A Comparison of Hazardous Scenarios in Architectures with Different Integration Types

Nanda Anugrah Zikrullah

Department of Mechanical and Industrial Engineering, Norwegian University of Science and Technology, Norway. E-mail: nanda.a.zikrullah@ntnu.no

Meine J.P. van der Meulen

Group Technology and Research, DNV GL, Norway. E-mail: meine.van.der.meulen@dnvgl.com

Gunleiv Skofteland

Process Technology Automation, Equinor, Norway. E-mail: gusk@equinor.com

Mary Ann Lundteigen

Department of Engineering Cybernetics, Norwegian University of Science and Technology, Norway. E-mail: mary.a.lundteigen@ntnu.no

Whether or not to allow some integration between process control and safety systems has been an ongoing debate amongst safety researchers and practitioners. The principle of keeping it simple and the principle of having segregation between the two systems are often considered as equal. The current trend is that traditional hardware implemented functions are, to an increasing extent, replaced by programmed functions and that control and safety systems rely on standard communication technologies and devices. Despite the goal of having physical segregation, the systems are no longer simple and without dependencies. Some programmable controllers have inbuilt solutions that can logically separate safety and non-safety (software and hardware) functions inside a single programmable system. It is, therefore, of interest to explore if some of these technological advances can have a positive effect on safety compared to the complexity from duplication of hardware required with segregation. Before such alternative design concepts are selected, it is necessary to evaluate if they are as safe as with physical segregation. The main objective of this paper is to identify and compare the hazards and hazardous scenarios for some selected hardware architectures ranging from complete segregation of process control and safety systems to full integration. This analysis applies the Systems-Theoretic Process Analysis (STPA) method, which has been developed to analyze complex and software-intensive systems. The result from the analysis of the selected architectures indicates that having integration will increase the number of possible scenarios leading to hazards. These scenarios may cause both safety and availability losses. This research is part of Safety 4.0, a joint industry project on research-based innovation that aims to develop a framework for safety demonstration of novel subsea technologies.

Keywords: Integration of process control and safety, subsea system, oil & gas industry, hazard analysis, systems-theoretic process analysis, STPA.

1. Introduction

The design of control systems for the subsea oil and gas industry has evolved throughout the years, starting from direct hydraulic systems in the 1960s, until the most recent all-electric systems (Bai and Bai, 2018). This evolution includes partial replacement of mechanical equipment with programmable controllers. Utilization of the latter components may allow various configurations of architectures between the process control and safety (PC&S) systems, i.e., by having different integration types (CCPS, 2016; Zikrullah et al., 2019).

Each architecture presents different kinds of scenarios leading to hazard. Hazard is defined as *a system state or set of conditions that, together with a particular set of worst-case environmental conditions, will lead to an accident (loss)* (Leveson, 2011). According to Leveson (2011), hazard analysis can be described as *investigating an accident before it occurs*.

Systems-Theoretic Process Analysis (STPA) is a new hazard analysis method that has been developed for the analysis of complex and software-intensive systems. While there are other hazard analysis methods for such systems, STPA pro-

Proceedings of the 30th European Safety and Reliability Conference and the 15th Probabilistic Safety Assessment and Management Conference
Edited by Piero Baraldi, Francesco Di Maio and Enrico Zio
Published by Research Publishing, Singapore.
ISBN: 978-981-14-8593-0; doi:10.3850/978-981-14-8593-0

vides several advantages. First, STPA considers hazard as a control problem. It allows the inclusion of scenarios where no failure occurs in the system (e.g., where multiple controllers provide conflicting commands) (Thomas et al., 2012). Also, it can identify interaction problems (e.g., caused by complex dependencies in the systems) (Aps et al., 2017). Theoretically, STPA can be used anytime during the design lifecycle. Leveson (2020) proposes to integrate the model used in STPA during conceptual architecture development where detailed information is not available. For more details on the comparison between STPA and other hazard analysis tools, see a technical report by Teikari (2014).

In the subsea oil & gas industry domain, STPA has been utilized to analyze different control procedures, such as the a integrity pressure protection system (Rachman and Ratnayake, 2015), the isolation of subsea wells (Kim et al., 2018), and the operation of subsea gas compression system (Kim et al., 2018). Zhang et al. (2019) also used STPA for availability assessment in subsea production. However, nobody has applied STPA for systems with different integration types, as presented in this paper. The objective of this paper is to identify and compare the hazards and hazardous scenarios for some selected hardware architectures ranging from complete segregation of PC&S systems to full integration. The goal is to provide a more unobstructed view of the effect of integration on safety.

