Journal Title
XX(X):1–19
©The Author(s) 2016
Reprints and permission:
sagepub.co.uk/journalsPermissions.nav
DOI: 10.1177/ToBeAssigned
www.sagepub.com/
SAGE

A comparison of hazard analysis methods capability for safety requirements generation

Nanda Anugrah Zikrullah[1], Hyungju Kim[2], Meine J.P. van der Meulen[3], Gunleiv Skofteland[1, 4] and Mary Ann Lundteigen[1]

Abstract

A safety-critical system comprising several interacting and software-intensive systems must be carefully analyzed to detect whether new functional requirements are needed to ensure safety. This involves an analysis of the systemic properties of the system, which addresses the effect of the interaction between systems and system parts. The paper compares two hazard analysis methods, which are often considered well-suited for such software-intensive systems: the Functional Hazard Analysis (FHA) and Systems-Theoretic Process Analysis (STPA). The focus is on the selection and improvement of the best methods, based on the lesson learned from the comparison of FHA and STPA. The analyses cover the hazard analysis processes, systemic properties, and the criteria of requirements. The paper concludes that STPA is the better choice over FHA. Insights are obtained to align both STPA and FHA methods with the broader topic on risk management, i.e., hazard analysis method improvement, cautionary thinking, uncertainty management, and resilience management.

Keywords

Hazard analysis, functional hazard analysis, systems-theoretic process analysis, functional requirement, software-intensive system

Introduction

When novel technologies involving more electronics and programmable systems are developed to increase the efficiency and safety of a system in the industry, it may lead to more complex interactions of hardware and software, with failure modes that are difficult to foresee. Failures may not only stem from component failures, but can also be systemic due to unintended interaction of component and functions[1,2]. Hence, it is important to select suitable analysis tools to identify possible ways in which the system might fail, including systemic failures. Many sectors rely on IEC 61508[3] to qualify novel Electrical/Electronic/Programmable Electronic technology for systems that are critical for ensuring industrial facilities' safety. According to the standard, a hazard analysis process is necessary before the system can be qualified for operation[3–5].

A good starting point before selecting a hazard analysis method is to define the relevant terms. Hazard is defined as *a source of danger that may cause harm to an asset*[6]. A hazardous event is *the point at which control of the hazard is lost*[6]. The event involves interaction between the hazards and the contextual conditions (e.g., environmental state or human activity). Hazard analysis is a process to identify hazards, hazard consequences, and the causal scenarios (or factors) leading to the hazards[5]. Management of such hazards (e.g., by prevention or mitigation) may result in additional system requirements that might affect its design, operation, and maintenance activities[3,7].

If the hazard analysis methods are to be applied to novel technologies, they must have several characteristics. For example, the methods should be suitable for analyzing functions, rather than their realization. This means that the analysis should consider the expected (or specified) behavior that may harm the system, rather than the actual behavior since many of the realization details are abstract[4–6,8]. Also, the method should facilitate a systemic approach[1,9], whereby the system elements and the implication of their interactions are revealed at the system level. Last, the methods should allow for a structured approach to producing new design and operation requirements based on hazardous scenarios[10]. The purpose is to integrate the hazard analysis results in the system development process. Based on the above-described characteristics, we identified several alternative methods of hazard analysis: Preliminary Hazard Analysis (PHA), Functional Hazard Analysis (FHA), Software System Failure Mode and Effect Analysis (SSFMEA), Hazard and Operability study (HAZOP), Systems-Theoretic Process Analysis (STPA), and Functional Resonance Analysis Method (FRAM)[1,2,5,11]. Some of these methods have been advocated as part of the sector-specific standards, including aerospace industry[4] (FHA), automotive industry[12]

[1]Norwegian University of Science and Technology (NTNU), Norway
[2]University of South-Eastern Norway (USN), Norway
[3]DNV GL, Norway
[4]Equinor, Norway

Corresponding author:
Nanda Anugrah Zikrullah, Department of Mechanical and Industrial Engineering, Norwegian University of Science and Technology (NTNU), Trondheim 7491, Norway
Email: nanda.a.zikrullah@ntnu.no

(HAZOP), and process industry[13] (PHA and HAZOP). STPA and FRAM are relatively recent hazard analysis methods that have attracted wide attention[14–16]. STPA has recently been recommended in ISO/PAS 21448[17] to ensure the safety of the intended functionality of autonomous vehicles. Variants of the above-described methods are not explored further in this paper (e.g., control-HAZOP is considered HAZOP). The only exception is in SSFMEA, which is a system-based analysis, whereas the original Failure Mode and Effect Analysis (FMEA) is a component-based analysis.

The long list of hazard analysis methods makes the selection for the most suitable method a challenge. The main objective of this paper is to analyze and compare the hazard analysis methods based on the characteristics mentioned above. The goal is to select and, where needed, improve the best method for hazard analysis of novel technology. The objective comprises the following three research questions:

RQ1. How do the selected hazard analysis methods identify the same or different functional hazards?
RQ2. How do the selected hazard analysis methods provide a systemic perspective on the system for analysis?
RQ3. What are the main differences between the derived safety requirements?

The remaining part of this paper is organized as follows. The next section provides a review of the list of hazard analysis methods and the preliminary selection made to limit the comparison process into two methods based on the derived characteristics. The *methodology* section describes the approaches to answer the research questions and includes the procedures for hazard analysis. The *case study* section describes the example from the oil & gas industry to demonstrate the two methods' capability. This is followed by a presentation of the *results* of the analysis and discussions on the findings. Section *overall implication* contains our recommendations and the implications for other subject areas. The final section concludes the finding in the paper.

Review of the hazard analysis methods

We reviewed the hazard analysis methods to limit the number of methods to be considered for further analysis into a maximum of two. We identified two attributes that capture the methods' functional and systemic characteristics: the ability to capture the undesired functional behavior and the linearity of the utilized accident model. The requirement generation characteristic requires an in-depth understanding of the methods' results. Hence, it was not considered suitable for inclusion as part of the preliminary review.

Ability to capture the undesired functional behaviors

During operation, the actual behavior of functions may deviate from expectations. Examples of the functional behavior are the realization of function (e.g., activated, not activated, when needed, not needed, as required, too short, or too much) and the function timing (e.g., correct, early, or late). The undesired functional behavior needs to be assessed according to the context (e.g., where and when it may occur) to be classified as a functional hazard.

All methods have different procedures to identify hazards (e.g., the required inputs, the process, and the outputs[10]). Some methods might have influenced each other during decades of development, resulting in substantially similar hazard identification procedures. For example, PHA was designed to analyze broader types of hazards, including energy source, functional, operational, component, material, lesson learned from other systems, undesired mishaps, and failure modes[5]. These hazards are captured through the use of a checklist. PHA is designed to be a preliminary analysis and has extensive coverage. The results of the analysis performed using the method suffer from the lack of depth, and therefore additional methods are needed to supplement the process.

Ericson[5] recommends using FHA for analysis of functional hazards because the method utilizes a list of functional hazard types (e.g., functional failure, operates incorrectly, and function timing). A variant of FHA called Functional Failure Analysis (FFA) focuses on how the function can fail[18]. Both of them are deemed the same method because they utilize a similar functional hazard type list. Many authors also consider FFA a variant of FMEA known as *predictive* FMEA, due to the utilization of the FMEA method[18]. The FMEA method involves systematic checking for possible combinations of functions, failure mode types, and operational mode. In this paper, the term FHA is used to represent FHA and FFA.

According to Pumfrey[18], both SSFMEA and (software) FHA utilize the same procedures to identify undesired functional behavior. SSFMEA is tailored to analyze the software's functional behavior. By contrast, HAZOP was initially developed to analyze hazard and operational problems in system design[6]. HAZOP analyzes combinations of parameters (e.g., flow or pressure) and guide words (e.g., more, less, no) to check the possible deviation from the design intent. STPA regards hazards as all unsafe control actions (UCAs) performed by controllers to the system (or controlled processes) that occur in a specific context[1]. Finally, FRAM checks whether the aggregation (or coupling) of the variability of all functions in the system may result in an increased, unchanged, or dampened variability at the system level[2].

Linearity of the accident model

Causal analysis processes for the hazards are developed based on an accident model. Hollnagel[2] states that the accident models can be classified into three types, based on differences in their principles of causality: simple linear models (e.g., the Domino model), complex linear models (e.g., the Swiss Cheese model), systemic model (e.g., the Systems-Theoretic Accident Model and Process (STAMP) and the Functional Resonance Accident Model). In a simple linear model, the accident is caused by a linear sequence of causes (e.g., failures, errors, or organizational problems). Here, the focus is to provide recommendations to eliminate one cause in the sequence. In a complex linear model, dependencies between events may affect the event sequence that results in accidents. To manage this dependency, the focus is shifted by strengthening the barriers and defenses. In a systemic model, the dependencies are not only due to a combination of events but also due to complex couplings between interacting components. An accident can be prevented by controlling the system state to prevent transition into an uncontrolled (unsafe) state[1,2].

Both the simple linear model and the complex linear model have been utilized in the causal analysis process of the hazard analysis methods such as PHA, FHA, SSFMEA, and HAZOP. Initially, the causal analysis focuses only on finding the direct root cause of a hazardous event (simple linear model). This approach works due to the simplistic type of system utilized at the time (i.e., mechanical or hydraulic system). When the number and complexity of the system's components increase (i.e., electronic system), the interaction or dependency may become a significant contributor to a hazardous event. Hence, a complex linear model is then adopted to the traditional method to increase the analysis coverage. The shift from utilizing a simple linear accident model to a complex linear accident model shows how the methods' causal analysis process is evolving depending on the system to be analyzed (i.e., simple or complex).

Recently, the systemic accident model has been developed to include the different complexity characteristic of the system. Leveson's[1] and Hollnagel's[2] criticize of the limited perspective of the linear accident models. According to them, while dependencies are considered already in the complex linear model, they still occur due to combinations of failures. A systemic model allows for identifying possible harmful interactions without failure in the system. STPA and FRAM are the hazard analysis methods that utilize the systemic accident model

Several comparison analysis results support their critics. For example, Leveson et al.[19] perform a comparison between STPA and the ARP 4761 safety assessment process and claim that the former is better for safety assessment. However, they did not indicate whether this difference in result is due to the accident model used or due to the flaws of the methods utilized in ARP4761. For example, to claim that FHA (part of ARP 4761) considers only failures during the analysis does not mean that it is limited to consider component failure as a cause. It is possible to expand the perspective to the systemic level and find that functional failure can also be caused by an interaction problem between two or more components (without any failure). This argument shows that the limitation in ARP4761 is not because of the method but by the accident model's limitation.

