level is affected the opposite way; Letdown flow still exits the RCS, but no new charging flow enters, leading to a decrease of the pressurizer level. The steam bubble in the pressurizer extends as a result of the reduced level, which leads to a reduction of the pressurizer pressure that goes along with the decrease of the pressurizer level. The behavior exhibited by the letdown flow is explained by two opposing effects. Letdown cooling is achieved in part through heat exchange with the charging flow. A loss of charging flow thus results in a temperature increase of the letdown, leading to a pressure increase and a resulting higher flow, which explains the initial increase in letdown after the pump failure. Eventually, the water in the letdown will flash to steam as a result of the loss of cooling, resulting in a loss of letdown flow, which explains the reduction of letdown flow after the initial increase. When charging flow is reinstated after activation of the back-up pump, letdown cooling is achieved through the charging flow once more, allowing a recovery of the nominal operating condition.

4.1.3. Boric acid injection

The third scenario, which is summarized in Fig. 13, shows the reaction of the plant to an increased injection of boric acid into the system. Borated acid absorbs neutrons in the reactor core and thereby reduces reactivity. The scenario can be separated into two different effects.

Changed charging flow source. The first effect, the increase of the level in the VCT, can be attributed to the fact that the charging flow, which would normally be taken from the VCT, is now largely replaced by a flow of water with a high boron concentration from a separate tank. Because letdown flow still enters the VCT, but outflow is reduced, the VCT level increases. This effect is a direct consequence of the deliberate increase of boric acid that was used to generate this scenario in the simulation. All other observed effects are a result of the reduced reactivity that results from the insertion of boric acid. In Fig. 13 it is shown that the increase of the VCT level is the only indication of abnormal behavior for multiple minutes, until the effects of the reduced reactivity become apparent.

Reduced reactivity. Reduced reactivity directly results in decreased power production. Initial indicators of decreased power production are the reduced cold-leg (TI410) and hot-leg (TI413) temperatures in the system. The general temperature decrease further leads to a reduction of the pressurizer level, which has two effects. On the one hand it results in a reduction of the pressurizer pressure due to a contraction of the water and subsequent expansion of the steam bubble in the pressurizer. On the other hand, the automatic control system tries to compensate for the loss of pressurizer level by increasing the charging flow. The reduction of the pressurizer pressure further leads to a reduction of the dependent letdown flow.

4.2. Analysis

The presented analysis comprises root cause analysis for the described fault-introducing scenarios based on the method introduced in Section 2 and the MFM model described in Section 3. The general assumption is made that the observed abnormal events can be traced back to a singular root cause and that the observed symptoms are direct or indirect consequences of the actual fault. The reasoning at the core of the analysis can be separated into three distinct steps: evidence generation, root-cause identification and root-cause ranking.

4.2.1. Evidence generation

The introduced functional models describe the qualitative causal dependencies between process variables. The qualitative nature of the models means that the described propagation can also only handle the qualitative states, which requires the conversion of quantitative process data to qualitative evidence data that contains information about whether the process variable is lower than expected, normal, or higher than expected. Common approaches to achieve this conversion involve the use of alarm thresholds. For the purpose of illustration, the plant reactions shown in Figs. 11–13, i.e. increasing or decreasing trends in a certain period of time are treated as high alarms or low alarms even though they may not necessarily reach the alarm trippoints listed in Table 1. The generated alarm evidences will be used as inputs of the MFM-based reasoning system.

4.2.2. Root-cause identification

By applying the alarm analysis system developed based on the method in Section 2, a scenario-free root-cause identification can be first performed on the built MFM model of PWR, which is a causal

