Next generation safety design for complex embedded systems

KANEMOTO Shigeru[1]

1. The University of Aizu, Tsuruga, Ikki-machi, Aizuwakamatsu-chity, Fukushima, 965-8580, Japan (shigeru.kanemoto@gmail.com)

Abstract: The next generation safety design for highly intelligent and complex systems is discussed. The conventional safety analysis methods such as FTA/ETA or FMEA are all 40-65 years old but our technology is very different today. The main difference is the introduction of computer software safety control. And, the conventional methods are difficult to be used for the complex system safety analysis, since the accidents are often caused by not simple component failures but complex interaction flaw among components and human actions. Hence, Nancy Leveson proposed the concept of STAMP (Systems-Theoretic Accident Model and Process) and the concrete procedure of hazard analysis, STPA (System-Theoretic Process Analysis) to solve the above problems. In the present paper, we discuss how STAMP/STPA is effective in the complex system safety analysis and how it is different from the conventional methods through two kinds of case studies. Also, we will discuss the possibility of STAMP/STPA utilization in NPP operation and maintenance works.

Keyword: safety critical system; embedded system; STAMP/STPA; FTA/ETA; FMEA

1 Introduction

Highly intelligent and complex embedded systems are going to be introduced in our daily life, such as automated driving cars or life support robots (nursing care robot). One of important interests in industrial people is the safety issues, since conventional safety analysis methods, such as FTA/ETA(Fault tree and event tree analysis) or FMEA(Failure mode and effect analysis), are difficult to use for their design. This difficulty is mainly caused by the use of computer software embedded in the system. The software makes safety control algorithms sophisticated and complex. The current safety standards like IEC61508 or ISO26262 for the programmable electronic devices do not take care of complicated software like AI technologies or human machine cooperative safety control. Furthermore, the connection with internet makes the problems difficult. This kind of complex system is sometimes called as 'System of Systems(SoS)'.

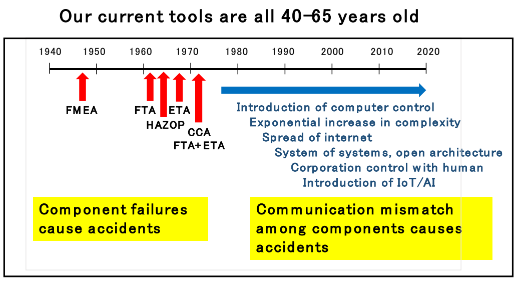
Our current tools are all 40-65 years old
1940
1950
1960
1970
1980
1990
2000
2010
2020
FMEA
FTA
ETA
HAZOP
CCA
FTA+ETA
Introduction of computer control
Exponential increase in complexity
Spread of internet
System of systems, open architecture
Corporation control with human
Introduction of IoT/AI
Component failures cause accidents
Communication mismatch among components causes accidents

Fig. 1 History of safety analysis tools and technology progress.

Nancy Leveson pointed out that the current safety analysis tools are all 40-65 years old but our technology is very different today[1], as shown in Fig. 1. She also suggests that hazards in these complex systems are often caused by not simple component failures but complex interaction flaw (communication mismatch) among components and human actions. She symbolically says these hazards as the emergent safety property of complex systems. In other words, the safety of the whole system is not the summation of the safety of sub-systems (components). So, new safety analysis methodologies, which are applicable to the complex systems, are highly expected. In order to solve this difficult problem, she proposed the concept of STAMP (Systems-Theoretic Accident Model and Process) and the concrete procedure of hazard analysis, STPA(System-Theoretic Process Analysis). The important assumption in STAMP/STPA is that accidents are caused by disorder of safety control actions and their feedback in a dynamic system. This is important difference from the conventional safety analysis in which accidents are usually induced by a chain of component failure and human error.

In the safety design of the conventional safety critical systems, especially, nuclear power plants (NPP), the above failure chain model is assumed and FTA/ETA or FMEA are extensively utilized. Although this procedure is very reasonable and useful in the design stage, it has some difficulties in atypical maintenance and operation works. The conventional hazard

Received date: May 15, 2017
(Revised date: May 29, 2017)

analysis procedures are not easy to apply atypical works since they assume one directional chain of failure propagation and are difficult to deal with hazards in components and human feedback adjustment behavior.