Yousefi et al.[15] compare AcciMap, STAMP, and FRAM. This comparison focuses on the systemic model and does not discuss the contrast with the linear model. In another research, Sulaman et al.[20] have a different claim. They perform a comparison between Software System FMEA (SSFMEA) and STPA for a collision-avoidance system and conclude that neither method is superior. Some hazards are unique to both SSFMEA and STPA. They claim that both methods complement each other. The SSFMEA method that they utilized focuses more on component failures and does not have a systemic perspective of the system due to the bottom-up approach.

The examples above show the systemic accident model's advantages over the linear accident model for causal analysis. This does not mean that the traditional hazard analysis methods (e.g., FHA) are not as good as the new hazard analysis methods (e.g., STPA and FRAM). The shift from a simple linear model to a complex linear model in the traditional methods indicates that they can apply a new accident model for improvement. If the systemic model is as better as it is claimed, research on its application need to be performed with the traditional methods. This would provide users with options to develop the traditional methods (if possible) or to utilize the new methods.

Table 1. Review of hazard analysis method attributes

Methods	Ability to capture undesired functional behavior	Linearity of the accident model
PHA	Type of functional hazards (can be expanded to other type of hazards)	Complex linear model
FHA	Type of functional hazards	Complex linear model
SSFMEA	Type of functional failures	Complex linear model
HAZOP	Combination of guidewords and parameters for process condition	Complex linear model
STPA	Type of unsafe control actions	Systemic model
FRAM	Aggregation of variability in the function	Systemic model

Method selection

Table 1 summaries the attributes of the reviewed hazard analysis methods. The varying abilities to capture the undesired functional behavior make it difficult to distinguish between each method. Therefore, the method selection is mainly based on the linearity of the accident model, with one method for each model. This is also to verify the claim for the systematic accident model advantages over the linear accident model. Logical reasoning and reviews of relevant literature are performed to support the decision.

For the complex linear model, PHA, FHA, and SSFMEA have similar procedures in capturing the undesired functional behavior, with FHA as the recommended method for analyzing functional hazards. Comparatively, HAZOP may not be suitable for analyzing novel technology due to a lack of detailed system design. Therefore, FHA is selected for the method with a complex linear model.

For the method with a systemic model, we refer to the comparison analysis by Yousefi et al.[15]. He finds that STPA is more capable of finding hazards systematically as compared to FRAM. We use this finding as the basis for the selection of STPA in the paper.

Methodology

The research methodology is as follows. First, FHA and STPA are performed separately on a case study. The functional list's input to both methods is controlled to be the same to accentuate the differences between both methods' results. It is validated by associating each function in the FHA to the function in the STPA. Both hazard analyses are performed by the same person (first author). This may introduce subjectivity in the assessment process. Verification is performed by all the authors on the presented results to reduce the subjectivity. The case study focus is on both method's ability to identify hazards and produce requirements. Therefore we decided not to do a risk assessment for both methods.

Then, a comparison analysis is performed to answer the RQ1. A mapping between FHA and STPA procedures is required for the comparison process, which is described later. The analysis focuses on analyzing the cause of the similarity or difference of the results from every step of the hazard analysis methods.

RQ2 is answered by comparing the properties of the causal scenarios with the system properties. We utilize the Composition, Environment, Structural, and Mechanism (CESM) model[21] as the reference system properties. Composition refers to every component that built the system (e.g., controller, sensor). Environment refers to the boundary condition in which the system may influence or be influenced by (e.g., water depth or temperature). Structure refers to the (physical or abstract) relation between the components or the components and the environment in the system (e.g., communication between components). Mechanism is a process that describes the behavior of a given component, structure, or environment (e.g., interaction in the software function). According to Wan[22], the CESM model can aid in investigating systemic behavior (i.e., emergence). Thus, we can evaluate whether these four properties in the hazard analysis method can lead to the identification of systemic causal scenarios.

RQ3 is answered by evaluating the requirements against the criteria for a requirement. While there is no consensus on what makes a good requirement, Holt et al.[23] state that these eight criteria should be considered: (1) identifiable, (2) clear, (3) solution-specific, (4) have ownership, (5) have origin, (6) verifiable, (7) able to be validated, and (8) have priority. (1) *Identifiable* refers to the ability of the requirements to be traced back to their cause. (2) *Clear* refers to the need to have unambiguous meaning for every requirement. (3) *Solution-specific* refers to the application of the requirements to a specific system. (4) *Have ownership* refers to the stakeholders that need to satisfy the requirements. (5) *Have origin* refers to the targeted subjects that need to follow the requirements. (6) *Verifiable* refers to the ability of the requirements to be checked for correctness by the designer. (7) *Able to be validated* refers to the ability of the requirement to be demonstrated for compliance. (8) *Have priority* refers to the relative level of importance of one requirement to the other. We assumed that the above criteria are necessary to form a requirement that can be utilized immediately for decision making. Thus, we can utilize them to evaluate whether the hazard analysis methods can provide such requirements.

Finally, all the research questions' analyses results are discussed at a higher level to conclude the selection of the better method for hazard analysis of novel technology. The research's implication is analyzed according to the risk management topic in general, to indicate the required next step for integration of the method with the safety assessment process.

The following subsections describe the FHA and STPA procedures. Modifications are applied based on the identified literature. Afterward, a mapping of FHA and STPA procedures is provided for the comparison analysis process.

Functional Hazard Analysis (FHA) procedure

FHA procedures have evolved over the years. It seems that there is no consensus on how exactly FHA should be performed[4,5,8,24]. While there are different wordings and number of steps in FHA from different sources, essentially, the procedure includes the following seven steps:

1. Describe the system. The system description may be obtained from the conceptual design and operation of the system and functional list[5].
2. Model the interactions of the functions. The model may be constructed based on the functional list. While this step is not recognized as a separate step in the referred documents, Ericson[5] recommends using a model to aid the analysis. Examples of the modeling methods are the functional flow diagram and Functional Analysis Structure method (FAST) diagram.
3. Identify hazards. Hazards may be identified systematically by checking the combinations between functions, operational modes, and functional failure modes[4]. The operational modes are obtained based on the conceptual operational procedures for each function. The functional failure mode is a generic list that is defined early before the hazard identification starts. Examples of functional failure modes: *functional loss*, *unintended activation*, and *incorrect operation*[24].
4. Identify consequences. Each consequence may be identified by checking the possible propagation effects from the functional hazard to the system level (e.g., using an inductive method[5]).
5. Analyze causal factors (or scenarios). A single (or a combination of) causal factor(s) may form a scenario that caused hazards. The causal factors are based on conceptual design and operation, the function model, and historical experiences. ARP 4761[4] focuses on causal factors due to failure. As argued in the previous discussion, it may be possible to expand the causal factors' perspective into possible scenarios involving multiple causal factors with no failure. No failure means that the system has been implemented according to the specification, but the specification lacks the ability to handle the scenarios.
6. Assess risk. The risks for every hazardous event are assessed from the magnitude of the consequences and the likelihood of every causal scenarios[4]. According to Rausand[6], the risk analysis process for the hazard analysis method may be qualitative (e.g., utilizing qualitative scale) or semi-quantitative (e.g., utilizing risk priority number).
7. Provide recommendations or generate functional requirements. Depending on the analysis purpose, it is possible to either directly recommend solution(s) to prevent/mitigate the hazard or to generate a functional requirement[24] as guidance during the detailed design process. The first option is preferable for mature technology with historical experience. For the conceptual design of new technology, functional

requirements are better as they do not limit the possible solutions. The functional requirements can be coupled with other methods (e.g., FTA, FMEA, and common cause analysis) to derive the non-functional requirements (e.g., reliability and safety performance requirements) as performed in ARP 4761[4].

Several researchers (see, e.g.,[8,24]) has demonstrated the FHA for hazard analysis at the system level and find several weaknesses of FHA. Allenby and Kelly[24] argue that the generic functional failure mode list in step 3 still has a limitation due to the overuse of incorrect operation hazard type as the complementary keyword to capture abstract functional failures. They propose to utilize HAZOP guide words to obtain more comprehensive safety requirements[24]. Besides, the processes of causal and consequence analysis are still based on a brainstorming process that does not guarantee the completeness of the results[5,8]. Wilkinson and Kelly[8] claim that it is challenging to discover coupling or dependent failure causal scenarios using the brainstorming process.

Based on the identified weaknesses above, we made several considerations for FHA's application in our study. First, system modeling was supported by using a FAST diagram. A FAST diagram depicts the model sequence and dependency between functions (e.g., main, supporting, and continuous)[25]. Each function is modeled as a box with connections to the other functions and may have different roles in the system (e.g., main function or supporting function). In the FAST diagram, the right function is the precursor of the function to the left (a sequence).

Next, HAZOP guidewords (i.e., omission, commission, late, early, and value) were utilized for a functional failure mode list as recommended by Allenby and Kelly[24] to have a comprehensive scope for the analysis.

For causal scenario analysis, we utilized the FAST diagram and the system conceptual design and operation. The possible causal scenario was obtained by identifying the potential agent (or component) performing the function and its dependency on the next function. Information from the conceptual design and operation is used to infer the agent's (e.g., temperature or pressure) possible external effect on the system. We decided not to go too deep into detail to maintain simplicity (e.g., rotor, stator, or motor shaft failure would be assumed as one pump motor hardware failure).

We developed a rule for safety requirement generation to transform the functional failure mode keywords into functional requirement keywords. The transformation rules are listed in Table 2.

Systems-Theoretic Process Analysis (STPA) procedure

STPA utilizes system theory and system thinking based on STAMP. The STPA procedure consists of four steps[26]:

1. Define the purpose of the analysis
   (a) Describe the system. The system description is based on the conceptual design and operation of the system and functional list.
   (b) Identify System-level Loss, System-level Hazards, and System-level Safety Constraints. They may be obtained through a brainstorming process based on system description and experience from similar systems.
2. Model the control structure
   (a) Identify controller responsibility and process model. They may be developed based on a system description. They describe how the controller responds to new/updated information.
   (b) Build the Hierarchical Control Structure (HCS) model. The model is constructed based on the functional list, controller responsibility, and process model. Every agent in the system (e.g., controller, controlled process, or supporting system) is modeled as a box. Each box may have connections (modeled as arrows) with other boxes based on the functions (e.g., control actions or feedbacks). In the HCS, the controller is an agent responsible for controlling agents at the lower hierarchy level.
3. Identify Unsafe Control Actions (UCA)
   (a) Identify UCAs. Each UCA may be identified by checking the combination between control actions, environmental conditions/system states, and UCA types. Control actions are obtained from the controller responsibilities. Environmental conditions are obtained from the process model. There are six types of UCA: control action not provided when needed, provided when not needed, provided too late, provided too early, stopped too soon, and applied too long.
   (b) Generate Controller Constraints (CC). Each CC may be generated by transforming the UCA type keywords into constraint keywords (e.g., not provided is transformed into must provide)[27].
4. Identify Loss Scenarios (LSc). Each scenario may be identified based on every aspect in the control loop (e.g., controller, sensor, actuator, controlled process, communication, and environmental influence).