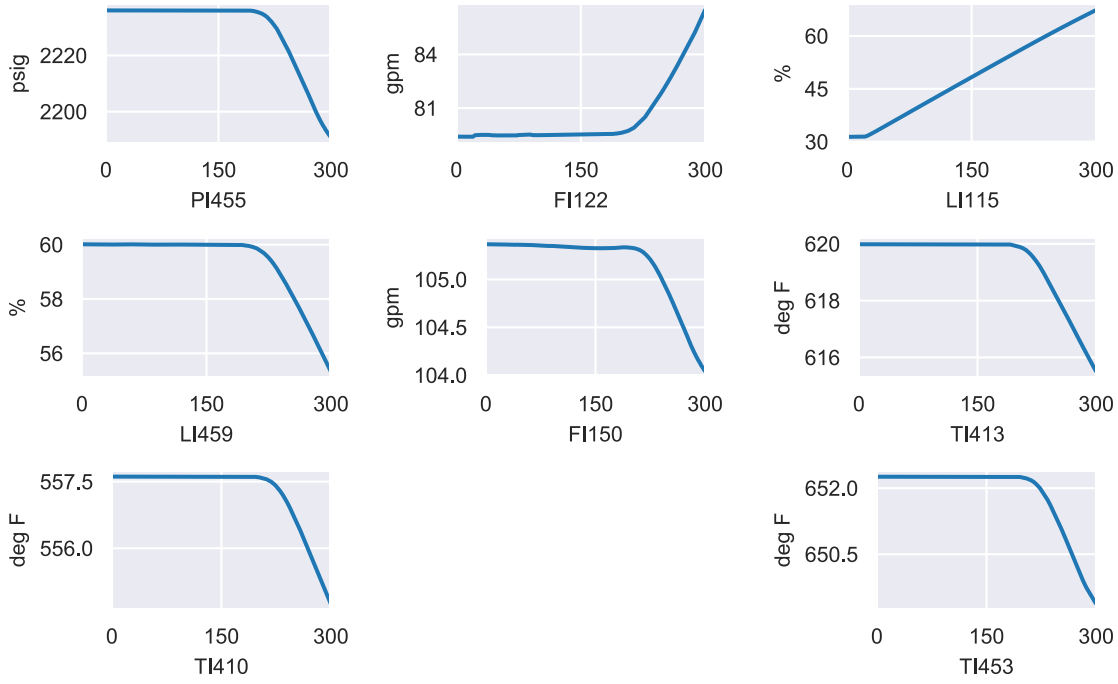
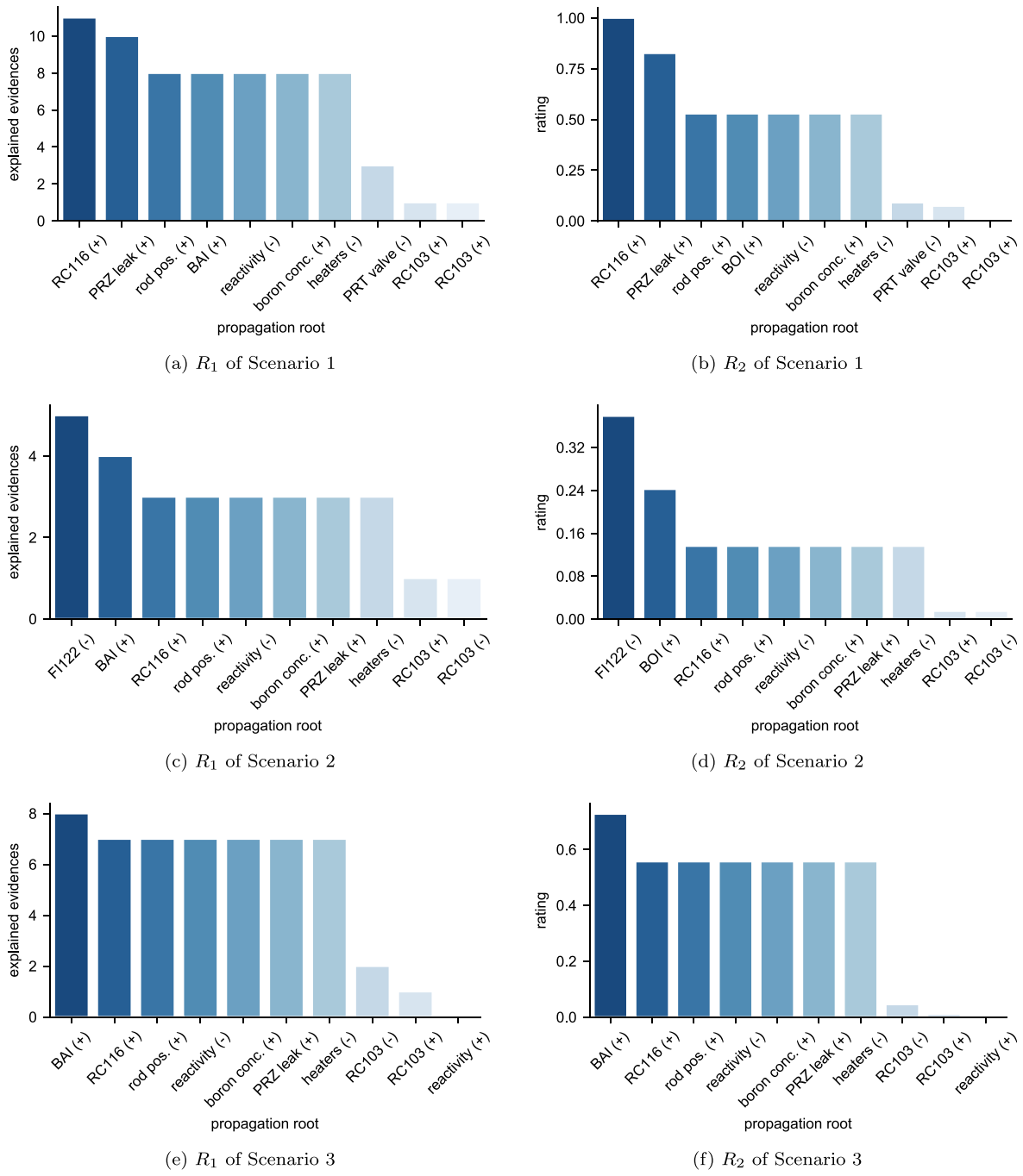


Fig. 13. Plant reaction to an increased concentration of borated acid in the reactor coolant. Time in seconds on all x-axes, units of the featured sensors on the *y*-axis.

Table 2
List of all potential root causes identified by the MFM-based alarm analysis system and their corresponding function label in the MFM model presented in Fig. 10.

No.	Root-cause function		Description	Root-cause state	
1	Reactivity	(*sou*1)	Changed reactivity level	High output	(+)
2				Low output	(−)
3	Rod position	(*sto*1)	Incorrect control rod position	High volume	(+)
4				Low volume	(−)
5	Boron conc.	(*sto*2)	Deviation of boron concentration in the RCS	High volume	(+)
6				Low volume	(−)
7	Energy to second	(*tra*1)	Reduced energy transport to secondary system	High flow	(+)
8				Low flow	(−)
9	RCL leak	(*bar*1)	Leak in the reactor coolant loops	Breach	(+)
10	RCP A	(*tra*_RCPA)	Reduced capacity/loss of reactor coolant pump	High flow	(+)
11				Low flow	(−)
12	RC116	(*tra*_RC116)	PORV malfunction	High flow	(+)
13				Low flow	(−)
14	PRZ leak	(*bar*2)	Leak in the pressurizer	Breach	(+)
15	RC103	(*tra*_RC103)	Spray valve malfunction	High flow	(+)
16				Low flow	(−)
17	Heaters	(*tra*_heaters)	Heater malfunction	High flow	(+)
18				Low flow	(−)
19	PRT valve	(*tra*_PRT)	Pressurizer relief tank valve malfunction	High flow	(+)
20				Low flow	(−)
21	FI122	(*tra*_FI112)	Reduced capacity/loss of charging pump	High flow	(+)
22				Low flow	(−)
23	LI115	(*sto*_LI115)	Low volume control tank level	Low volume	(−)
24	Boric acid injection	(*tra*_BAI)	Pressurizer evaporation anomaly	High flow	(+)
25				Low flow	(−)