The remainder of this paper is organized as follows. Section 2 introduces the alternative concept design for PC&S systems, considering the integration type. Section 3 explains the theoretical foundation of STPA. Section 4 presents the study case of this paper. Section 5 presents the analysis results and discusses the findings. The final section identifies essential areas for further work.

2. Design of Process Control and Safety Systems for Subsea Oil & Gas Industry

2.1. *Independence vs. integration*

According to Drogoul et al. (2007), segregation (or independence) should be considered as one of the safe design principles to enhance safety. However, the decision to have independence or integration between process control and safety systems has been an ongoing debate among safety researchers and practitioners (Gruhn and Cheddie, 2006). IEC 61508 (2010) allows sharing the safety and non-safety elements as long as the requirement in part 1, clause 7.4.2.3 is followed. There is a limitation on the maximum safety integrity level that the system can achieve.

The aim of independence is mainly to have freedom from interference when performing the intended function (IEC 61508, 2010). In contrast, applying integration introduces new interactions to the system that can affect its functionality. While there are measures to avoid/prevent unwanted interactions between integrated components, the system may need additional tests, analyses, and operational burdens to achieve the required functional reliability (CCPS, 2016).

In the process industries, both process control and safety systems are typically considered as a separate protection layer to achieve safety. However, integration may remove the contribution of the process control system as a protection layer (CCPS, 2014).

According to CCPS (2016), there are several issues to be addressed before claiming safety for the integrated PC&S system, as follows:

(1.) The functional capabilities to perform the intended functions.
(2.) The integrity of the functional performance.
(3.) The protection against writes.
(4.) The accessibility to control and change the safety functions.
(5.) The barrier against cyber-threats.
(6.) The protection against environmental issues (e.g., temperature or chemical corrosion).

In practice, for some systems (e.g., subsea oil & gas system), the option of complete independence may increase the complexity of the resulting hardware architecture. For a subsea environment with limited accessibility, performing maintenance on this complex hardware architecture may be another operational burden. The development of Commercial of The Shelf (COTS) programmable controllers that provide logical separation between process control and safety may be an alternative to solve this issue. To allow for acceptance of the integration concept for practical applications, there is a need to ensure that the integrated system can achieve the requirement in IEC 61508 (2010) associated with the safety integrity level requirements that have been derived.

2.2. *Integration concept for process control and safety (PC&S) systems*

Generally, the design of PC&S systems considering integration is common to any process industry. A paper by Steinhauser (2019) discusses the integration of PC&S systems in the batch process industry.

The integration concept may be applied to any element in a control loop (e.g., sensor, logic solver, actuator, and communication). It is impossible to include all the solution spaces for integration if one would consider all the possible combinations. CCPS (2016) provides a classification type for PC&S systems, focusing on the logic solver and the communication network. A

Table 1. Breakdown of the proposed integration type classification

CCPS classification	Integration type	Physical components	Programmable logic	Network component	PC&S interaction
Air-gapped systems	A. Complete independence	Separated	Separated	None	None
Interfaced systems	B. Conditional independence	Separated	Separated	Interfaced	Limited by firewall
Integrated systems with isolated networks	B. Conditional independence	Separated	Separated	Separated	Limited by firewall
Integrated systems with shared networks	B. Conditional independence	Separated	Separated	Shared	Limited by firewall
Combined system with strong dependency	C. Partial integration	Shared	Separated	Shared	Limited by logical separation
	D. Complete integration	Shared	Shared	Shared	Depends on the configuration

new classification has been derived to simplify the CCPS classification, as follow:

- *A. Complete independence*. Each system has its control loop without any exchange between each other.
- *B. Conditional independence*. Communication between the PC&S systems (vary depending on the configuration) allows the exchange of information between systems. Independence between the two systems can still be achieved by limiting the access from the process control system to the safety system with a firewall to avoid unintended interactions. In the opposite direction access is not restricted.
- *C. Partial integration*. The integration occurs only at the hardware components of the logic solver (vary depending on the configuration). Logical separation exists between the two systems (IEC 61508 (2010) Part 3 Annex F specifies how to achieve logical separation between software elements on a single computer).
- *D. Complete integration*. Both systems completely share the use of the logic solver. No clear distinction between process control and safety logic in the programmed software.