Several researchers have demonstrated STPA for analysis of complex systems[28–30] and found several weaknesses of STPA. Due to the attempt to increase the hazard coverage, STPA suffers from a state explosion of the number of UCAs to be analyzed[31]. Prioritization is required as follow up to focus the available resource. Also, the use of STPA is not straightforward since it requires the analyst to develop an HCS. This may not be a familiar task for the common practitioner of hazard analysis[28]. Finally, Kim et al.[30] also question the absence of stop criteria preventing the analyst from going too deep into the details.

Based on the identified weaknesses above, we have made considerations for applying STPA in our study.

First, we did not perform a prioritization for STPA since it does not conform with the original intent of the STPA method by Leveson[1]. She argues that the main strength of STPA is to derive a comprehensive list of safety constraints. Those interested in the risk analysis process for STPA may refer to the paper by Kim et al.[31].

Next, we utilized a recommendation by Kim et al.[32] when modeling the system. They propose to include the

Table 2. Transformation rule from keywords into FHA functional requirement and STPA controller constraints

Keywords	FHA functional requirements	STPA controller constraints
Omission / Not provided (when needed)	. . . must be provided . . .	. . . must provide . . .
Commission / Provided (when not needed)	. . . must not be provided	. . . must not provide. . .
Provided too late	. . . must work within required time . . .	. . . must provide . . . within required time . . .
Provided too early	. . . must not start working too early	. . . must not provide . . . too early . . .
Stopped too soon	–	. . . must provide . . . continuously as required . . .
Applied too long	–	. . . must stop providing . . . after the condition changes
Provided wrong value	. . . must be provided correctly . . .	–

power supply as part of the control action and include it for UCA identification. This may avoid the omission of essential hazards from the analysis. The power supply was modeled as a supply function with a green arrow in the HCS model.

Like FHA, we developed a rule to transform the type of UCA keywords into controller constraints keywords for controller constraint generation. The transformation rules are listed in Table 2.

Comparison analysis procedure

The descriptions of FHA and STPA procedures show that they have different methods and perspectives on analyzing hazards. However, the core objectives of each step are similar. For example, step 3 *identify hazards* of FHA and step 3 *identify UCAs* of STPA are processes to identify hazardous events (or hazards in STPA). Table 3 shows the mapping of both FHA and STPA procedures based on each step's core objectives. The listed terms for each step of FHA and STPA denote the different terms used by each method during the specific phase of the analysis. Table 3 also shows how the process of FHA (2a-6a) and STPA (2b-6b) are different.

The mapping of both method procedures allows comparing the case studies' results in each analysis step. The analysis is performed at a higher level to avoid the influence of technical discussion that may blur both method's characteristics and presented in separate discussions. Specific to the comparison of causal scenarios and safety requirements, we utilized the previously mentioned approaches to answer the RQ2 and RQ3.

Case Study

The Åsgard subsea compression system in Norway[33] inspires the case study, where two protection systems (process control and safety) exist independently of each other. The integration of process control & safety concept is a novel technology applied as an alternative solution to reduce the complexity of the physical architecture[9,34]. This concept is part of the use case in the Safety 4.0 project, where the goal is to develop a standardized safety demonstration approach for novel subsea technologies[34]. This concept may increase software complexity, thus decreasing the confidence in its functional capability. This case study is deemed as sufficiently complex and relevant for use in our study.

System description

The system process flow diagram is illustrated in Figure 1. Redundant equipment and utility systems (e.g., network switches) are not illustrated in the Figure 1 for simplification. The subsea compression system consists of a scrubber, a compressor, and a pump. The system's goal is to ensure high gas flows and recovery rates from the well. The liquid mixture is recovered from the well and goes to the scrubber for separation. The dry gas is then compressed in a compressor, while a pump pumps the separated liquid. Both the dry gas and the liquid are then delivered to the topside facility for further processing. The study focuses specifically on the control and safety mechanism in the pump. A high voltage electronic power unit is used to power the pump operation. Here, the Process Control System (PCS) is utilized to maintain the level of liquid inside the scrubber by changing the pump's speed. If the liquid level gets too low, the gas can go through the pump (gas blow-by) and cause overpressure downstream[35]. The Process Shut Down system (PSD) is implemented to increase the pump protection system's integrity by shutting down the pump in case of the low-low (a technical term to describe the low limit for PSD) level detection in the scrubber.

The PCS loop consists of level sensors, Master Control Station (MCS), operator, PCS node, driver controller, and other systems. The level sensor detects the deviation of process condition and sends the signal to MCS for automatic logic solver response. Information from the MCS is also provided to the operator to see whether manual intervention is required. Depending on the control loop mode (automatic or manual), the PCS node needs to select the prioritized response (from either the MCS or the operator command) to the driver controller for regulating (increase or decrease) the pump speed.

The PSD loop consists of level sensors, MCS, PSD node, relay & breaker, operator, and other systems. The level sensor detects whether abnormal condition occurs in the system and informs the MCS for automatic logic solving response. During an abnormal condition, MCS needs to automatically shut down the equipment by passing information through the PSD node to relay & breaker to stop the pump's power supply. It is also possible to receive shutdown command from other systems in case of emergency. In this case, the operator is responsible for shutting down the power supply directly.

Table 3. Mapping of FHA and STPA procedures

FHA term	FHA	(Generic) hazard analysis procedures	STPA	STPA term
–	1	System description	1	–
Functional analysis structure technique diagram	2a	System modelling	3b	Hierarchical control structure
Hazardous events	3a	Hazard identification	4b	Unsafe control action
Consequence	4a	Consequence identification	2b	System-level loss & System-level hazard
Causal scenario	5a	Causal scenario analysis	5b	Loss scenario
Safety requirement	6a	Safety requirement generation	6b	Controller constraints

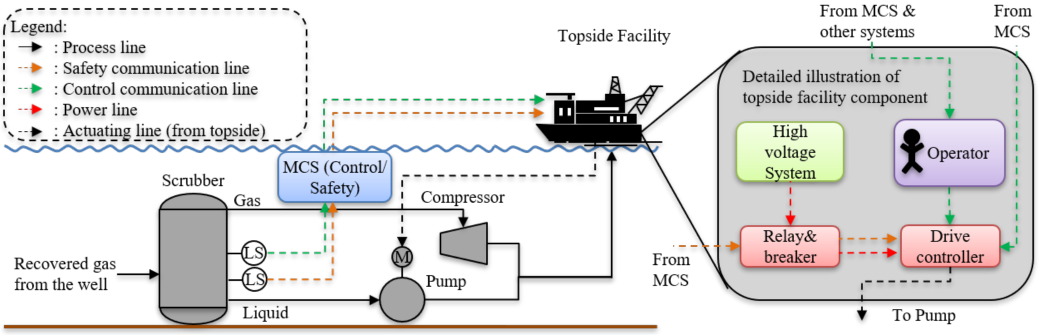


Figure 1. Simplified process flow diagram of subsea gas separation and compression to topside facility with the communication lines for PCS and PSD

In this system, a physical integration with a logical separation concept[9] is implemented at the Master Control Station. It means that the PCS and PSD share the same hardware while separated logically in the software architecture. They are designed to work parallel to each other, with the safety system has higher priority over the process control system when utilizing the same hardware resources.

Results

FHA results

The functions of the described system were modeled in the FAST diagram, as illustrated in Figure 2. The top path describes the pathway for activation of safety function while the bottom path describes the pathway for activation of process control function. Each function's operational mode was specified based on the output of the targeted function's preceding function and condition. For example, the operational mode of *aut. command pump shutdown* function was the output of *detect abnormal level* (i.e., normal, low, or low-low) and the condition of *detect pump status* (i.e., running, unknown, or stopped). The complete functional list and operational mode are listed in Table 4.

Examples of the FHA results for step 3a-5a are presented in Table 5. The hazard identification process identified 64 hazardous events from 168 possible combinations (between functions, operational modes, and the failure mode list). Identification of the consequences showed that 21 HEs might result in Con1 *equipment damage*, 40 HEs might result in Con2 *unnecessary loss of production*, and 3 HEs might result in both types of losses. The causal analysis process identified 206 possible Causal Scenarios (CaS) associated with the 64 hazardous events (HE).