reasoning process assuming that each of the process variables listed in Table 1 will produce either high or low alarm in a single system running. All possible root causes that lead to each corresponding alarm can be obtained from the reasoning outputs, which are shown in Table A.1 in Appendix A. Note that there are many cases that the identical root cause of an alarm appears several time. This indicates that there are different propagation routes for the root-cause that can lead to the same alarm eventually. When multiple alarms occur, all root causes that can be identified for the situation are the sum of the corresponding root-cause columns. To be summarized, there are 25 in total categories of potential root-cause candidates that may result in any alarm situation occurred in the plant, which are listed in Table 2. To be simplified, the root-cause of TRA2: ROD INSERT(HIGH/LOW) and the root-cause of TRA3: ROD EXTRACT(LOW/HIGH) are simplified as STO1: ROD POSITION(HIGH/LOW), and the root-cause of TRA4: BORIC INJECTION(HIGH/LOW) and the root-cause of TRA5: BORIC DILLUTION(LOW/HIGH) are simplified as STO2: BORON CONC.(HIGH/LOW). The actual root cause in any specific scenario must be included in the list. A root cause typically corresponds directly to a component malfunction, such as a valve leakage, or to a vital process abnormality, such as reduced reactivity. For the conducted case study, the true root cause is the failure introduced for each scenario. Therefore, the purpose of the latter ranking work is to validate whether those failures



(a) R_1 of Scenario 1

(b) R_2 of Scenario 1

(c) R_1 of Scenario 2

(d) R_2 of Scenario 2

(e) R_1 of Scenario 3

(f) R_2 of Scenario 3

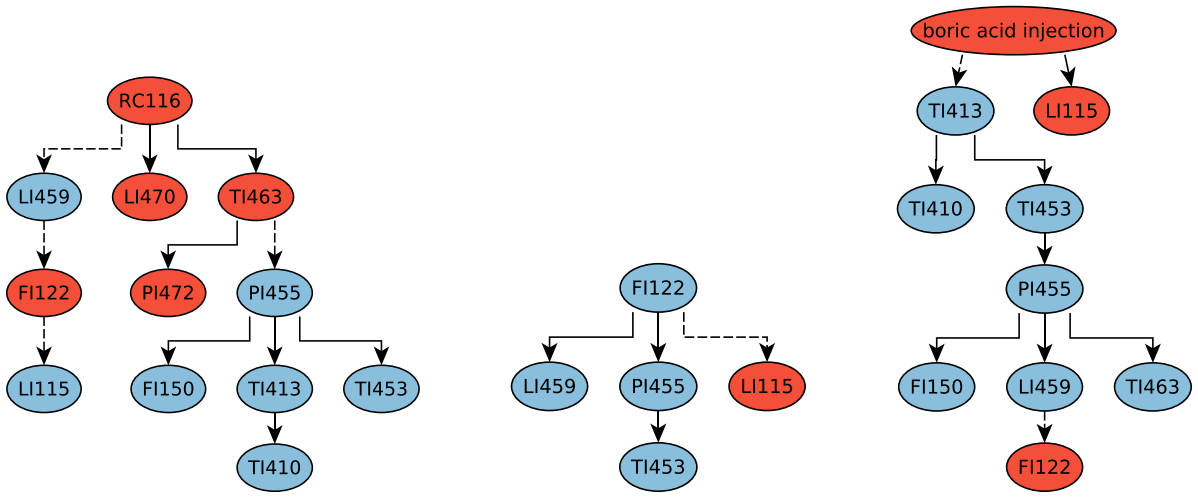
Fig. 14. Root-cause ranking for the three operating scenarios.

introduced to the simulator are the root causes that produce the alarm situations.

4.2.3. Root-cause ranking

Root causes for three operating scenarios presented in Section 4.1 are evaluated by applying the two reward functions presented in Section 2.4. The results of the analysis and the causal trees that correspond to the most likely root-causes are shown in Fig. 14 and Fig. 15, respectively. For the purpose of illustration, Fig. 14 only shows the ranking rates of the first ten identified root causes in each scenario.

Scenario 1: Pressurizer relief valve leak. The results of applying the presented reward functions are shown in Fig. 14(a)–(b). It can be observed that both reward function produce similar results, but that the reward function that penalizes unexplained and not-observed evidences leads to a stronger separation of the two most likely root-causes from the rest. The actual root-cause, which is a leak in the pressurizer relief valve, is correctly identified by both reward functions, as the root-cause "RC116 (+)" signifies a higher than normal flow through the relief valve. The second most likely root-cause is identified as a leak in the pressurizer, which is logical since a leak through the relief valve produces the same symptoms as a pressurizer leak at a different location in the pressurizer. The distinguishing evidence that distinguishes the two root-causes in the reasoning system is the high level in the relief tank, which would be expected to drop with the pressurizer level unless the leak occurred through the relief valve into the relief tank. The other five root-cause candidates that are identified as somewhat likely show up because of the reduced system pressure and its dependent symptoms. In the propagation tree for the identified best



(a) Relief valve failure (b) Charging pump failure (c) Boric acid injection

Fig. 15. Causal tree that corresponds to the most likely root-cause for each of the presented operating scenarios, according to the automated reasoning system.

root-cause scenario "RC116 (+)" in Fig. 15(a), it can be seen that the reduction of the pressurizer pressure "PI455" results in reactions from four further process variables. It follows that any root-cause candidate that influences the pressure can support at least five of the observed evidences.

Scenario 2: Primary charging pump failure. The results of the presented reward functions are shown in Fig. 14(c)–(d). It can be seen that the actual root cause "FI122 (-)" is identified as the most likely root-cause and that it is identified as the only truly likely root-cause by the second rating function. However, it can also be seen that only five of the six evidences can be explained by the causality represented in the functional model. Specifically, the reaction of the letdown flow is not expected, because neither the cooling effect of the charging flow on the letdown flow nor the possibility of letdown evaporation is modeled in the utilized causal model. Both effects are outside the current modeling scope and can thus not be predicted correctly, resulting in a decreased performance of the root-cause analysis for this case.

Scenario 3: Boric acid injection. The results of the application of the presented reward functions are shown in Fig. 14(e)–(f). It can be observed that the boric acid injection (BAI) is identified as the main root-cause, because it is the only cause that explains both the general reduction of energy in the system as well as the increase of the volume control tank level. It is further noticeable that six other root-causes explain a large amount of the observed evidences. This is explained by the fact that a general reduction of energy in the system can be caused by multiple factors and will produce very similar fault-signatures in most cases. Further, the root-causes "rod pos. (+), boron conc. (+) and reactivity (-)" can all be summarized as the same root-cause, since all of them have a reduction of reactivity as a direct consequence, making their separate mention in the analysis somewhat redundant.