Table 1 lists the breakdown of each integration type according to the associated components and interaction between PC&S systems. There are three architectures by CCPS (2016) that are combined into the *conditional independence* type of integration due to similarities between the use of network components. The CCPS classification of *combined system with strong dependency* is split into two integration types to distinguish the different types of integration that may exist in the logic solver (e.g., logical separation or shared software space).

3. Systems-Theoretic Process Analysis (STPA)

The STPA processes identify hazards from the variation of the control action that can be unsafe during a particular set of conditions. This hazard may develop further to become (unwanted) losses. Typically, the application of STPA produces a set of safety constraints that may limit the possible system behavior from the unwanted state.

The STPA procedures are, as follows (Leveson and Thomas, 2018):

(i) *Define purpose of the analysis*. The analysis boundary includes the system description, system-level losses, hazards, and safety constraints.
(ii) *Model the control structure*. A hierarchical control structure (HCS) is built based on the system (expected) interactions and behaviors during the predetermined conditions (or available information).
(iii) *Identify Unsafe Control Action (UCA)*. Keywords are used to determine whether every possible control action in the system during a set of worst-case environmental conditions will lead to UCAs. These keywords are (1) control action is provided, (2) not provided, (3) provided too early/too late, and (4) stopped too soon/applied too long.
(iv) *Identify loss scenarios*. These scenarios are developed based on assessing every aspect in a control loop (including, e.g., feedback error, control algorithm flaws, component failures, transmission problem, incorrect actuation, and the combination between them).

4. Study Case

The study case is obtained from a subsea dry gas compressor provided by API RP 17V (2015).

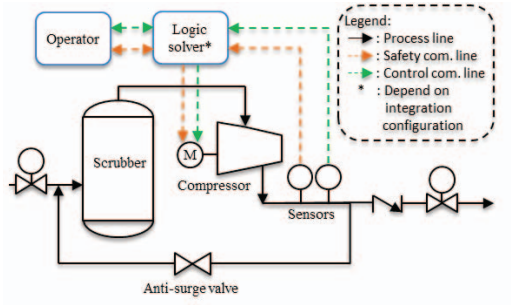

Fig. 1. Process flow diagram of a subsea compression system including process control and safety systems (developed based on API RP 17V (2015))

While it is similar to the study case used in the paper by Kim et al. (2018), this paper distinguished itself by including both process control and safety system in the loop. Also, the analyzed system is later developed at a higher level of abstraction and focuses solely on the analysis of the integrated component. This is because the objective of the paper is to compare the effect of integration, not to perform a full hazard analysis of the compressor.

Figure 1 shows the adapted process flow diagram for a subsea compression system. The system purpose is to compress the gas from the subsea flowline to the topside. Due to compression, there is a possibility to have a high temperature at the outlet of the compressors that needs to be detected by (a set of) sensors. A process control system (PCS) is used to control the compressor speed and maintain the temperature at standard conditions. The information from the sensor is processed by the PCS logic solver to provide an actuation command to the PCS actuator (in this case, a variable speed drive (VSD)). A safety system (SS) used similar information from other sensor(s) at the outlet of the compressor. At very high temperature, the SS logic solver provides a shutdown command to the SS actuator (in this case, a relay and switch) by releasing power to the system. Human operators can provide a command to the compressor through the PCS system. When there is a need for an emergency shutdown, the human operators need to shutdown the power supply system to stop the compressor operation. Human operators may also perform a normal shutdown to the system for maintenance by inhibiting the safety system (for a predefined time) and stopping the compressor manually through the PCS system.

5. Results and discussion

This section presents the selected results from STPA analyses of the subsea dry gas compression systems with different integration types. The discussions are based on thorough comparison between the obtainable results from each step of STPA.

At the start of the analyses, it was required to define the system-level losses, hazards, and safety constraints as the boundary of the analysis. The results are captured in Table 2. The unwanted losses were related to safety issues (environment for SL1 and significant cost for SL2) and availability issues (minor cost for SL3). The boundaries of the analysis were identical for all integration types.