Safety Requirements (SR) are generated for the functions based on the identified HEs and CaSs. 64 SRs corresponded one to one to the identified HEs. The identified CaSs were included in the SRs as guidance during the formulation of prevention/mitigation solutions. Examples of the SRs based on the HEs listed in Table 5 are (the SR format is *SRId. SR [CaSId]*. SRId and CaSId refer to the numbering of the SR and the related CaS):

- SR001. Stop pump function must be provided within the required time when there is shutdown command, and the pump status is running / unknown [CaS001]
- SR015. Aut. command pump shutdown function must not be provided when scrubber level status is normal, and the pump status is running / unknown [CaS050-056]
- SR043. Command change pump output function must be provided correctly when the priority check result is to change pump output, and the pump status is running [CaS125-126]
- SR048. Aut. pump output change command function must be provided when scrubber level status is low, and the pump status is running / unknown [CaS142-148]

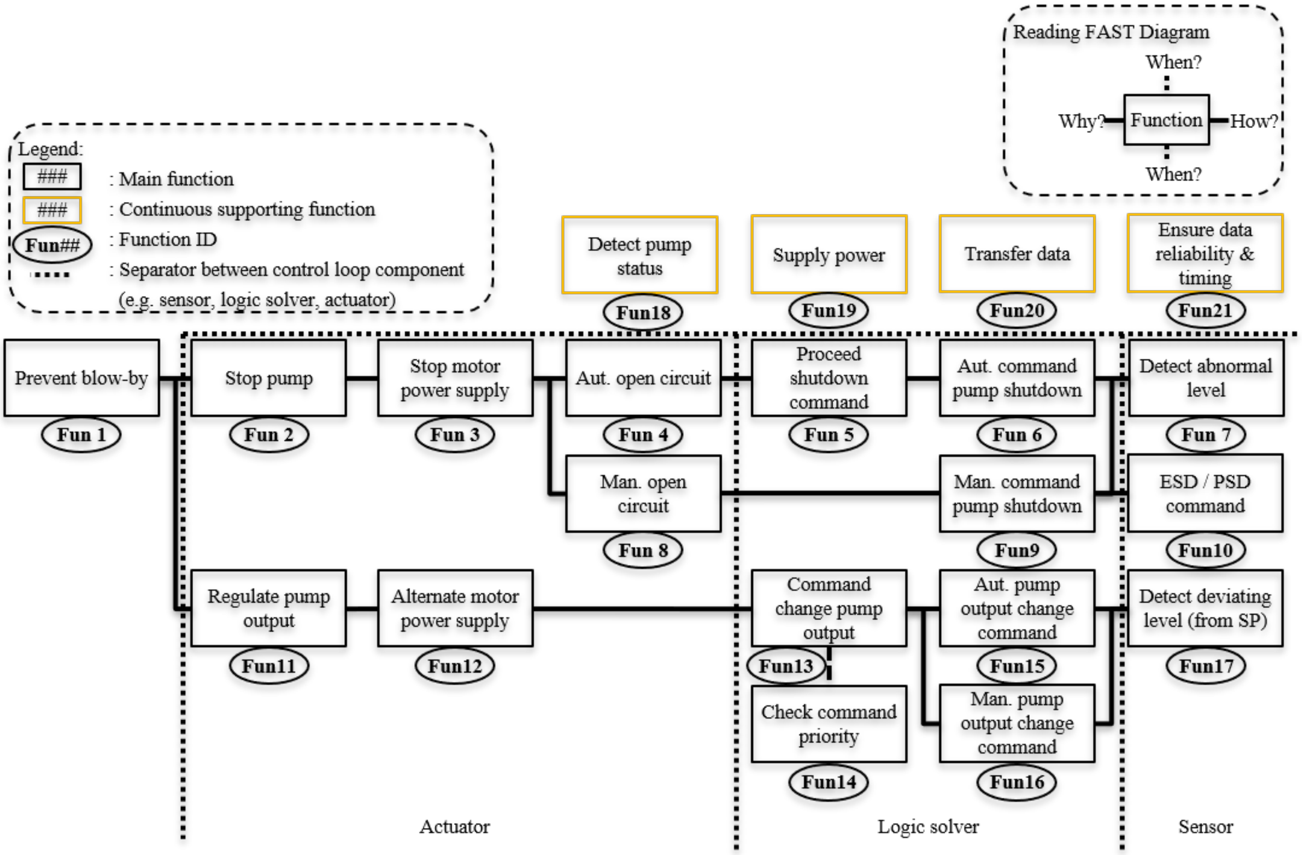


Figure 2. FAST diagram of pump protection system

STPA results

The boundaries of STPA analysis were the System-level losses, System-level hazards (H), and System-level safety constraints, as listed in Table 6. The equipment protection system was modeled as an HCS in Figure 3. The complete list of functions, associated agents, function types, and process models are listed in Table 4.

UCAs were identified from the combination of control actions, process models, and UCA types. In total, out of 134 identified combinations, 56 were classified as UCAs. 15 UCAs might result in H1, 32 UCAs in H2, and 9 UCAs in H3. Table 6 shows that H.1 corresponds to L1 (15 UCAs), while both H2 and H3 correspond to L2 (41 UCAs combined). Examples of identified UCAs are (the UCA format is *UCAId. UCA [HId]*. UCAId and HId refer to the numbering of UCA and H):

- UCA001. Pump motor provides stop pump command to the pump too late when there is a shutdown command, and the pump status is running / unknown [H1]
- UCA015. MCS provides Aut. command pump shutdown to the PSD node when Scrubber level status is normal and the pump status is running / unknown [H3]
- UCA026. Pump motor stops providing regulate pump output to the pump too soon before the condition there is a command to change pump output, and the pump is running changes [H2]
- UCA027. Pump motor provides regulate pump output to the pump too long after the condition, there is a command to change pump output, and the pump is running changes [H2]
- UCA044. MCS does not provide Aut. pump output change command to the PCS node when scrubber level status is low, and the pump status is running / unknown [H2]

The control loops associated with every UCA were analyzed further to identify the Loss Scenario (LSc). There are 346 identified LScs. Examples of the LScs are (The format of LSc is *UCAId.LScId. LSc*. UCAId and LScId refer to the numbering of UCA and LSc. UCAId.LScId shows the link between every LSc to the associated UCA):

- UCA001.LSc001. Local battery as spare power prevents an automatic shutdown of the pump
- UCA015.LSc093. Problem in the control path caused by unreliable data from topside communication
- UCA015.LSc094. Problem in the control path information caused by topside communication failure
- UCA015.LSc095. Problem in the received information caused by unreliable data from subsea communication
- UCA015.LSc096. Problem in the received information caused by subsea communication failure
- UCA015.LSc097. Problem in the controlled process due to PSD node hardware failure
- UCA015.LSc098. Problem in the controlled process due to PSD node software error
- UCA015.LSc099. Problem in the controller due to MCS hardware failure

Table 4. Functional list of the pump protection system for FAST diagram & HCS

FAST ID	Function	Operational mode / Process model (Condition)	HCS ID	Function type	Agent	Target
Fun02	Stop pump	Shutdown command (Yes/No) Pump status (Run/Stop/Unknown)	C01	Control	Pump motor	Pump
Fun03	Stop motor power supply	Shutdown command (Yes/No) Pump status (Run/Stop/Unknown)	C02	Control	Driver controller	Pump motor
Fun04	Aut. open circuit	Shutdown command (Yes/No) Pump status (Run/Stop/Unknown)	C03	Control	Relay & breaker	High voltage system
Fun05	Proceed shutdown command	Shutdown command (Yes/No) Pump status (Run/Stop/Unknown)	C04	Control	PSD node	Relay & breaker
Fun06	Aut. command pump shutdown	Scrubber level status (PSD) (Normal/Low/Low-low) Pump status (Run/Stop/Unknown)	C05	Control	MCS	PSD node
Fun07, Fun20	Provide scrubber level status (PSD)	Process condition (Normal/Abnormal)	F06	Feedback	Sensor PSD	MCS
Fun20	Provide process information	Process condition (Normal/Abnormal)	F07	Feedback	MCS	PSD node
Fun20	Provide process information	Process condition (Normal/Abnormal)	F08	Feedback	PSD node	Operator
Fun08	Man. open circuit	Shutdown command (Yes/No) Pump status (Run/Stop/Unknown)	C09	Control	High voltage system	Relay & breaker
Fun09	Man. command pump shutdown	Shutdown command (Yes/No) Pump status (Run/Stop/Unknown)	C10	Control	Operator	High voltage system
Fun10, Fun20	ESD / PSD command	Process condition (Normal/Abnormal)	F11	Feedback	Other systems	Operator
Fun11	Regulate pump power output	Alternate power command (Yes/No) Pump status (Run/Stop/Unknown)	C12	Control	Pump motor	Pump
Fun12	Alternate motor power supply	Alternate power command (Yes/No)	C13	Control	Driver controller	Pump motor
Fun13	Command change pump output	Command priority result (Act/no Act) Pump status (Run/Stop/Unknown)	C14	Control	PCS node	Driver controller
Fun14	Check command priority	Human Command (Yes/No) MCS Command (Yes/No) Priority status (MCS/Human)	C15	Control	PCS node	PCS node
Fun15	Aut. pump output change command	Scrubber level status (PCS) (Normal/Low/Low-low) Pump status (Run/Stop)	C16	Control	MCS	PCS node
Fun16	Man. pump output change command	Scrubber level status (PCS) (Normal/Low/Low-low) Pump status (Run/Stop)	C17	Control	Operator	PCS node
Fun17, Fun20	Provide scrubber level status (PCS)	Process condition (Normal/Abnormal)	F18	Feedback	Sensor PCS	MCS
Fun20	Provide process information	Process condition (Normal/Abnormal)	F19	Feedback	MCS	PCS node
Fun20	Provide process information	Process condition (Normal/Abnormal)	F20	Feedback	PCS node	Operator
Fun18, Fun20	Provide pump motor status	Process condition (Normal/Abnormal)	F21	Feedback	Pump motor	MCS
Fun18, Fun20	Provide relay status	Process condition (Normal/Abnormal)	F22	Feedback	Relay & breaker	PSD node
Fun18, Fun20	Provide driver controller status	Process condition (Normal/Abnormal)	F23	Feedback	Driver controller	PCS node
Fun19	Supply Power	Process condition (Normal/Abnormal)	S24	Supply	High voltage system	Relay & breaker
Fun19	Maintain power supply	Process condition (Normal/Abnormal)	S25	Supply	Relay & breaker	Driver controller

- UCA015.LSc100. Problem in the controller due to MCS (safety) software error
- UCA015.LSc101. Problem in the controller due to unintended interaction between PCS and SIS that cause software error
- UCA015.LSc102. Problem in the received information due to level sensor (safety) hardware failure
- UCA015.LSc103. Problem in the received information due to level sensor (safety) software error

CCs are generated based on the transformation rule to the identified UCAs. 56 CCs correspond one to one to the identified UCAs. The identified LScs are listed to show the possible scenarios, possibly affecting the fulfillment of the constraint. Examples of the generated CCs are (the CC

Table 5. Example of Functional Hazard Analysis (FHA) results

HE Id	Function	Operational mode	Failure mode	ConID	Consequence	CaSD	Causal scenario
HE001	Stop pump	There is shutdown command and the pump status is running / unknown	Function provided too late	Con01	Equipment damage	CaS001	Local battery as spare power prevents auto shutdown of pump
HE. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .
HE015	Aut. command pump shutdown	Scrubber level status is normal and the pump status is running / unknown	Function commission error	Con02	Unnecessary loss of production	CaS050	Topside communication provides unreliable data to the equipment
						CaS051	Topside communication failure
						CaS052	Subsea communication provides unreliable data to the equipment
						CaS053	Subsea communication failure
						CaS054	MCS hardware failure
						CaS055	MCS (safety) software failure
						CaS056	Unintended interaction between PCS and SIS that cause software error
HE. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .
HE043	Command change pump output	Priority check result is to change pump output, and the pump status is running	Function provided wrong value	Con02	Unnecessary loss of production	CaS125	Topside communication provides unreliable data to the equipment
						CaS126	Software error in the PCS part of the MCS
HE. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .
HE048	Aut. pump output change command	Scrubber level status is low and the pump status is running	Function omission error	Con02	Unnecessary loss of production	CaS142	Topside communication provides unreliable data
						CaS143	Topside communication failure
						CaS144	Subsea communication provides unreliable data
						CaS 145	Subsea communication failure
						CaS146	MCS hardware failure
						CaS147	Software error in the PCS part of the MCS
						CaS148	Unintended interaction between PCS and SIS that cause software error
HE064	. . .	. . .	. . .	. . .	. . .	. . .	. . .