5. Discussion

It has been shown that the MFM-based alarm analysis identifies the correct root-cause for all investigated scenarios with a certain degree of alarm flood. It is further observed that the introduction of the number of expected evidences, i.e. $n_{expected}$ as a parameter of the reward function seems to separate the actual root-cause from other candidates more distinctly. While this is promising, a general consideration of the potential bias introduced by this inclusion must be evaluated in further case studies in order to ensure the soundness of the applied function. Combining the analysis presented in Fig. 14 and Fig. 15, it is further observable that the distinction of the correct root-cause from other likely candidates is achieved through the correct identification of the primary causal dependencies. This reflects the real situation in the control room, where operators are often quickly able to connect clusters of specific symptoms and then proceed to investigate the true cause for the occurrence of this cluster of evidences. This aspect of the analysis becomes apparent when comparing the scenarios featuring the leak in the PORV and the boric acid injection. Both exhibit similar symptoms that point at a general loss of pressure and energy in the system, but a clear separation is possible through the identification, that only a leak in the PORV can result in effects on the PRT, while only boric acid injection could lead to an increase in the VCT level in combination with the remainder of the symptoms. It should be noted that this analysis is only possible due to the white-box nature of the MFM-based reasoning system. Comparing to data-driven alarm analysis methods, the obtained transparency from MFM allows a straightforward analysis of why the reasoning system assumes that a specific root-cause explains the current scenario, enabling human operators to evaluate the performance and trustworthiness of the reasoning system and to make suggestions about further improvements.

The present work can be seen in a research stream that leverages functional knowledge for the purpose of maintaining situation awareness of operators during alarm floods. Fig. 16 illustrates how the proposed method together with several MFM-based alarm processing techniques developed in history can contribute to situation awareness in different levels, and correspondingly what remaining tasks require operators to accomplish. The early attempt of using hierarchical information of MFM in design of human–machine interface including alarm display can facilitate operators' direct perception and analytical reasoning, which can increase situation awareness from the cognitive perspective, comparing to presenting alarm flood in poor interfaces [52]. However, operators still need to perceive the alarm flooding situation by themselves and perform the remaining tasks to respond to events. The more advanced alarm processing is the technique that is able to separate the root-cause alarm from many consequential alarms, i.e. alarm classification, for example the work of Larsson [27]. Dahlstrand [33] goes a step further to diagnose the actual fault that leads to alarm flood from the identified root-cause alarm. It is believed that provision of root-cause ranking or optimization by this study can further enhance the situation awareness on the basis of multiple root-cause candidates, because operators are only required to work on the most likely root cause(s).

The underlying assumption of alarm analysis is that a complex alarm situation is due to one or few root causes. However, in real-time situations like the one in Fig. 8, the developed alarm analysis system

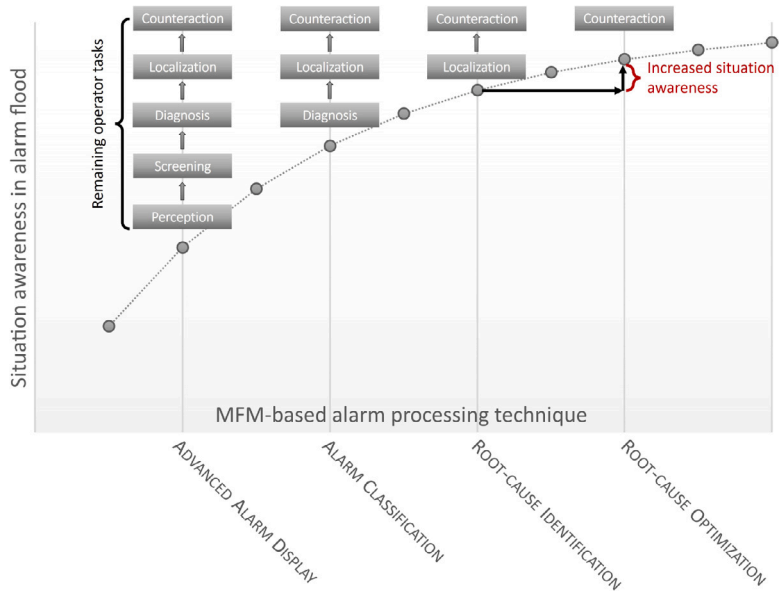


Fig. 16. Contributing to situation awareness in alarm flood by different MFM-based alarm processing techniques and remaining operator tasks given the corresponding technique is provided.

cannot recognize whether the alarm flood is resulted from a single fault or multiple faults occurred simultaneously, even though the most likely root cause can be distinguished from less possible root causes by applying the ranking algorithm. It is also necessary to differentiate root causes that have the same highest ranking rate, which makes difficult to determine the true fault.

6. Conclusions

The high-level nature and the ability of explaining causal dependencies allow MFM to quickly identify the alarm root causes, which is the key to apply in real-time alarm flooding situations. Instead of only separating initial alarms from consequential alarms, the current MFM-based causal reasoning algorithm is able to further diagnose the root-cause faults that may not be directly evidenced. In order to further improve the situation awareness of operators, given multiple root-cause candidates have been identified by MFM, this paper proposes a root-cause ranking algorithm, which can be used to determine the actual root cause. The present alarm analysis method is validated by a case study on PWR. Realistic emergency scenarios are simulated using a full-scale plant simulator at HAMMLAB. An MFM model of PWR is developed and utilized for automated reasoning about the process condition in each scenario, using optimization criteria for the reasoning system that mimic human operators decision criteria. The reasoning results show that the MFM-based alarm analysis is capable of identifying the correct root-cause in all tested scenarios. It is shown that the reasoning process on the MFM model is transparent and humanly interpretable, enabling operators and process experts to evaluate the reasoning performance based on their own process knowledge, which is one of the major requirements that enable this technology to be adopted in real control rooms of NPP. The analysis further reveals that the automated reasoning system for alarm analysis distinguishes the most-likely root-cause from other likely candidates by correctly identifying primary evidences that are directly causally dependent on the true root-cause, mimicking a human operator reasoning process. Future works include considering the importance of alarm evidences. It is necessary to investigate whether secondary-priority and tertiary-priority evidences should be valued equally to primary-priority evidences, as has been done in this study, or whether better diagnostic performance can be achieved by considering them distinctly. This issue is vertical for the root-cause identification and ranking method to be applied in the real-time alarm analysis.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This research was funded by the Danish Offshore Technology Centre, Denmark.