The analysis results for every integration types started to differ at step 2-4 of STPA. The differences are discussed in the following subsections.

5.1. *Hierarchical control structure comparison*

The system descriptions were modelled into HCSs for every integration type as shown in Figure 2-5. Some elements were common in the control loop, e.g., controller (human operator), sensor (PCS and SS sensor), actuator (PCS and SS actuator), and transmission line (the link between elements). Every controller had its process model based on the process information in the system description.

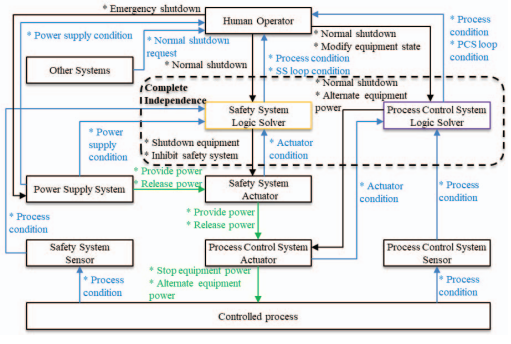

Fig. 2. Hierarchical control structure of type A. complete independence PC&S systems

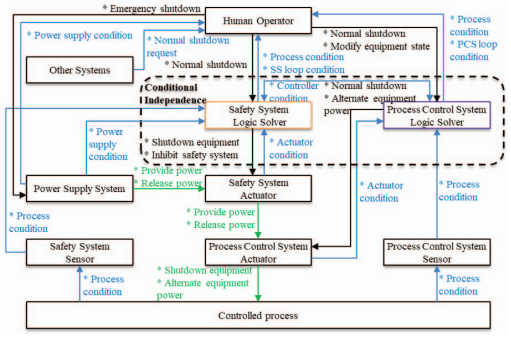

Fig. 3. Hierarchical control structure of type B. conditional independence PC&S systems

Table 2. System-level losses, hazards and safety constraints

System-level Losses (SL)	System-level Hazards (SH)	System-level Safety Constraints (SSC)
SL1. Hazardous material release to the sea	SH1. Loss of containment of hazardous material to the environment	SSC1. Equipment must be able to contain dangerous material from release to the environment
SL2. Damages to valuable equipments	SH2. Equipment operates outside normal operating condition	SSC2. Equipment must be protected from extreme operating conditions
SL3. Unnecessary interruption or reduction in hydrocarbon production	SH3. Equipment operates outside optimal operating condition	SSC3. Equipment must be operated within optimal operating conditions
	SH4. Unintended stop of equipment	SSC4. Equipment must be available to work as intended

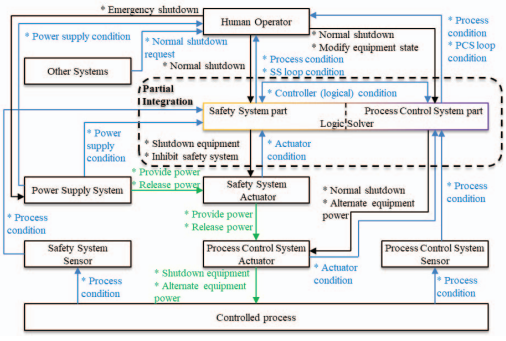

Fig. 4. Hierarchical control structure of type C. partial integration PC&S systems

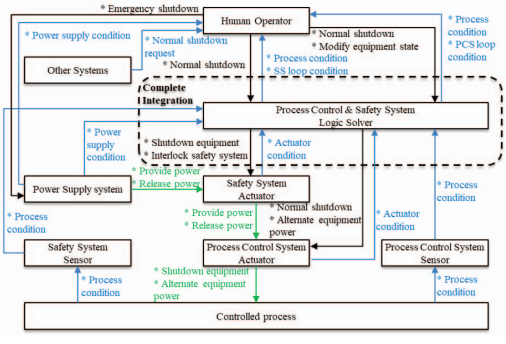

Fig. 5. Hierarchical control structure of type D. complete integration PC&S systems

Every element was connected by the black arrow, which represents the control command, blue arrow, which represents the feedback path, and green arrow, which represents the physical forces and electrical power.