In another view of safety design, the established NPP safety design based on defense in depth concept eventually failed to survive Fukushima accident caused by Tsunami in 2011. Tsunami and earthquake induced SBO (Station blackout) as a typical common cause failure against all emergency power supply devices, since these devices are located at under floors and flooded by Tsunami, also, they were designed by water cooling. However, the unit 5 & 6 plants in the same site survived SBO, since they had one emergency power supply cooled by air and located at outside of reactor buildings. This typically suggests the importance of diversity design. As another example which suggests the importance of diversity, the concept of defense in depth and diversity (D^3) is proposed in full digital reactor protection system design[7]. These experiences of safety design in NPP could be useful for the above mentioned complex embedded system safety design in both positive and negative aspects.

In the working group in Japanese administrative agency, IPA/SEC (Information technology promotion agency/Software reliability enhancement center), we have discussed the above complex embedded system safety issues and published some reports[3]. Based on these reports, the merits and limitations of the conventional and new safety design methodologies are reviewed and discussed. Also, the two simple case studies of STAMP/STPA are given for human-machine cooperative control system hazard analysis to demonstrate how STAMP/STPA works in complex system safety design. Through these case studies, the discussion will be made on what should be the next generation safety design.

2 Review of safety analysis methods

2.1 Accident models

Conventionally, the domino accident model or the Swiss cheese accident model is widely used to explain why the accident occurs[4]. Like a series of dominos falling, a component failure or operator error sequentially causes a next event, and eventually, leads to an accident. Removing any domino would break the chain and prevent an accident. The Swiss cheese model assumes a similar chain of component failures and erroneous actions. Randomly located holes representing individual weaknesses come into alignment and induce an accident by passing through several layers safety barriers.

However, as mentioned in introduction, accidents in complex systems are often caused by miscommunication among components. In other words, disorder of safety control action or feedback of safety information causes the accidents. This means we have to understand the complex system accident behaviors as the dynamic system. Hence, Erik Hollnagel proposed Functional Resonance Accident Model (FRAM)[5,6], based on the stochastic resonance theory in nonlinear dynamic systems. Here, he assumes that the small performance deviation in the subsystem is amplified through nonlinear feedback and leads to an accident. In the similar context, Nancy Leveson proposed the STAMP concept[1]. She proposes accident causes by the disorder of control action and its feedback in the safety control structure diagram.

Besides the above two kinds of accident models, one way and feedback failure propagation models, there are many other system behavior description models, which mainly describe normal system behaviors. MATLAB/Simulink[8] is extensively used for embedded system design as de facto standard in industries to simulate dynamic system behaviors. Although this is a very practical tool for system design, it is not easy to use for qualitative hazard analysis. MFM (Multi-level Flow Modeling)[9] is another example of qualitative simulation for dynamic systems like chemical or nuclear power plants. Since this model can describe both a part-whole component structure and means-end functional relations, it is possible to automatically deduce failure propagation, that is, to make FTA automatically[10,11]. In the embedding system design, SysML (System modeling language) or AADL (Architecture analysis & design language) are often used for system modeling[12]. These models are going to be used for complex system design. However, in order to use them for safety analysis of SoS, there remain many issues to be solved.

2.2 Conventional hazard analysis methods

The above accident models are to conceptually explain how the failure or error propagate in the system and lead to the accident. But, in the safety design process, it is necessary to prepare the concrete procedures for finding hazards hidden in the system design. In this section, we briefly review a part of the conventional tools shown in Fig. 1.

FTA is the most widely used risk evaluation method in a design stage for safety critical systems like NPP. It begins with an undesirable event as a top-event, proceeds in a top-down way to identify the causes of the undesirable event, and, summarizes as a tree structure document. In NPP safety design, ETA is combined with FTA to identify the above undesirable event. The first step of ETA is to identify an initiating significant event like pipe rupture or loss of power. Next, the set of barriers or protective functions to prevent an accident are listed in the anticipated sequence of operation. Then, a logical tree is constructed by tracing forward in time to insert success or failed scenarios of each barrier. Finally, it is determined whether each scenario leads to an accident. In contrast to FTA/ETA, FMEA is a typical bottom-up approach. It starts from failure modes of lower level components and evaluates their effects to the top level system safety.