Appendix A. Root causes identified for each single alarm observation

See Table A.1

Table A.1
Root causes identified for each single alarm evidence observed for the sensors listed in Table 1.

Root	TI413				TI410				LI115				FI122			
Cause	High		Low		High		Low		High		Low		High		Low	
#1	reactivity	High	LI115	Low	LI115	Low	reactivity	High	reactivity	Low	reactivity	High	reactivity	High	LI115	Low
#2	tra3	High	reactivity	Low	reactivity	Low	tra3	High	reactivity	Low	reactivity	High	reactivity	High	reactivity	Low
#3	heaters	High	tra3	Low	tra3	Low	heaters	High	reactivity	Low	reactivity	High	reactivity	High	reactivity	Low
#4	heaters	Low	heaters	High	heaters	Low	tra5	High	tra3	Low	tra3	High	tra3	High	reactivity	Low
#5	tra5	High	heaters	Low	FI122	Low	FI122	High	tra3	Low	tra3	High	tra3	High	tra3	Low

(*continued on next page*)

Table A.1 (*continued*).

#6	FI122	High	FI122	Low	RC116	Low	RC116	High	tra3	Low	tra3	High	tra3	High	tra3	Low
#7	RC116	High	RC116	High	tra2	High	tra2	Low	heaters	Low	heaters	High	heaters	High	tra3	Low
#8	RC116	Low	RC116	Low	tra5	Low	tra4	Low	heaters	Low	heaters	High	heaters	High	heaters	Low
#9	tra2	Low	tra2	High	tra4	High	BAI	Low	heaters	Low	heaters	High	heaters	High	heaters	Low
#10	tra4	Low	tra5	Low	BAI	High	RC103	Low	FI122	Low	tra5	High	tra5	High	heaters	Low
#11	BAI	Low	tra4	High	bar1	Breach	RC103	Low	RC116	Low	tra5	High	tra5	High	RC116	Low
#12	RCP_A	Low	BAI	High	bar1	Breach	bar2	Breach	RC116	Low	tra5	High	tra5	High	RC116	Low
#13	RCP_A	Low	RCP_A	High	RC103	High			RC116	Low	FI122	High	FI122	High	RC116	Low
#14	tra1	Low	RCP_A	High	RC103	High			tra2	High	RC116	High	RC116	High	FI122	Low
#15	bar1	Breach	tra1	High					tra2	High	RC116	High	RC116	High	tra2	High
#16	RC103	High	bar1	Breach					tra2	High	RC116	High	RC116	High	tra2	High
#17	RC103	Low	bar1	Breach					tra5	Low	tra2	Low	tra2	Low	tra2	High
#18	RC103	Low	RC103	High					tra5	Low	tra2	Low	tra2	Low	tra5	Low
#19	bar2	Breach	RC103	High					tra5	Low	tra2	Low	tra2	Low	tra5	Low
#20			RC103	Low					tra4	High	tra4	Low	tra4	Low	tra5	Low
#21			bar2	Breach					tra4	High	tra4	Low	tra4	Low	tra4	High
#22			bar2	Breach					tra4	High	tra4	Low	tra4	Low	tra4	High
#23									BAI	High	BAI	Low	BAI	Low	tra4	High
#24									BAI	High	BAI	Low	BAI	Low	BAI	High
#25									BAI	High	BAI	Low	BAI	Low	BAI	High
#26									BAI	High	BAI	Low	RCP_A	Low	BAI	High
#27									RCP_A	High	RCP_A	Low	RCP_A	Low	RCP_A	High
#28									RCP_A	High	RCP_A	Low	RCP_A	Low	RCP_A	High
#29									RCP_A	High	RCP_A	Low	RCP_A	Low	RCP_A	High
#30									RCP_A	High	RCP_A	Low	RCP_A	Low	RCP_A	High
#31									RCP_A	High	RCP_A	Low	RCP_A	Low	RCP_A	High
#32									RCP_A	High	RCP_A	Low	tra1	Low	RCP_A	High
#33									tra1	High	tra1	Low	tra1	Low	tra1	High
#34									tra1	High	tra1	Low	tra1	Low	tra1	High
#35									tra1	High	tra1	Low	bar1	Breach	tra1	High
#36									RC103	High	bar1	Breach	bar1	Breach	RC103	High
#37									RC103	High	bar1	Breach	bar1	Breach	RC103	High
#38									RC103	High	bar1	Breach	bar1	Breach	RC103	High
#39									RC103	High	bar1	Breach	bar1	Breach	RC103	High
#40											bar1	Breach	bar1	Breach		
#41											bar1	Breach	RC103	Low		
#42											RC103	Low	RC103	Low		
#43											RC103	Low	RC103	Low		
#44											RC103	Low	RC103	Low		
#45											RC103	Low	bar2	Breach		
#46											bar2	Breach	bar2	Breach		
#47											bar2	Breach	bar2	Breach		
#48											bar2	Breach	bar2	Breach		
#49											bar2	Breach	bar2	Breach		
#50											bar2	Breach				