Based on a discussion with experts from the industry, it was assumed that the integration does not change the required control action in the systems. However, information that can be provided through the link between the PC&S systems may be used as a consideration when performing the specified control actions. According to Leveson (2020), the HCS should depict the control structure that does not necessarily reflect their physical architecture (especially during the conceptual operation phase where the architecture is not known yet). However, the HCS can be refined further to include (known) physical interactions in the system.

In this study case, we denote the differences between each HCS by modifying some elements in the HCS within the black dotted box to include the effect of integration. For example, in Figure 4, partial integration in the logic solver was modeled into one single box that was separated by an invisible layer (dotted lines) to show the logical separation between PCS and SS part in the logic solver. The links between elements (e.g., control action or feedback) were still located under the respective part. This model was proposed to show the distinction with the model of separated hardware, as in Figure 2 and 3.

5.2. *Unsafe control action comparison*

The STPA processes managed to identify 46 distinct UCAs. As shown in Figure 6, the number of identified UCAs were identical for all integration types. The possible reasoning is that the attempt to have integration in the system does not inherently

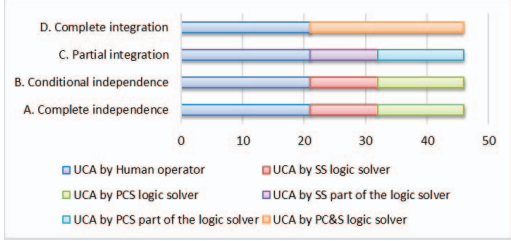

Fig. 6. Number of the UCAs for system with different integration types

change how the control action is performed (as per our initial assumption). Therefore, the development of a control action into unsafe control action during a particular set of conditions is similar for every configuration.

The only difference between the different integration types is the element performing the control action. While there were slight differences in the name of controller for integration type C and D, they were inherently the same controller as the one from integration type A and B (e.g., PCS logic solver, PCS part of the logic solver and PC&S logic solver are the hardware that has the same PCS controller responsibilities). To summarize, human operators contributed the highest number of UCAs (20) as compared to UCAs by PCS (16) and SS (11).

Some examples of UCAs are presented in Table 3. These UCAs correspond only to SH2 (10 UCAs), SH3 (23 UCAs), and SH4 (13 UCAs). More UCAs correspond to availability issues (36 UCAs from SH3 and SH4 that are linked to SL3) than safety issues (10 UCAs from SH2 that is linked to SL2).

5.3. *Loss scenario comparison*

Every UCA was analyzed further to identify the loss scenarios (LSc). Figure 7 shows the number of LScs for all the integration types.

A breakdown of the LScs shows that for all integration types, component failures (70 LScs) contribute the most to the number of scenarios. Erroneous feedback (37 LScs in type A, 40 LScs in type B-D) and human error (35 LScs in type A and 32 LScs in type B-D) provide a significant number of hazardous scenarios. For integration type C and D, unintended interaction (36 LScs) represents new hazardous scenarios that do not exist in the system with integration type A and B. This results in more scenarios in integration type C and D than type A and B.

Some examples of loss scenarios associated with their UCAs are presented in Table 3.

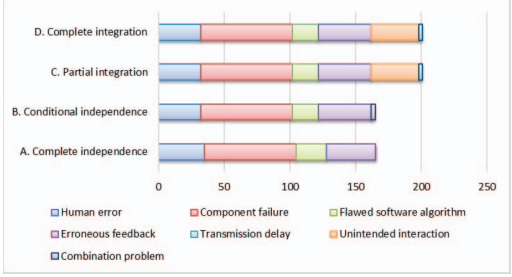

Fig. 7. Number of the loss scenarios for system with different integration types

5.4. *Discussion*

Identification of hazards and hazardous scenarios by STPA on the PC&S system with different integration types provides several useful insights. They are supported by selected examples that are presented in Table 3.

First, from the table, UCA 22 is one example of UCA that is identical for every integration type. It has been previously discussed that integration does not change how each controller should respond to a particular condition. However, having more integration in the system (e.g., in type C and D) may result in different scenarios that can cause hazards. Both A.LSc118.UCA22 and B.LSC120.UCA22 are scenarios that are identical for every integration type. However, C.LSc122.UCA22 and D.LSc121.UCA22 are unique scenarios that may occur only due to the integration of the logic solver hardware.