The feature of these conventional hazard analyses is the one directional thinking in time domain. It is not easy to think interaction of control action and its feedback or interaction of human and machine.

2.3 STAMP/STPA[1]

As mentioned previously, STAMP is one of valuable models for complex system hazard analysis. The most beneficial feature of STAMP in safety design is to provide the concrete procedures, STPA, to analyze hazards. STPA consists of the following four steps:

Step0(1): Define accidents, hazards and safety constraints of the target system. Here, hazards are defined to be 'state' which induces the accident if appropriate control actions are not given. And, the safety constraints are defined to prevent these hazardous states.

Step0(2): Define control structure diagram. Here, control actions(CAs) and its feedback(FB) are explicitly defined from the safety control viewpoints. The simple example is shown in Fig. 2.

Step1: Extract unsafe control actions (UCAs) leading to hazards. Here, four kinds of unsafe control actions are defined:

(N) A control action required for safety is not provided or is not followed.

(P) An unsafe control action is provided that leads to a hazard.

(T) A potentially safe control action is provided too late, too early, or out of sequence.

(D) A safe control action is stopped too soon or applied too long.

Here, these UCAs are examined for each CA in the control structure diagram. The abbreviations, N/P/T/D, represent 'Not provided', 'Provided', 'Timing' and 'Duration', respectively. The four kinds of unsafe control action are logically explained in Fig. 3. It could say this classification is exclusive and comprehensive. The unsafe actions of 'Providing' and 'Not providing' reminds us 'Commission' and 'Omission' error in human factor analysis. As more precise classifications are made for typical human error modes, 'Timing' and 'Duration' error modes are specially picked up in STPA, which are often observed in plant control system accidents. So, it should be noted their might be other typical error mode classification according to individual application fields.

Step2: Identify hazard causal factors or scenarios which induce UCAs based on the control structure diagram. Also, component safety constraints (CSCs) for each component are defined from identified HCFs. These CSCs are detailed constraints for lower level system safety design. These hazard causal factors can be typically listed up as follows:

(1) Unsafe inputs,

(2) Unsafe control algorithms,

(3) Inconsistent, incomplete, or incorrect process model,

(4) Inadequate feedback,
(5) Flaw of actuators and controlled processes,
(6) Out-of-range disturbance, conflict control actions, or environment.

Figure 4 shows these factors on the control structure diagram. By looking up this figure, domain engineer can list up hazard causal factors or scenarios. It should be noted that this figure is just hints for hazard scenario thinking and there might be other hints according to individual domains.

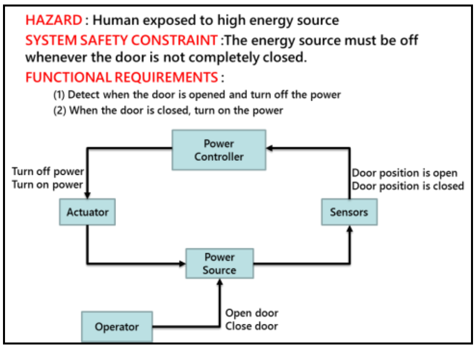
HAZARD : Human exposed to high energy source
SYSTEM SAFETY CONSTRAINT :The energy source must be off whenever the door is not completely closed.
FUNCTIONAL REQUIREMENTS :
(1) Detect when the door is opened and turn off the power
(2) When the door is closed, turn on the power
Power Controller
Turn off power
Turn on power
Door position is open
Door position is closed
Actuator
Sensors
Power Source
Operator
Open door
Close door

Fig. 2 Safety constraint and control structure diagram for power control system.