Table 6. System-level losses, hazards and safety constraints identified on STPA

L Tag	System-Level Loss (L)	H Tag	System-Level Hazard (H)	SC Tag	System-Level Safety Constraint (SC)
L1	Equipment damage	H1	Equipment operates outside normal operating condition	SC1	Equipment must be protected from extreme operating conditions that can result into damage
L2	Unnecessary loss of production	H2	Equipment operates outside optimal operating condition	SC2	Equipment must be operated within optimal operating conditions
		H3	Unintended stop of equipment when needed	SC3	equipment must be available to work when needed

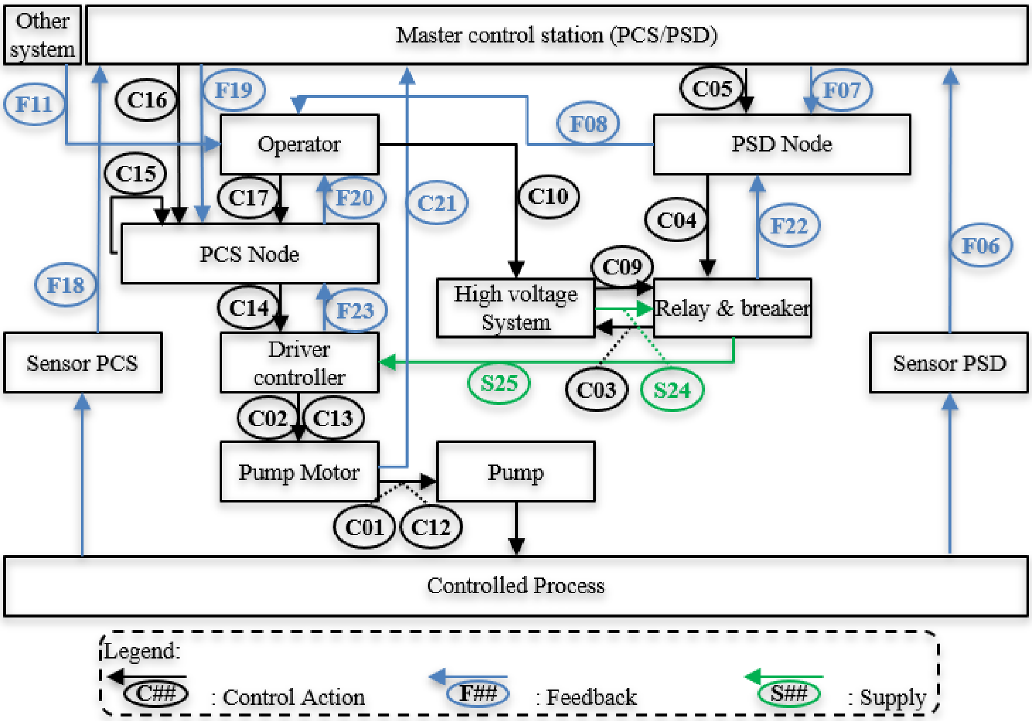


Figure 3. HCS of pump protection system

format is *CCId. CC [LScId]*. CCId and LScID refer to the numbering of CC and the related LSc):

- CC001. Pump motor must provide stop pump to the pump within the required time when there is shutdown command, and the pump status is running / unknown [LSc001]
- CC015. MCS must not provide aut. command pump shutdown to the PSD node when scrubber level status is normal, and the pump status is running / unknown [LSc093-103]
- CC026. Pump motor must provide regulate pump output to the pump continuously as required when there is a command to change pump output, and the pump status is running [LSc149-152]
- CC027. Pump motor must not stop providing regulate pump output to the pump before the condition there is a command to change pump output, and the pump status is running changes [LSc153-157]
- CC044. MCS must provide aut. pump output change command to the PCS node when scrubber level status is low, and the pump status is running / unknown [LSc252-262]

Discussion

The following sections contain discussions of the comparison results from the case study.

Comparison of the modeling techniques

Both analyses utilized a model to assist hazard identification, consequence identification, and causal scenario analysis processes. FHA and STPA utilized different models, the FAST diagram for the former and HCS for the later. Three properties distinguish the two models: model type, function type, and process flow.

The FAST diagram is a model of sequential functions, while HCS is a control structure model. In the FAST diagram, as seen in Figure 2, the focus is to depict how each function interacts with other functions in a structured and sequential manner to achieve the desired function. It is unknown which agent (system or subsystem) performs each function. Also, the interactions between the system with the environment are not modeled. Comparatively, HCS modeled the conceptual system operation as a structure of control loops. Every function (e.g., control action, feedback, or supply) has a subject (performing the function) and an

Table 7. Differences in function classification between FAST diagram and HCS

Modeling differences	FAST function type	FAST ID	HCS function type	HCS ID
Type 1	Main	Fun02 – Fun06, Fun08, Fun09, Fun11 – Fun13, Fun15, Fun16	Control action	C01 – C05, C09, C10, C12 – C14, C16, C17
Type 2	Main	Fun07, Fun10, Fun17	Feedback	F06, F11, F18
Type 3	Supporting	Fun14	Control action	C15
Type 4	Continuous	Fun19	Supply	S24, S25
Type 5	Continuous	Fun18, Fun20, Fun21	Feedback	F07, F08, F19, F20 – F23

object (the target of the function). For example, Figure 3 shows that *C03. aut. open circuit control action* is performed by *relay & breaker* (subject) to the *high voltage system* (object). Due to the association of function to subject and object, it is possible to have several agents performing the same function. For example, *high voltage system* and *relay & breaker* has responsibility to maintain power supply function (represented as *S24 supply power* and *S25 maintain power supply*). These comparable functions are only modeled as a single function *Fun18 supply power* in the FAST diagram. In HCS, it is possible to model the influence from the environment (anything outside the system boundary) to the system in the HCS by modeling it as a box performing a function to the agent.

The FAST diagram and the HCS classified the functions into different types. In the FAST diagram, each function is classified either as a main, a supporting, or a continuous function. In the HCS, each function is classified either as a control action, a feedback, or a supply function. Since the analyzed system is the same, it is possible to map every function's classification between the FAST diagram and the HCS. The summary of the mapping is listed in Table 7. For example, the function *stop pump* is classified as the main function *Fun02* in the FAST diagram and as control action *C01* in the HCS (type 1). In another example, the function *detect abnormal level* is a main function *Fun07* in the FAST diagram and is a feedback *F06* in the HCS (type 2). This mapping is unique for this equipment protection system and may be different depending on the investigated system.

For the process flow, it is clear how every function's sequential process is modeled in the FAST diagram. The horizontal (left-right) sequence shows how the function to the right of the selected function is the causative function, while the function to the left is the reactive function. In contrast, HCS models the hierarchy in a vertical (top-down) relation. It depicts how one controller has higher authority than the agent (e.g., another controller or controlled process) at the lower hierarchy level. This vertical hierarchy does not show the system operational process (i.e., the starting, the preceding, the following, and the finishing point).

To understand the HCS (i.e. in Figure 3), it is necessary to read the controller responsibility and process model at any given point (e.g. in Table 4). For example, *MCS*'s responsibility as the controller is to provide *C05 aut. command pump shutdown* to the *PSD node*. From the HCS, *PSD node* has two output pathways, *C04 proceed shutdown command*, or *F08 provide process information*. From the Table 4, *C04* has a process model *shutdown command status* that indicates *PSD node* responsibility to pass the shutdown command to the *relay & breaker*. In contrast, *F08* shows *PSD node* responsibility to provide feedback information to the *operator*. F08 is not consistent with the control action *C05*. It is more logical to have *C04* as the following operational sequence after *C05*. This way of reasoning is necessary to gain an understanding of the system process from the HCS. Arguably, for a more complex controller (with a higher number of input/output functions), it would be more difficult to understand the step-by-step sequence of the function for people who never looked into the system before the analysis.

These three differences between the modeling of the FAST diagram and HCS may affect the latter hazard analysis process that will be discussed in the later section.

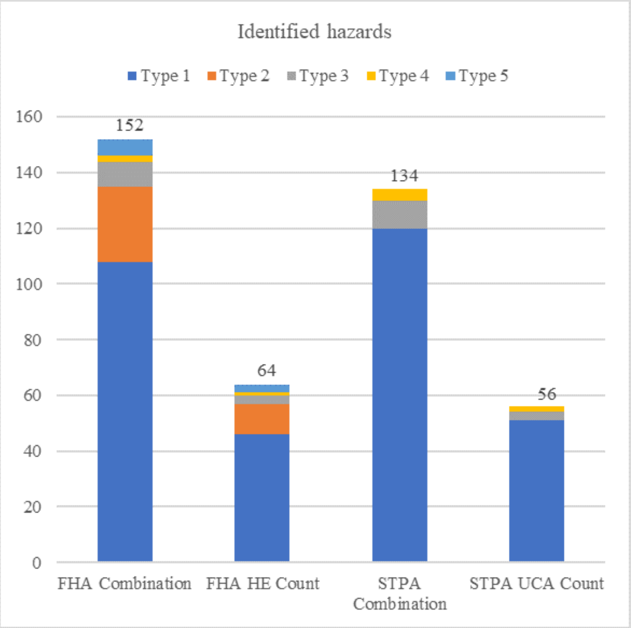


Figure 4. Comparison between the number of the assessed combination from FHA and STPA process and the identified HEs and UCAs (types refer to Table 7)

Comparison of the hazardous events and unsafe control actions

Figure 4 shows statistics of the identified HEs and UCAs from the pump protection system. It appears that FHA captured a higher number of HEs than STPA did with UCAs. It is due to three reasons: the use of keywords for hazard identification, the function type classification in the selected model, and the modeling approach.