Second, for a system with integration type A, it has a higher reliance on the human operator for its decision making. In this configuration, there is no direct link between PCS and SS (see Figure 2). The human operator is an essential layer of protection in case of problems in the automated systems. The difference between scenario A.LSc118.UCA22 and B.LSC120.UCA22 shows that human error can be prevented by having an algorithm that can check whether it is possible to select the prohibited command (during a particular condition). For a system with integration type B-D, the algorithm can be implemented due to the ability to communicate directly between the two systems to allow PCS/SS condition check.

Third, there is no difference between the identified scenarios for a system having integration type C and D. One main reason is that having logical separation does not mean that the possibility of unintended interaction is removed. It just means that the engineers have, to the best of their ability, defined and limited the possible interaction paths. This issue will affect the safety demonstration process later (based on the produced safety constraints). Arguably, the system with integration type C has an easier safety demonstration process than type D due to clear separation between the PCS and SS logical architecture.

Fourth, the availability issues are more apparent when the system has more integration (type C and D). On scenarios C.LSc199.UCA35 and D.LSc199.UCA35, component failure of the shared hardware still represents a possible and major scenario leading to hazards. Countermeasures such as redundancy are vital to ensure that the availability of the systems is achieved. For the system with integration type A and B, the availability issues are shared by both PCS and SS logic solver (with possible redundancy on both controllers).

Table 3. Examples of UCAs and LScs for every integration type

Integration type	UCAs	Loss scenarios
A. Complete independence	**A.UCA22** SS logic solver provides shutdown equipment command to SS actuator too late when the gas temperature is very high and the compressor is running **[SH2]**	**A.LSc118.UCA22** Problem in the transmitted information (e.g., due to delay) prevents immediate response by the logic solver
	A.UCA25 SS logic solver does not provide shutdown equipment command to SS actuator when there is normal shutdown request, the compressor is running, and the PCS condition is not ok **[SH3]**	**A.LSc131.UCA25** During this condition it is necessary for the human operator to provide shutdown command from the SS instead of SS inhibition command. However, human error (e.g., due to wrong procedure or no feedback information) prevents the provision of such command
B. Conditional independence	**B.UCA22** SS logic solver provides shutdown equipment command to SS actuator too late when the gas temperature is very high and the compressor is running **[SH2]**	**B.LSc120.UCA22** Algorithm flaw in the SS logic solver (e.g., due to timer or conditional algorithm) increases the processing time to provide the required response
	B.UCA25 SS logic solver does not provide shutdown equipment command to SS actuator when there is normal shutdown request, the compressor is running, and the PCS condition is not ok **[SH3]**	**B.LSc135.UCA25** During this condition it is necessary for the human operator to provide shutdown command from the SS instead of SS inhibition command. A combination failure due to human error (that provide wrong response) and flaws in the software algorithm (that should have prevent the availability to choose inhibition command) may cause the UCA
C. Partial integration	**C.UCA22** SS part of the logic solver provides shutdown equipment command to SS actuator too late when the gas temperature is very high and the compressor is running **[SH2]**	**C.LSc122.UCA22** Resource sharing problem on the hardware delays the execution time of the command
	C.UCA35 PCS part of the logic solver provides normal shutdown command to PCS actuator when there is no normal shutdown request and the compressor is running **[SH4]**	**C.LSc199.UCA35** Component failure of the shared logic solver may move the system to a safe state
D. Complete integration	**D.UCA22** PC&S logic solver provides shutdown equipment command to SS actuator too late when the gas temperature is very high and the compressor is running **[SH2]**	**D.LSc121.UCA22** Unintended overwrites from the PCS to SS part in the logic solver delays the proper command from SS part to the system
	D.UCA35 PC&S logic solver provides normal shutdown command to PCS actuator when there is no normal shutdown request and the compressor is running **[SH4]**	**D.LSc199.UCA35** Component failure of the PC&S logic solver may move the system to a safe state

Finally, STPA provides help when addressing some issues (listed by CCPS) before claiming safety for the system with integration. For example, it provides us information on what scenarios that can hinder the functional capabilities of the system to perform its intended function. Other issues such as protection against writes, accessibility to control and change the safety function, barrier against cyber-threats, and the protection against environmental issues are covered by assessing whether these issues may lead to possible loss scenarios at a detail level (for every UCA). However, due to the qualitative nature of its analysis, STPA provides little help to identify whether the integrity of the functional performance has been achieved or not.