Root	FI150				RC116				RC103				LI459			
Cause	High		Low		High		Low		High		Low		High		Low	
#1	reactivity	High	reactivity	Low	RC116	High	RC116	Low	RC103	High	RC103	Low	reactivity	Low	reactivity	High
#2	tra3	High	tra3	Low									reactivity	Low	reactivity	High
#3	heaters	High	heaters	Low									reactivity	Low	reactivity	High
#4	tra5	High	tra2	High									tra3	Low	tra3	High
#5	tra2	Low	tra5	Low									tra3	Low	tra3	High
#6	tra4	Low	tra4	High									tra3	Low	tra3	High
#7	BAI	Low	BAI	High									heaters	Low	heaters	High
#8	RCP_A	Low	RCP_A	High									heaters	Low	heaters	High
#9	RCP_A	Low	RCP_A	High									heaters	Low	heaters	High
#10	tra1	Low	tra1	High									RC116	Low	tra5	High
#11	bar1	Breach	RC103	High									RC116	Low	tra5	High
#12	RC103	Low											RC116	Low	tra5	High
#13													tra2	High	RC116	High
#14													tra2	High	RC116	High
#15													tra2	High	RC116	High
#16													tra5	Low	tra2	Low
#17													tra5	Low	tra2	Low
#18													tra5	Low	tra2	Low
#19													tra4	High	tra4	Low
#20													tra4	High	tra4	Low
#21													tra4	High	tra4	Low
#22													BAI	High	BAI	Low
#23													BAI	High	BAI	Low
#24													BAI	High	BAI	Low
#25													RCP_A	High	RCP_A	Low
#26													RCP_A	High	RCP_A	Low
#27													RCP_A	High	RCP_A	Low

(*continued on next page*)

Table A.1 (*continued*).

#28													RCP_A	High	RCP_A	Low
#29													RCP_A	High	RCP_A	Low
#30													RCP_A	High	RCP_A	Low
#31													tra1	High	tra1	Low
#32													tra1	High	tra1	Low
#33													tra1	High	tra1	Low
#34													RC103	High	bar1	Breach
#35													RC103	High	bar1	Breach
#36													RC103	High	bar1	Breach
#37													RC103	High	bar1	Breach
#38															bar1	Breach
#39															bar1	Breach
#40															RC103	Low
#41															RC103	Low
#42															RC103	Low
#43															RC103	Low
#44															bar2	Breach
#45															bar2	Breach
#46															bar2	Breach
#47															bar2	Breach
#48															bar2	Breach
#49																
#50																

Root	LI470				PI455				PI472				TI463			
Cause	High		Low		High		Low		High		Low		High		Low	
#1	PRT valve	Low	PRT valve	High	reactivity	High	LI115	Low	RC116	High	RC116	Low	RC116	High	RC116	Low
#2	RC116	High	RC116	Low	reactivity	Low	reactivity	High	bar2	Breach						
#3	bar2	Breach			tra3	Low	reactivity	Low								
#4					tra3	High	tra3	Low								
#5					heaters	High	tra3	High								
#6					heaters	Low	heaters	High								
#7					tra5	High	heaters	Low								
#8					RC116	High	tra5	High								
#9					RC116	Low	RC116	High								
#10					tra2	High	RC116	Low								
#11					tra5	Low	tra2	High								
#12					tra4	High	tra5	Low								
#13					tra2	Low	tra4	High								
#14					tra4	Low	tra2	Low								
#15					BAI	High	tra4	Low								
#16					BAI	Low	BAI	High								
#17					RCP_A	High	BAI	Low								
#18					RCP_A	High	RCP_A	High								
#19					RCP_A	Low	RCP_A	High								
#20					RCP_A	Low	RCP_A	Low								
#21					tra1	High	RCP_A	Low								
#22					tra1	Low	tra1	High								
#23					bar1	Breach	tra1	Low								
#24					RC103	High	bar1	Breach								
#25					RC103	Low	bar1	Breach								
#26					RC103	Low	bar1	Breach								
#27					bar2	Breach	RC103	High								
#28							RC103	High								
#29							RC103	Low								
#30							bar2	Breach								
#31							bar2	Breach								
#32																
#33																
#34																
#35																
#36																
#37																
#38																
#39																
#40																
#41																
#42																
#43																
#44																
#45																
#46																
#47																
#48																
#49																
#50																

(*continued on next page*)

Table A.1 (*continued*).

Root	TI453				RCP_A			
Cause	High		Low	High		Low		
#1	reactivity	High	LI115	Low	RCP_A	High	RCP_A	Low
#2	tra3	High	reactivity	Low				
#3	heaters	High	tra3	Low				
#4	tra5	High	heaters	Low				
#5	FI122	High	FI122	High				
#6	RC116	High	RC116	Low				
#7	tra2	Low	tra2	High				
#8	tra4	Low	tra5	Low				
#9	BAI	Low	tra4	High				
#10	RCP_A	Low	BAI	High				
#11	RCP_A	Low	RCP_A	High				
#12	tra1	Low	RCP_A	High				
#13	bar1	Breach	tra1	High				
#14	RC103	Low	bar1	Breach				
#15	RC103	Low	bar1	Breach				
#16	bar2	Breach	RC103	High				
#17			RC103	High				
#18								
#19								
#20								
#21								
#22								
#23								
#24								
#25								
#26								
#27								
#28								
#29								
#30								
#31								
#32								
#33								
#34								
#35								
#36								
#37								
#38								
#39								
#40								
#41								
#42								
#43								
#44								
#45								
#46								
#47								
#48								
#49								
#50								

References

[1] EPRI, Alarm Processing Methods: Improving Alarm Management in Nuclear Power Plant Control Rooms, 1003662, Technical Report, Electric Power Research Institute, Palo Alto, CA, 2003.

[2] X. Wu, Z. Li, A review of alarm system design for advanced control rooms of nuclear power plants, Int. J. Hum.–Comput. Interaction 34 (6) (2018) 477–490, http://dx.doi.org/10.1080/10447318.2017.1371950.

[3] J.C. Laberge, P. Bullemer, M. Tolsma, D.V.C. Reising, Addressing alarm flood situations in the process industries through alarm summary display design and alarm response strategy, Int. J. Ind. Ergon. 44 (3) (2014) 395–406, http://dx.doi.org/10.1016/j.ergon.2013.11.008.

[4] J. Wang, F. Yang, T. Chen, S.L. Shah, An overview of industrial alarm systems: Main causes for alarm overloading, research status, and open problems, IEEE Trans. Autom. Sci. Eng. 13 (2) (2016) 1045–1061, http://dx.doi.org/10.1109/TASE.2015.2464234.