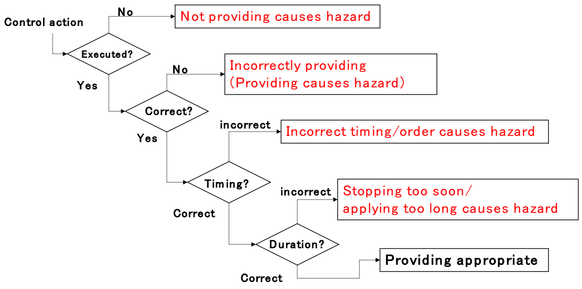

Fig. 3 Logic tree for UCA classification

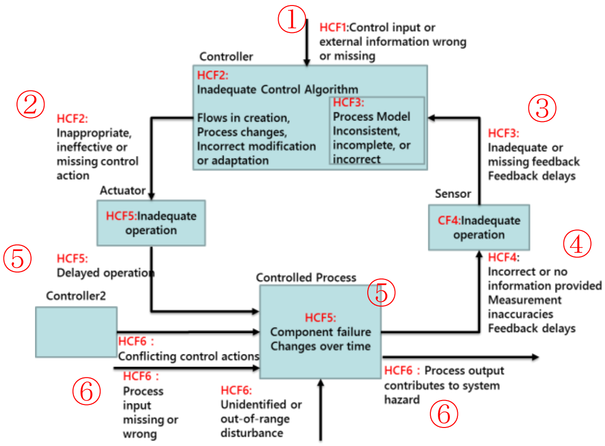
①
HCF1:Control input or external information wrong or missing
Controller
HCF2:
Inadequate Control Algorithm
Flows in creation, Process changes, Incorrect modification or adaptation
HCF3:
Process Model Inconsistent, incomplete, or incorrect
②
HCF2:
Inappropriate, ineffective or missing control action
③
HCF3:
Inadequate or missing feedback
Feedback delays
Actuator
Sensor
HCF5:Inadequate operation
CF4:Inadequate operation
④
HCF4:
Incorrect or no information provided
Measurement inaccuracies
Feedback delays
⑤
HCF5:
Delayed operation
Controller2
Controlled Process
⑤
HCF5:
Component failure
Changes over time
HCF6 :
Conflicting control actions
HCF6 : Process output contributes to system hazard
⑥
HCF6 :
Process input missing or wrong
HCF6:
Unidentified or out-of-range disturbance
⑥

Fig. 4 General hazard causal factors for considering causal scenarios

The procedure of STPA seems to be very simple and too much qualitative, and so, a typical question often comes up. Is this simple and qualitative procedure really useful for hazard analysis of complex system? Traditional hazard analysis is very sophisticated and made by using detailed design knowledge. Also, detailed quantitative simulation is made to evaluate risk of the accident. However, these hazard analyses are just made after completing detailed architecture design. On the other hand, STPA is a top down approach in which the system level safety constraints are defined in abstracted and hierarchy safety control structure diagram, and, the component safety constraints are deduced from the control structure via traceable inference process. This process can make clear the relations among safety requirements, supposed contexts of system use cases and safety specifications. This top down design approach is remarkable difference from the conventional bottom up safety design approaches like FTA or FMEA. Hence, we can expect STAMP/STPA would contribute to think flexibly accident scenarios including unanticipated scenarios. The top down approach can also reduce design costs, especially, in the complex embedded system design, since it prevents a return work often observed in the system architecture design. Of course, STPA does not contradict the conventional hazard analysis like FTA. STPA gives us a broad perspective of the system safety and the conventional hazard analysis reminds us importance of the business proverb, ‘God is in the details’. It is important to combine both a broad perspective and a narrow focus in the complex system safety design.

3 Case study

3.1 Simple chemical plant safety control

In order to investigate the usefulness of STAMP/STPA, we made a simple virtual test case using the chemical plant simulator, which was developed by IPA/SEC WG[3]. The simulators are made by Simulink to make the control algorithms easily understandable. The chemical plant simulator shown in Fig. 5 has tanks, valves, sensors and a control system including emergency safety control logic. The water of Tank-2 is drawn up by the pump and pour into Tank-1. The water level of Tank-1 is controlled at a constant level by PID controller and control valve, CV1. Here, the accident of this system is assumed as the overflow from Tank-1, and, the hazard, which leads to the accident, is the state where the water level of Tank-1 goes above the alarm threshold. In order to avoid the hazard, the emergency drain valve, EV1, is equipped and automatically opened when the Tank-1 level exceeds an alarm set point. The plant operator can also override the EV1 opening action by manipulating the emergency stop button on the screen.