The keywords comparison can be seen in Table 8. Overlapping keywords result in the identification of the same type of HEs and UCAs. For example, *HE001* (in Table 5) and *UCA001* (in section *STPA result*) are inherently the same

Table 8. Comparison between failure mode & UCA types

Type of failure mode	Type of UCA
Omission error	Not provided (when needed)
Commission error	Provided (when not needed)
Late	Provided too late
Early	Provided too early
-	Stopped too soon
-	Applied too long
Value	-

type of hazardous events. However, for keywords such as stopped too soon, applied too long, and wrong value with no comparable guidewords in the other methods (the two former keywords for STPA and the following keywords for FHA), the hazards identified by utilizing these keywords were unique to the particular method. For example, STPA did not identify UCA similar to *HE043*, while FHA did not identify HE similar to *UCA026* and *UCA027*.

Second, as mentioned during modeling technique comparison, some functions are classified differently between the FAST diagram and HCS. While it does not affect FHA's hazard analysis, the classification affected the identification process of STPA. STPA considers hazardous events as unsafe control actions. It results in a limitation of the hazard identification process only to include the control action and the supply functions. Therefore, functions classified as type 2 and type 5 from Table 7 are analyzed for possible HEs in FHA, while not analyzed for possible UCAs in STPA. Figure 4 shows that there is no orange-colored box (type 2) and light blue-colored box (type 5) in both the STPA combination and the identified UCAs.

Finally, how the FAST diagram and HCS approached to model the system also contributed to the number of identified hazards. Some functions can be performed by several agents (i.e., one function in the FAST diagram can be two or more functions in HCS). This modeling approach increased the number of identified hazards in the STPA. For example, analysis of *S24 supply power* and *S25 maintain power supply* in STPA resulted in two UCAs, while analysis of *Fun18 supply power* in FHA resulted in one HE.

Comparison of the consequences and system-level losses

The identified consequences and System-level Losses for FHA and STPA are the same: equipment damage and unnecessary loss of production. However, they were derived differently. In FHA, consequences are assessed as a possible effect of HEs (inductive technique). Comparatively, the loss results in STPA were identified at the start as unwanted loss caused by a system-level hazard that needs to be avoided (obtained either from past experiences with a similar system or from standards and regulations). These Ls became the starting point for deductive analysis in STPA to identify UCAs. This different perspective affects the analysis boundary.

The identified Losses in STPA limit the boundary of the analysis to the pre-described system-level hazards and losses. The focus of STPA was then to determine what type of possible UCAs can result in the pre-described Losses. That culminated in comprehensive top-down traceability from the Loss – Hazards – UCAs – LScs (shown in the inclusion of various IDs resulting from UCAs). Arguably, it is tough to have a complete list of unwanted Ls and Hs from the start of the analysis, especially if it is performed for novel technology. When encountering this problems, Leveson[26] recommended to start the analysis at a higher level of abstraction. This, however, caused the resulting list of Ls and Hs to be too generic and necessitates an iteration process to ensure completeness. The top-down method of STPA shows its limitation when there are omitted system-level losses or system-level hazards. In this condition, it is necessary to redo the analysis from the start to check whether there are any omitted UCAs or LScs from the analysis. This problem soon becomes unmanageable for a larger and complex system. In comparison, The FHA process is not limited by the identified consequences. When there is a change in the system, what needs to be done is to check whether the identified hazard's implication results in the same/different consequence.

Comparison of the causal scenarios and loss scenarios

Table 9 presents a statistic of the analyzed CaS and LSc by FHA and STPA for the pump protection system. It shows that in contrast to the higher number of identified HEs than UCAs, the number of identified LScs is significantly higher than the CaSs. This is caused by how the utilized model aids the causal scenario analysis process and the availability of other relevant information.

In FHA, all HEs are associated with a function. The analysis scope of the CaS from the FAST diagram is limited to the respective function and its immediate connection, as shown in Figure 5. Comparatively, the LScs are analyzed from their associated control loops that provide the UCAs. A control loop includes all necessary functions to perform the control action function (e.g., actuating, logic solving, and sensing function). Therefore, the analyzed control loop may include several agents that perform different functions, as shown in Figure 6. This results in a higher number of identified loss scenarios. For example, *HE015* and *UCA015* are the same type of functional hazard for a logic solver *MCS*. Causal scenario analysis of *HE015* identified seven distinct CaSs, while loss scenario analysis of *UCA015* identified eleven distinct LScs. FHA's causal scenario analysis process was only able to find scenarios related to the *MCS* (that perform logic solving function) and *topside* and *subsea communication* (that transfer the function from the previous function and to the following function). STPA's loss scenario analysis process managed to find additional unique scenarios related to the *PSD node* (that perform actuating function) and the *(safety) level sensor* (that performs sensing function).

As described in the procedure of FHA and STPA, both the FAST diagram and HCS models are utilized as the aid for the causal analysis process. The FAST diagram is a model that depicts how every mechanism in the system is connected structurally. It does not specify any components and how they interact with the environment. The analyst must identify the C and E properties (of the CESM model) from other information sources. First, each function is associated with the agent (or composition) performing it. Then conceptual design and operation are utilized to check whether there will

Table 9. Comparison between the number of analyzed scenarios

Type	FHA count	STPA count
Hazard	64	56
Causal Scenario	206	346
– Caused by composition	108	247
– Caused by environment	0	0
– Caused by structure	87	80
– Caused by mechanism	11	19

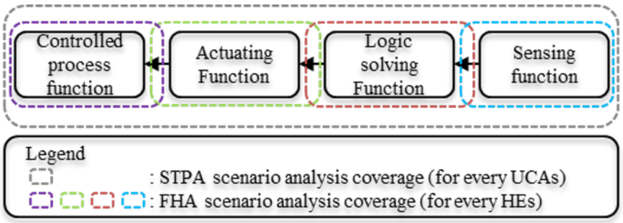

Controlled process function
Actuating Function
Logic solving Function
Sensing function
Legend
: STPA scenario analysis coverage (for every UCAs)
: FHA scenario analysis coverage (for every HEs)

Figure 5. Example of loss scenario analysis perspective comparison based on the FAST diagram

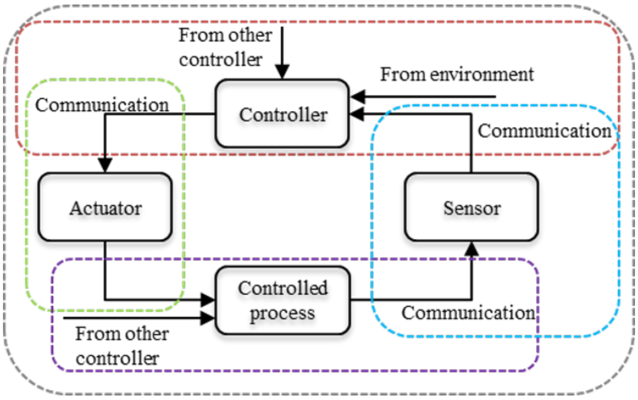


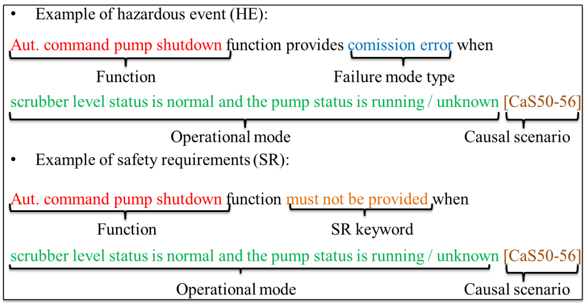
Figure 6. Example of loss scenario analysis perspective comparison based on the HCS

be a process condition (or environmental effect) that may cause a hazard.

In contrast, all aspects of CESM are modeled into the HCS (see Figure 6). An HCS depicts how the Agent (or composition) is connected structurally to each other by performing functions (or mechanism). Influence from the environment can be added to the model to consider the possible implications to the UCA. For example, Table 9 shows the classification of the identified causal scenarios based on the CESM properties. While both techniques cover all the CESM properties for the causal scenario analysis, the HCS provides more help due to the inclusion of all the model properties. It reduces the omission possibility when checking the causal scenarios from several information sources.

Comparison of the safety requirements and controller constraints

FHA derived 64 SRs, while STPA derived 56 CCs. These requirements/constraints are obtained solely from the identified hazards. The SRs and CCs are evaluated based on the eight criteria for requirement[23]: (1) identifiable, (2) clear, (3) solution-specific, (4) have ownership, (5) have origin, (6) verifiable, (7) able to be validated, and (8) have priority.



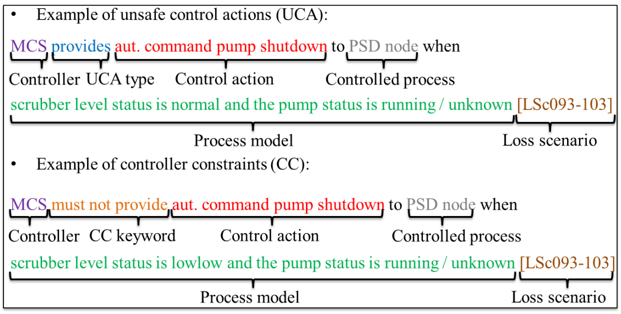
- Example of hazardous event (HE):

Aut. command pump shutdown function provides comission error when scrubber level status is normal and the pump status is running / unknown [CaS50-56]

- Example of safety requirements (SR):

Aut. command pump shutdown function must not be provided when scrubber level status is normal and the pump status is running / unknown [CaS50-56]

Figure 7. Example of key attributes tagging on the HEs and SRs (colors are used to distinguish between the key attributes)



- Example of unsafe control actions (UCA):

MCS provides aut. command pump shutdown to PSD node when scrubber level status is normal and the pump status is running / unknown [LSc093-103]

- Example of controller constraints (CC):

MCS must not provide aut. command pump shutdown to PSD node when scrubber level status is lowlow and the pump status is running / unknown [LSc093-103]

Figure 8. Example of key attributes tagging on the UCAs and CCs (colors are used to distinguish between the key attributes)

(1) *Identifiable*. Both SRs and CCs are derived to ensure the safety of the system. They can be accounted for the hazard analysis process of FHA and STPA. If any changes arise in the system, the listed requirement may not be applicable anymore, depending on the implication of change to the analyzed system.

(2) *Clear*. The derived SRs and CCs achieved this by utilizing the key attributes from the HEs and UCAs to word the requirements/constraints. The examples of the tagging from the key attributes to the generated SRs and CCs are shown in Figure 7 and Figure 8. For example, an SR is identifiable by its composition of function, SR keyword, operational mode, and causal scenario. Similarly, a CC is recognizable by its composition of the controller, CC keyword, control action, controlled process, process model, and loss scenario. The transformation rule from HEs to SRs and UCAs to CCs in Table 2 makes the derived SRs and CCs more evident.