6. Conclusion and further work

This paper has discussed the application of STPA for analysis of various architectures of process control and safety system in the subsea oil & gas industry with different integration types. The discussion has been made on selected results and observable findings from the analysis.

One of the main takeaway from the analysis is that applying more integration to the PC&S

will change how the system behaves and results in more scenarios that can lead to both safety and availability losses. The designer of such a system needs to prepare countermeasures to avoid or prevent the occurrence of the scenarios.

Furthermore, the analysis of loss scenarios in STPA still has a heavy reliance on the knowledge about the possible system behavior and its assumptions. This problem, however, is similar to other hazard analysis methods. For the knowledge of the system, it is recommended that the system follows a technology qualification plan to let the analyst have more experience with the system. For the assumptions, a procedure to check and update assumptions (including the affected analysis results) during the later development of the system is needed.

Finally, the produced safety constraints from STPA should be used as guidance for the safety demonstration process. Demonstration of safety against complex scenarios in a software-intensive system still proves to be a major problem even if the possible scenarios are known. Focus on the hardware in the loop tests or digital twin procedures to provide evidence is needed as an essential research area for development.

Acknowledgement

This work has been written under the Safety 4.0 project that has been partly funded by the Research Council of Norway as part of the Petromaks 2 programme [grant number 281877/E30]. The author would like to thank the Research Council of Norway, the industrial, and university partners involved in this project for the support.

References

API RP 17V (2015). Recommended practice for analysis, design, installation, and testing of safety systems for subsea applications. Standard, American Petroleum Institute.

Aps, R., M. Fetissov, F. Goerlandt, P. Kujala, and A. Piel (2017). Systems-theoretic process analysis of maritime traffic safety management in the gulf of finland (Baltic Sea). *Procedia Engineering 179*, 2–12.

Bai, Y. and Q. Bai (2018). *Subsea engineering handbook*. Gulf Professional Publishing.

CCPS (2014). *Guidelines for Initiating Events and Independent Protection Layers in Layers of Protection Analysis*. John Wiley & Sons.

CCPS (2016). *Guidelines for Safe Automation of Chemical Processes*. John Wiley & Sons.

Drogoul, F., S. Kinnersly, A. Roelen, and B. Kirwan (2007). Safety in design–can one industry learn from another? *Safety Science 45*(1-2), 129–153.

Gruhn, P. E. and H. Cheddie (2006). Safety instrumented systems: design, analysis, and justification.

IEC 61508 (2010). Functional safety of electrical/electronic/programmable electronic safety-related systems – part 1-7. Standard, International Electrotechnical Commission.

Kim, H., M. A. Lundteigen, A. Hafver, F. B. Pedersen, G. Skofteland, et al. (2018). 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.

Kim, H., M. A. Lundteigen, A. Hafver, F. B. Pedersen, G. Skofteland, C. Holden, and S. J. Ohrem (2018). 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*, pp. 1467–1476. CRC Press/Balkema.

Leveson, N. (2011). *Engineering a safer world: systems thinking applied to safety*. Engineering systems. Cambridge, MA: The MIT Press.

Leveson, N. (2020). *An Improved Design Process for Complex, Control-Based Systems Using STPA and a Conceptual Architecture*. Draft retrieved 2020−02−11. Unpublished.

Leveson, N. and J. Thomas (2018). *STPA handbook*.

Rachman, A. and R. C. Ratnayake (2015). 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)*, pp. 11–15. IEEE.

Steinhauser, E. P. (2019). Addressing the challenges of implementing safety instrumented systems in multi-product batch processes. *Journal of Loss Prevention in the Process Industries 57*, 164–173.

Teikari, O. (2014). CORSICA task 4.1 hazard analysis method of digital I&C systems. *Technical Report VTT-R-03821-14*.

Thomas, J., F. Lemos, and N. Leveson (2012). Evaluating the safety of digital instrumentation and control systems in nuclear power plants. *NRC Technical Research Report 2013*.

Zhang, J., H. Kim, Y. Liu, and M. A. Lundteigen (2019). Combining system-theoretic process analysis and availability assessment: A subsea case study. *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability*, 520–536.

Zikrullah, N. A., M. J. P. van der Meulen, H. Kim, and M. A. Lundteigen (2019). 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)*, pp. 2928–2936. Research Publishing.