In this simulator, the emergency mitigation control logic is also equipped to suppress the water level before reaching to the alarm level[3]. Here, EV1 is open for just 5 seconds when the Tank-1 level exceeds an alert set point or the operator requires manually, and, prohibits additional operation for next 10 seconds. The details of the mitigation logic is omitted here, and, referred to the reference[3,13]. It is noted that this mitigation function is not the safety functions, and so, it is excluded from the following STAMP/STPA analysis. However, this kind of timing-sensitive and human-machine interference problem would be important in future complex embedded systems. In the above mentioned IPA/SEC WG[3], this issue was also discussed in context of what kinds of V&V should be made for complex control algorithms.

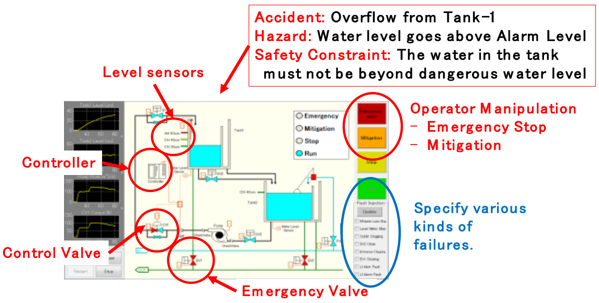
Accident: Overflow from Tank-1
Hazard: Water level goes above Alarm Level
Safety Constraint: The water in the tank must not be beyond dangerous water level
Level sensors
Controller
Control Valve
Emergency Valve
Operator Manipulation
- Emergency Stop
- Mitigation
Specify various kinds of failures.
Emergency
Mitigation
Stop
Run

Fig. 5 Overview of chemical plant model.

The first step of STPA is to define accidents, hazards and safety constraints, which are shown in Fig. 5. Here, we assume overflow from Tank-1 as an accident. The hazardous state which induces this accident is water level goes above the alarm level. And, reverse expression of this hazard becomes the safety constraint of the system. Then, we can describe the safety control structure as shown in Fig. 6. This figure seems to be a little bit strange from the viewpoints of engineers who are familiar to hardware design. However, abstracted expression, or, essential function of the safety control in the system can be regarded as emergency valve (EV1) control by human and computer. Water level of Tank-1 is referential information to judge EV1 opening control. Then, the safety control actions can be defined by CA1-CA3 which are shown by red arrows. The feedback (FB) corresponding to these actions are also shown by blue arrows. Also, we assume pressure from upper organization by the dotted arrow to the operator, which is commonly observed in industry organizations.

This visualized definition of abstracted safety control structure is not a unique solution, and, there are other type of expression[11]. However, it is worth to discuss the safety control mechanism from different viewpoints based on this visualization.

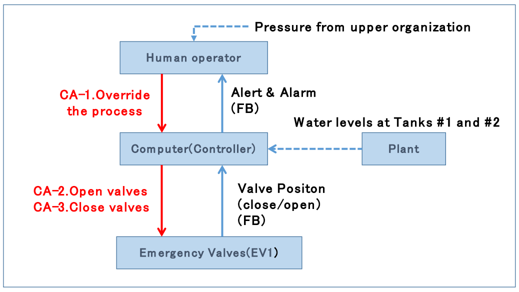
Pressure from upper organization
Human operator
CA-1.Override the process
Alert & Alarm (FB)
Water levels at Tanks #1 and #2
Computer(Controller)
Plant
CA-2.Open valves
CA-3.Close valves
Valve Positon (close/open) (FB)
Emergency Valves(EV1)

Fig. 6 Control structure diagram of chemical plant.

Step-1 and 2 of STPA are straightforward. Based on the three control actions in Fig. 6, the UCA table can be made as shown in Table 1. Here, symbolization such as UCA-1N means control action number and a type of unsafe control actions. This symbolization seems to be trivial but very useful to avoid confusion of consecutive discussions by team members. In step-2, hazardous scenarios are analyzed according to the type of UCAs. Concrete examples are shown below in (1)-(5). Here, Items (1) and (2) are computer related malfunctions and (3)-(5) are human related ones. The relation with the conventional hazard analysis is discussed below.

Table 1 Unsafe control action table.