(3) *Solution-specific*. Both FHA and STPA are performed to analyze a specific system. The derived SRs and CCs are only applicable, given the context and scope of the analysis initially defined. Like the first criterion, the SRs and CCs may not be applicable anymore if any changes occur.

(4) *Have ownership*. In the context of systemic hazard analysis of functions, it is best performed during the early design phase. Both SRs and CCs need to be followed by the designer to develop the system's detailed design. The stakeholder may change if the hazard analysis is performed at the different phases of design.

(5) *Have origin*. In FHA, the SRs subject is the function itself. It does not specify which component (system or subsystem) needs to follow the requirement. It allows the decision-maker to assign any agent that needs to carry out the

functions. In STPA, the CCs subject is a specific controller (see Figure 8) that needs to be constrained. If the control action is assigned to a different agent, the initial requirement does not apply anymore. Another STPA process needs to be done to check whether additional CCs are required for the new agent.

(6) *Verifiable*. If the requirement is too detailed and technical, the requirements may not be satisfied by the new technology and limit the options for solutions. Both SRs and CCs are functional requirements. They do not limit the possible solution as long as it is possible to achieve the required functionality. The causal/loss scenarios can be used as guidance to satisfy the requirements (e.g., by identifying the barrier to eliminate, prevent, or mitigate the scenarios). Both SRs and CCs also need to be checked against the system's original functional requirement. There may be conflicting requirements due to the different perspectives in the initial requirement (e.g., between achieving safety of the system or availability of the production).

(7) *Able to be validated*. In this paper, the SRs and CCs are qualitative requirements. It is difficult to justify whether the derived requirements can be achieved or not given the current form of the SRs and CCs (without measurable criteria). FHA is originally a semi-quantitative hazard analysis tool. Typically, a risk assessment process is integrated into the FHA process (see section *FHA procedure*) to validate the requirements (e.g., by quantifying the effects of risk reductions and checking them against the risk criteria). In contrast, STPA is originally not supported by quantitative measures due to Leveson's skepticism with the individual number assignment (e.g., for likelihood assessment)[1]. Only recently that Kim et al.[31] proposed a semi-quantitative approach for risk analysis with STPA.

(8) *Have priority*. In this paper, we do not perform any prioritization of the safety requirement. As discussed previously, the semi-quantitative measures are also used to prioritize essential requirements in both FHA and STPA (although still need further research for the latter method).

Conclusion of the comparison analysis

Table 10 provides a summary of the comparison results. To assess the implication, we need to bring the results one step higher and reflect on the analysis to answer the RQs and our initial objective. Based on the analysis to answer the RQ1, the comparison indicates that both methods are similarly suitable for analyzing novel technology. It is unnecessary to utilize FHA and STPA simultaneously since they capture similar types of functional hazards and scenarios. From the analysis to answer the RQ2 and RQ3, STPA has two advantages over FHA: the modeling technique captures all four systemic properties, and the safety requirements structure complies with more criteria of a requirement.

If looking into the system model, the STPA's modeling technique captures all four systemic properties of a system, while FHA's modeling technique can only capture two systemic properties. This makes the causal analysis process of STPA easier than FHA due to the latter's need to refer to other documents/models for support. From the criteria of a requirement, we identify that every safety requirement in STPA has been assigned to an agent (e.g., physical component or human). This makes the safety requirement of STPA ready for use, while an additional process is required in FHA to identify the agent.

Based on the reasons above, we conclude that STPA is more suitable than FHA to analyze novel technology. Due to their focus on functionality, rather than the realization, both methods are theoretically general enough to be used across different application areas. Our recommendation is valid in the process industry, as demonstrated in this study, and in the aerospace industry[19], where FHA is the recommended methods[4]. STPA demonstration in other industrial applications, e.g., medical[29,36], and maritime[37], indicates its versatility across different subject areas and implies that our recommendation can be relevant as well.

Overall implications

This section discusses the implications of the findings with several topics in the risk management area.

Insights into hazard analysis methods

The comparison analysis highlights the differences between FHA and STPA procedures that can be used as lessons learned to improve both methods. For example, we found several unique hazards to FHA and STPA due to the different keywords used by each method. FHA and STPA may increase hazards coverage by borrowing the missing keywords (refer to Table 8) and used them for the identification of hazardous events or UCAs.

In STPA, the feedback functions are not considered for UCAs' identification, which results in a lower number of the hazards. Error in the feedback functions (e.g., detection error) are later identified as possible scenarios that lead to the UCA (see discussion on causal scenario comparison). Therefore, there is a lower risk of omission by not considering the feedback functions as UCA. It is not necessary to modify the STPA procedure based on this issue.

For the modeling technique, HCS captures more systemic properties of CESM than the FAST diagram. In FHA, this is complemented by analyzing the remaining properties from other information sources. Utilizing the HCS model (or similar model that captures CESM properties in a single model) during the FHA's causal analysis process would reduce the omission possibility of relevant scenarios.

The analysis using STPA in our study case produces a significantly higher number of causal scenarios than FHA. Most of the scenarios found by STPA are caused by similar causal factors that are redundant with FHA causal factors. When analyzing the type of causal factors in scenarios, we found that both methods still suffer the same limitation for identifying either scenario due to a single point causal factor (e.g., component failure or software error) or known scenario (due to simple interaction). This is contrary to Leveson et al.[19] findings that analysis using STPA could find causal scenarios that could not be found by analysis using FHA. Currently, both hazard analysis methods still rely heavily on expert judgment and historical experiences. For novel technology involving complex software-intensive systems, experts and experiences' advantages are lower (due to limited information). Having a systemic perspective when analyzing the causal scenarios does not imply capturing systemic causal scenarios. We conclude that the systemic

Table 10. Summary of the comparison analysis

FHA vs. STPA steps	FHA	STPA
FAST diagram vs. HCS	· Model system as structured functional diagram · No information on the subject that performs the function (system or subsystem) · No information on interaction with the environment · Function is classified as main, support and continuous · Function sequence is clear	· Model system as a structured control loop · Every function has a subject (that perform the function) and object (target of the function) · Possible to model effect on the environment · Function is classified as control, feedback, and supply · Control sequence is ambiguous · One function can be performed by several agents (e.g., for pass-through of function)
Hazardous Events (HE) vs. Unsafe Control Actions (UCA)	· Type of failure mode: omission error, commission error, late, early, value · Analyze every function as possible HEs · When analyzing comparable functions and utilizing similar keywords, both methods find the same type of hazards	· Type of UCA: not provided (when needed) provided (when not needed), provided too late, provided too early, stopped too soon, applied too long · Does not analyze feedback function as possible UCAs (not a control action). · When analyzing comparable functions and utilizing similar keywords, both methods find the same type of hazards
Consequence vs. System-level Loss	· Identified as result of HEs (bottom-up approach)	· Need to be defined at first (top-down approach) · Boundary of the analysis
Causal Scenario (CaS) vs. Loss Scenario (LSc)	· The analyzed scenarios are considered from the hazardous function and its interaction with connecting function · Need additional information sources to assess the compositional (e.g. component failure) and environmental problem (e.g. pressure influence)	· The analyzed scenarios are considered from every function in the control loop · The compositional, environment, structure and mechanism properties of a system is included into single HCS model
Safety Requirement (SR) vs. Controller Constraint (CC)	· Does not have an agent as the subject for a requirement · Has established procedure to apply criteria for validation and prioritization	· Have an agent as the subject for a requirement · Originally, do not include criteria for validation and prioritization. Recent works indicate the attempt for prioritization.

model used by STPA does not have an advantage over the complex linear model used by FHA. Therefore, a procedure to analyze systemic scenarios caused by multiple point problems and unknown scenarios is required.

Insights into the cautionary principle

Both FHA and STPA generate qualitative requirements with equal weight for all the identified requirements. This is in line with the cautionary principle that *if the consequences of an activity could be serious or subject to uncertainties, then cautionary measures should be taken and, or the activity should not be carried out*[38,39]. However, according to Kim et al.[31], equal weight does not provide decision-making support. For example, our case studies with FHA and STPA identified many scenarios for a small system with only 12 components. Without prioritization, the decision-maker would not be able to select the most critical requirements as resources and time for implementation are limited.

When the application is safety-critical, risk and reliability requirements are applied to the functions (not treated in the paper). The requirements (expressed as safety integrity level requirements) stem from as low as reasonably practicable principle. Some functions may not be implemented with the same integrity (or having different priority levels). If the requirements have low priorities (e.g., having minimal consequences, or less uncertainty), they can be assumed to have an insignificant impact on the system. Thus, not conflicting with the cautionary principle. While risk analysis is already an established practice in FHA, the experience with risk analysis on STPA is low. Kim et al.[31] approach has a risk of screening out essential scenarios. Further research is vital to improve and validate the latter approach with other study cases.

Insights into the uncertainty management

Assumptions used during the hazard analysis process imply that the result's validity may have uncertainties. For example, during the modeling process, the analyst would have an initial preconception of the system behavior. Karanikas[40] presented at least ten types of assumptions during each step of the STPA procedure. Similarly, assumptions are used when performing FHA for our case study.

We found that Both FHA and STPA do not guide on communicating both assumptions and their uncertainties sufficiently. The results of FHA and STPA only covers what were considered as hazardous scenarios and did not record what was not found (e.g., safe scenarios). The omission of such results may be dangerous as the latter scenarios' assumptions may deviate (due to uncertainties) and result in the need to consider such scenarios as hazardous. Bjerga et al.[41] recommended performing a separate assessment to analyze the implication of uncertainty in the assumption in their work for treating uncertainty in risk analysis of a complex system. The recommended methods, the assumption-deviation risk[42], may also be useful for the management of uncertainties in both FHA and STPA. Here, the strength of knowledge in every assumption is assessed for the risk of deviation. Suppose the deviations have a significant impact on the analysis results, an additional study may be performed on the assumptions to increase the strength of knowledge and minimize the deviation effect.

Insights into the resilience management

Resilience refers to the system's ability to react and recover from disturbances[43]. In the context of safety, we focus only on the disturbances that may cause losses. The safety requirements of FHA and STPA are limited on the prevention of hazardous events (i.e., react part of resilience). We found that there is a lack of attention on managing the resilience if the hazardous events (leading into losses) (i.e., react part of resilience) and the consequences do occur (i.e., recover part of resilience). Even if the safety requirements of FHA or STPA are fulfilled, there is no guarantee that this would result in a perfectly safe system. Thus, management on the missing part mentioned above would be necessary to increase the system's safety.