[5] J.E. Larsson, B. Öhman, A. Calzada, J. DeBor, New solutions for alarm problems, in: 5th International Topical Meeting on Nuclear Plant Instrumentation Controls, and Human Machine Interface Technology, NPIC and HMIT 2006, 2006, pp. 922–929.

[6] W. Brown, J. O'Hara, J. Higgins, Advanced alarm systems: Revision of guidance and its technical basis, NUREG/CR-6684, Technical Report, Brookhaven National Laboratory, Upton, NY, 2000.

[7] B. Öhman, Alarm analysis on large systems using multilevel flow models, IFAC Proc. Vol. 33 (28) (2000) 95–100, http://dx.doi.org/10.1016/s1474-6670(17)36816-7.

[8] K. Parsa, S. Member, M. Hassall, M. Naderpour, Process alarm modeling using graph theory : Alarm design review and rationalization, IEEE Syst. J. 15 (2) (2021) 2257–2268.

[9] Z. Simeu-Abazi, A. Lefebvre, J.P. Derain, A methodology of alarm filtering using dynamic fault tree, Reliab. Eng. Syst. Saf. 96 (2) (2011) 257–266, http://dx.doi.org/10.1016/j.ress.2010.09.005.

[10] L. Abele, M. Anic, T. Gutmann, J. Folmer, M. Kleinsteuber, B. Vogel-Heuser, Combining knowledge modeling and machine learning for alarm root cause analysis, IFAC Proc. Vol. (IFAC-PapersOnline) 46 (9) (2013) 1843–1848, http://dx.doi.org/10.3182/20130619-3-RU-3018.00057.

[11] Z. Cai, L. Zhang, J. Hu, Y. Yi, Y. Wang, Comprehensive alarm information processing technology with application in petrochemical plant, J. Loss Prev. Process Ind. 38 (2015) 101–113, http://dx.doi.org/10.1016/j.jlp.2015.08.010.

[12] J. Hu, Y. Yi, A two-level intelligent alarm management framework for process safety, Saf. Sci. 82 (2016) 432–444, http://dx.doi.org/10.1016/j.ssci.2015.10.005.

[13] F. Wen, C. Chang, Tabu search approach to alarm processing in power systems, IEE Proc., Gener. Transm. Distrib. 144 (1) (1997) 31–38, http://dx.doi.org/10.1049/ip-gtd:19970716.

[14] S. Lai, T. Chen, Methodology and application of pattern mining in multiple alarm flood sequences, IFAC-PapersOnLine 28 (8) (2015) 657–662, http://dx.doi.org/10.1016/j.ifacol.2015.09.043.

[15] M. Schleburg, L. Christiansen, N.F. Thornhill, A. Fay, A combined analysis of plant connectivity and alarm logs to reduce the number of alerts in an automation system, J. Process Control 23 (6) (2013) 839–851, http://dx.doi.org/10.1016/j.jprocont.2013.03.010.

[16] S.Y. Yim, H.G. Ananthakumar, L. Benabbas, A. Horch, R. Drath, N.F. Thornhill, Using process topology in plant-wide control loop performance assessment, Comput. Chem. Eng. 31 (2) (2006) 86–99, http://dx.doi.org/10.1016/j.compchemeng.2006.05.004.

[17] H. Jiang, R. Patwardhan, S.L. Shah, Root cause diagnosis of plant-wide oscillations using the concept of adjacency matrix, J. Process Control 19 (8) (2009) 1347–1354, http://dx.doi.org/10.1016/j.jprocont.2009.04.013.

[18] F. Yang, P. Duan, S.L. Shah, T. Chen, Capturing Connectivity and Causality in Complex Industrial Processes, in: SpringerBriefs in Applied Sciences and Technology, Springer International Publishing, Cham, 2014, http://dx.doi.org/10.1007/978-3-319-05380-6.

[19] R. Parvez, W. Hu, T. Chen, Real-time pattern matching and ranking for early prediction of industrial, Control Eng. Pract. 120 (61903345) (2022) 105004, http://dx.doi.org/10.1016/j.conengprac.2021.105004.

[20] J. Folmer, D. Pantförder, B. Vogel-Heuser, An analytical alarm flood reduction to reduce operator's workload, in: Human-Computer Interaction. Users and Applications, Springer, Berlin, Germany, 2012, pp. 297–306, http://dx.doi.org/10.1007/978-3-642-21619-0_38.

[21] Y. Meng, X. Song, D. Zhao, Q. Liu, Alarm management optimization in chemical installations based on adapted HAZOP reports, J. Loss Prev. Process Ind. 72 (January) (2021) 104578, http://dx.doi.org/10.1016/j.jlp.2021.104578.

[22] B. Zhou, W. Hu, T. Chen, Pattern extraction from industrial alarm flood sequences by a modified CloFAST algorithm, IEEE Trans. Ind. Inform. 18 (1) (2022) 288–296, http://dx.doi.org/10.1109/TII.2021.3071361.

[23] G. Dorgo, J. Abonyi, Sequence mining based alarm suppression, IEEE Access 6 (2018) 15365–15379, http://dx.doi.org/10.1109/ACCESS.2018.2797247.

[24] M. Bauer, N.F. Thornhill, A practical method for identifying the propagation path of plant-wide disturbances, J. Process Control 18 (7–8) (2008) 707–719, http://dx.doi.org/10.1016/j.jprocont.2007.11.007.

[25] EPRI, Model-Based Root Cause Analysis for Information Overload Management, 1012490, Technical Report, Electric Power Research Institute, Electric Power Research Institute, Palo Alto, CA, 2006.

[26] M. Lind, Multilevel Flow Modelling of Process Plant for Diagnosis and Control, Risø-M-2357, Technical Report, RisøNational Laboratory, Roskilde, Denmark, 1982.

[27] J. Larsson, Model-based alarm analysis using MFM, Annu. Rev. Autom. Program. 16 (Artificial Intelligence in Real-time Control) (1991) 121–126, http://dx.doi.org/10.1016/0066-4138(91)90020-C.

[28] J.E. Larsson, Diagnosis based on explicit means-end models, Artificial Intelligence 80 (1) (1996) 29–93, http://dx.doi.org/10.1016/0004-3702(94)00043-3.