For the hazardous events leading to the losses part, we may learn from the process to generate the safety requirements in FHA and STPA. We suggest two-step procedures. The first is to add safety requirements to prevent or mitigate the losses. For example, in FHA, the requirements may be formed as follows, *hazardous event #xx should not lead to consequence #xx*. Similar requirements can be formed for STPA by changing the hazardous event with UCA. Then, FHA or STPA may be coupled with consequence assessment methods (e.g., cause-consequence diagram[5] or event tree analysis[5,6]). The later step is similar to the causal analysis procedure, where the results are attached to the requirements as guidance for scenarios that need to be prevented.

For the consequences part, we suggest forming the requirements to recover from the consequences. For example, *system should be able to recover from consequence #xx*. The requirement allows the decision-maker to formulate the recovery approach specific for each system condition.

Conclusion

This paper has carried out a systematic comparison of FHA and STPA with a case study from an equipment protection system in the oil and gas industry. We compared each step of the analysis individually (i.e., during system modeling, hazard identification, consequence identification, causal scenario analysis, and safety requirement generation). We have analyzed how each process is beneficial to identify functional hazards, provide a systemic perspective of the system, and generate safety requirements. This study found that STPA is more suitable than FHA to analyze the investigated system due to the advantages of the modeling technique used and the format of safety requirements that it generated. Recommendations are provided to improve FHA and STPA based on the lessons learned from each method. If the recommendations are implemented, FHA and STPA may have a similar level of capability and may replace each other. Obvious that this finding differs with other research claim where STPA is shown to be significantly better than the compared methods[19] or is required as supplementary methods[20]. Further works by investigating other system with different functionalities and complexities are required to verify our claim.

The selection of STPA (or FHA as an alternative) requires further works to be aligned with the functional safety standard, i.e., IEC 61508. We have discussed our insights related to the cautionary principle, uncertainty management, and resilience management as guidance for further works. First, the risk assessment process for prioritization of STPA results needs to be validated. Second, there is a need to investigate the assumption-deviation risk method to manage uncertainty in the FHA or STPA results. Another future work is to test and validate the suggested procedures for improving the resilience management in FHA and STPA.

Acknowledgements

This paper has been written under the Safety 4.0 project. The author(s) would like to thank the Research Council of Norway, industrial and university partners involved in this project for the support. For more information about the project, see https://www.dnvgl.com/research/oil-gas/safety40/index.html

The author(s) are grateful for the two anonymous reviewers' comments and insights into the earlier version of the paper.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: The Research Council of Norway has partly funded this work as part of the Petromaks 2 program [grant number 281877/E30].

References

1. Leveson N. *Engineering a safer world: systems thinking applied to safety*. Engineering systems, Cambridge, MA: The MIT Press, 2011. ISBN 978-0-262-01662-9.
2. Hollnagel E. *FRAM, the functional resonance analysis method: modelling complex socio-technical systems*. Ashgate, 2012. ISBN 1-351-93596-8.
3. IEC 61508, Functional safety of electrical/electronic/programmable electronic safety-related systems – part 1-7. Standard, International Electrotechnical Commission, 2010.
4. ARP 4761, Guidelines and methods for conducting the safety assessment process on airborne systems and equipments. Standard, SAE, The Engineering Society for Advancing Mobility Land Sea Air and Space, 1996.
5. Ericson CA. *Hazard analysis techniques for system safety*. 2nd ed. Hoboken, New Jersey: Wiley, 2016. ISBN 1-119-10170-0.

6. Rausand M. *Risk assessment: theory, methods, and applications*. Statistics in practice, Hoboken, N.J.: Wiley, 2011. ISBN 1-118-28111-X.
7. Alexander IF and Maiden N. *Scenarios, stories, use cases: through the systems development life-cycle*. John Wiley & Sons, 2005.
8. Wilkinson P and Kelly T. Functional hazard analysis for highly integrated aerospace systems. *IET Conference Proceedings* 1998; : 4–4(1).
9. Zikrullah NA, van der Meulen MJP, Kim H et al. Clarifying implementation of safe design principles in IEC 61508: Challenges of novel subsea technology development. In *Proceedings of the 29th European Safety and Reliability Conference (ESREL)*. Research Publishing. ISBN 978-9-811-12724-3, pp. 2928–2936.
10. Raspotnig C and Opdahl A. Comparing risk identification techniques for safety and security requirements. *The Journal of Systems & Software* 2013; 86(4): 1124–1151.
11. Raheja D. Software FMEA: A missing link in design for robustness. Technical report, SAE, The Engineering Society for Advancing Mobility Land Sea Air and Space, 2005.
12. ISO 26262, Road vehicles — Functional safety – part 1-12. Standard, International Organization for Standardization, 2016.
13. IEC 61511, Functional safety - Safety instrumented systems for the process industry sector – part 1-4. Standard, International Electrotechnical Commission, 2016.
14. Ishimatsu T, Leveson NG, Thomas JP et al. Hazard analysis of complex spacecraft using systems-theoretic process analysis. *Journal of Spacecraft and Rockets* 2014; 51(2): 509–522.
15. Yousefi A, Rodriguez Hernandez M and Lopez Peña V. Systemic accident analysis models: A comparison study between AcciMap, FRAM, and STAMP. *Process Safety Progress* 2019; 38(2).
16. de Carvalho EA, Gomes JO, Jatobá A et al. Employing resilience engineering in eliciting software requirements for complex systems: experiments with the functional resonance analysis method (fram). *Cognition, Technology & Work* 2020; : 1–19.
17. ISO/PAS 21448, Road vehicles–safety of the intended functionality. Standard, International Organization for Standardization, 2019.
18. Pumfrey DJ. *The principled design of computer system safety analyses*. PhD Thesis, University of York, 1999.
19. Leveson N, Wilkinson C, Fleming C et al. A comparison of STPA and the ARP 4761 safety assessment process. Technical report, MIT PSAS, 2014.
20. Sulaman SM, Beer A, Felderer M et al. Comparison of the FMEA and STPA safety analysis methods—a case study 2019; 27(1): 349–387.
21. Bunge M. *Emergence and Convergence: Qualitative Novelty and the Unity of Knowledge*. Toronto: University of Toronto Press, 2003. ISBN 978-1-4426-2821-2.
22. Wan PYz. Emergence à la systems theory: epistemological totalausschluss or ontological novelty? *Philosophy of the Social Sciences* 2011; 41(2): 178–210.
23. Holt J, Perry SA and Brownsword M. *Model-Based Requirements Engineering*. The Institution of Engineering and Technology, 2011. ISBN 978-1-849-19487-7.
24. Allenby K and Kelly T. Deriving safety requirements using scenarios. In *Proceedings Fifth IEEE International Symposium on Requirements Engineering*. IEEE. ISBN 0-7695-1125-2, pp. 228–235.
25. ASTM E2013-12, Standard practice for constructing FAST diagrams and performing function analysis during value analysis study. Standard, ASTM International, 2012.
26. Leveson N and Thomas J. *STPA handbook*. 2018.
27. Thomas IV JP. *Extending and automating a systems-theoretic hazard analysis for requirements generation and analysis*. PhD Thesis, Massachusetts Institute of Technology, 2013.
28. Rachman A and Ratnayake RC. Implementation of system-based hazard analysis on physical safety barrier: A case study in subsea HIPPS. In *2015 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*. IEEE, pp. 11–15.
29. Masci P, Zhang Y, Jones P et al. A hazard analysis method for systematic identification of safety requirements for user interface software in medical devices. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 10469. Springer Verlag. ISBN 978-3-319-66196-4, pp. 284–299.
30. Kim H, Lundteigen MA, Hafver A et al. Application of systems-theoretic process analysis to a subsea gas compression system. In *Safety and Reliability – Safe Societies in a Changing World – Proceedings of the 28th International European Safety and Reliability Conference, ESREL 2018*. CRC Press/Balkema. ISBN 978-0-815-38682-7, pp. 1467–1476.
31. Kim H, Lundteigen MA, Hafver A et al. Utilization of risk priority number to systems-theoretic process analysis: A practical solution to manage a large number of unsafe control actions and loss scenarios. *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability* 2020; : 1748006X2093971DOI:10.1177/1748006X20939717.
32. Kim H, Lundteigen MA, Hafver A et al. Application of system-theoretic process analysis to the isolation of subsea wells: Opportunities and challenges of applying STPA to subsea operations. In *Offshore Technology Conference*. Offshore Technology Conference.
33. Vinterstø T, Birkeland B, Ramberg RM et al. Subsea compression–project overview. In *Offshore Technology Conference*. Offshore Technology Conference.
34. DNV GL. Safety 4.0, 2018. Retrieved 2020−08−20. URL : https://www.dnvgl.com/research/oil-gas/safety40/index.html.
35. API RP 17V, Recommended practice for analysis, design, installation, and testing of safety systems for subsea applications. Standard, American Petroleum Institute, 2015.
36. Antoine B. *Systems Theoretic Hazard Analysis (STPA) applied to the risk review of complex systems: an example from the medical device industry*. PhD Thesis, Massachusetts Institute of Technology, 2013.
37. Rokseth B, Utne IB and Vinnem JE. Deriving verification objectives and scenarios for maritime systems using the systems-theoretic process analysis. *Reliability Engineering & System Safety* 2018; 169: 18–31.
38. Aven T and Renn O. Improving government policy on risk: Eight key principles. *Reliability Engineering & System Safety* 2018; 176: 230–241.
39. Aven T. The cautionary principle in risk management: Foundation and practical use. *Reliability Engineering & System Safety* 2019; 191: 106585.
40. Karanikas N. Documentation of assumptions and system vulnerability monitoring: The case of system theoretic process

analysis (stpa). *International Journal of Safety Science (IJSS)* 2018; 2(1): 84–93.

41. Bjerga T, Aven T and Zio E. Uncertainty treatment in risk analysis of complex systems: The cases of stamp and fram. *Reliability Engineering & System Safety* 2016; 156: 203–209.
42. Aven T. Practical implications of the new risk perspectives. *Reliability Engineering & System Safety* 2013; 115: 136–145.
43. Hollnagel E, Woods DD and Leveson N. *Resilience engineering: concepts and precepts*. Ashgate Publishing, Ltd., 2006.