[29] M. Gomez-Fernandez, K. Higley, A. Tokuhiro, K. Welter, W.K. Wong, H. Yang, Status of research and development of learning-based approaches in nuclear science and engineering: A review, Nucl. Eng. Des. 359 (August 2019) (2020) 110479, http://dx.doi.org/10.1016/j.nucengdes.2019.110479.

[30] J. Ouyang, M. Yang, H. Yoshikawa, Y. Zhou, J. Liu, Alarm analysis and supervisory control of PWR plant, in: Proceedings of Cognitive Systems Engineering in Process Control, CSEPC 2004, 2004, pp. 61–68.

[31] J.E. Larsson, B. Öhman, C. Nihlwing, H. Jokstad, L. Iren, J. Kvalem, M. Lind, Alarm reduction and root cause analysis for nuclear power plant control rooms, in: Proceedings Enlarged Halden Program Group Meeting, 2005, pp. 1–11.

[32] Ø. Berg, M. Kaarstad, J.E. Farbrot, C. Nihlwing, T. Karlsson, B. Torralba, Alarm systems, in: A.B. Skjerve, A. Bye (Eds.), Simulator-Based Human Factors Studies Across 25 Years - the History of the Halden Man-Machine Laboratory, Springer-Verlag, London, UK, 2011, pp. 155–168, http://dx.doi.org/10.1007/978-0-85729-003-8_10.

[33] F. Dahlstrand, Consequence analysis theory for alarm analysis, Knowl.-Based Syst. 15 (1–2) (2002) 27–36, http://dx.doi.org/10.1016/S0950-7051(01)00118-6.

[34] F. Dahlstrand, Alarm analysis with fuzzy logic and multilevel flow models, in: Research and Development in Expert Systems XV, Springer, London, UK, 1998, pp. 173–188, http://dx.doi.org/10.1007/978-1-4471-0835-1_12.

[35] D. Kirchhübel, X. Zhang, M. Lind, O. Ravn, Identifying causality from alarm observations, in: International Symposium on Future Instrumentation & Control for Nuclear Power Plants, ISOFIC 2017, 2017, pp. 1–6.

[36] X. Zhang, Assessing Operational Situations (Ph.D. thesis), Technical University of Denmark, 2015.

[37] B. Chandrasekaran, Functional representation and causal processes, Adv. Comput. 38 (1994) 73–143, http://dx.doi.org/10.1016/S0065-2458(08)60176-X.

[38] J.C. Joe, C.R. Kovesdi, MTO-3 . 1 : A human factors evaluation of an advanced human system interface for the generic pressurized water reactor simulator, in: Enlarged Halden Programme Group Meeting, no. May, 2019, pp. 1–10.

[39] E.K. Nielsen, M.V. Bram, J. Frutiger, G. Sin, M. Lind, A water treatment case study for quantifying model performance with multilevel flow modeling, Nucl. Eng. Technol. 50 (4) (2018) http://dx.doi.org/10.1016/j.net.2018.02.006.

[40] E.K. Nielsen, A. Gofuku, X. Zhang, O. Ravn, M. Lind, Causality validation of multilevel flow modelling, Comput. Chem. Eng. 140 (2020) 106944, http://dx.doi.org/10.1016/j.compchemeng.2020.106944.

[41] M. Lind, X. Zhang, Functional modelling for fault diagnosis and its application for NPP, Nucl. Eng. Technol. 46 (6) (2014) 753–772, http://dx.doi.org/10.5516/NET.04.2014.721.

[42] W. Wang, M. Yang, Implementation of an integrated real-time process surveillance and diagnostic system for nuclear power plants, Ann. Nucl. Energy 97 (2016) 7–26, http://dx.doi.org/10.1016/j.anucene.2016.06.002.

[43] M. Song, A. Gofuku, Planning of alternative countermeasures for a station blackout at a boiling water reactor using multilevel flow modeling, Nucl. Eng. Technol. 50 (4) (2018) 542–552, http://dx.doi.org/10.1016/j.net.2018.03.004.

[44] M. Song, A. Gofuku, M. Lind, Model-based and rule-based synthesis of operating procedures for planning severe accident management strategies, Prog. Nucl. Energy 123 (2020) 103318, http://dx.doi.org/10.1016/j.pnucene.2020.103318.

[45] M. Lind, Modeling goals and functions of complex industrial plants, Appl. Artif. Intell. 8 (2) (1994) 259–283, http://dx.doi.org/10.1080/08839519408945442.

[46] C. Reinartz, D. Kirchhübel, O. Ravn, M. Lind, Generation of signed directed graphs using functional models, in: A. A., K. J. (Eds.), IFAC PapersOnLine 52 (11) (2019) 37–42, http://dx.doi.org/10.1016/j.ifacol.2019.09.115.

[47] V. Venkatasubramanian, R. Rengaswamy, S.N. Kavuri, A review of process fault detection and diagnosis: Part II: Qualitative models and search strategies, Comput. Chem. Eng. 27 (3) (2003) http://dx.doi.org/10.1016/S0098-1354(02)00161-8.

[48] J.A. Bondy, U.S.R. Murty, Graph Theory with Applications, Elsevier, New York, 1976.

[49] M. Lind, An introduction to multilevel flow modeling, Nucl. Saf. Simul. 2 (1) (2011) 22–32.

[50] E. Arroyo, Capturing and Exploiting Plant Topology and Process Information as a Basis to Support Engineering and Operational Activities in Process Plants (Ph.D. thesis), Helmut-Schmidt-Universität, Hamburg, 2017.

[51] J. Wu, M. Lind, X. Zhang, K. Pardhasaradhi, S. Pathi, C. Myllerup, Knowledge acquisition and representation for intelligent operation support in offshore fields, Process Saf. Environ. Prot. 155 (2021) 415–443, http://dx.doi.org/10.1016/j.psep.2021.09.036.

[52] J. Itoh, A. Sakuma, K. Monta, An ecological interface for supervisory control of BWR nuclear power plants, Control Eng. Pract. 3 (2) (1995) 231–239, http://dx.doi.org/10.1016/0967-0661(94)00081-Q.