Control Action	Not providing causes hazard(N)	Providing causes hazard(P)	Incorrect Timing / Order(T)	Stopped Too Soon / Applied too long(D)
CA-1:Human operator overrides the computer process	Operator doesn't override the computer process when it is causing the overflow. (UCA-1N)	Operator overrides the computer process working as intended, and causes the overflow. (UCA-1P)	Operator overrides the computer process more than X seconds after water level is at the alarm level.(UCA-1T)	Effect of override continues without being noticed, and causes the overflow with no computer control.(UCA-1D)
CA-2:Computer opens supplying valves and closes drainage valves	Safe-side action	Computer opens supplying valves or closes drainage valves when water level is at the alarm level. (UCA-2P)	Computer opens supplying valves or closes drainage valves before water level falls below the alarm level.(UCA-2T)	Safe-side action
CA-3:Computer closes supplying valves and opens drainage valves	Computer doesn't close supplying valves or doesn't open drainage valves when the water level reaches the alarm level.(UCA-3N)	Safe-side action	Computer closes supplying valves or opens drainage valves more than X seconds after water level is at the alarm level. (UCA-3T)	Computer stops closing supplying valves or opening drainage valves, before it is fully closed/opened. (UCA-3D)

(1) Computer related UCAs-(2P,2T,3N,3T)

Scenario-1: Computer is unaware (or cannot notice in time) that water level is at the alarm level.

- No (or incorrect) information on water level provided in time for computer

Scenario-2: Computer cannot properly manipulate the valves.

- Error in software design to control the valves
- Actuator to open/close the valve is not working though signal comes in.

(2) Computer related UCA-(3D)

Scenario-1: Computer believes it has closed/opened the valves by the fact that computer sent (or the valves received) the signal to do so, while they have not actually closed/opened.

- No (or incorrect) feedback on the actual position of valves

(3)Human related UCAs-(1N,1T)

Scenario-1: Operator is unaware (or cannot notice in time) that the computer does not work properly.

- No (or incorrect) information on water level or computer instruction provided in time for human operator
- Distracted by another task
- Believes that computer always works properly

Scenario-2: Operator cannot properly manipulate the valves.

- Doesn't know how to override and manipulate the valves
- Design error in override function (Computer cannot properly process the conflicting instruction from operator.)

(4) Human related UCA-(1P)

Scenario-1: Operator misunderstands the computer process while it is working properly.

- Incorrect information on water level or computer instruction provided for human operator
- Does not understand how the computer is designed to work

Scenario-2: Operator overrides the computer process inadvertently.

- Design error in override function (easy to be mistaken)

Scenario-3: Operator overrides the computer process intentionally to increase the output from the plant in exchange for safety margin.

- Tough pressure or production quota from management
- Safety culture is not well established

(5) Human related UCA-(1D)

Scenario-1: Effect of override continues after operator took the proper action, while he/she does not recognize it.

- Believes that effect of override terminates after operator took the proper action.

Figure 7 is the fault tree made by the conventional FTA. The accident, overflow, could be caused by three types failure modes, those are:

Gr-A: Emergency system troubles which include actuator, sensor, computer hardware and software
Gr-B: Operator error
Gr-C: Ordinary control system troubles

The Gr-C troubles are excluded in STPA since it focuses on just the safety related systems. Gr-A corresponds to items (1) and (2) of hazard scenarios in STPA. The conventional results of Gr-A seem to be concrete more than those of STPA. But, this conventional one just focusses on the rated power operation and lists up just the trouble of emergency valve, EV1. However, in STPA, the valve operation troubles under both water supplying and draining phases are analyzed. Although these scenarios are very abstracted, they are useful to design the algorithm or interlock for a start-up phase as the component safety constraints. STPA results of (3)-(5) correspond to Gr-B operator error. It is seen at a glance that STPA can analyze various type of human error scenarios flexibly. A typical example is 'Does not understand how the computer is designed to work', which is a typical feature of computer control system and different from the conventional analog control. When the constant process variables like the water level are observed in the display, the operator cannot know whether the computer is down or the process variable is very stable and normal. The common trick of computer hacker is disguise by camouflaging these process variables by normal constant values and attacks the remaining control algorithms.

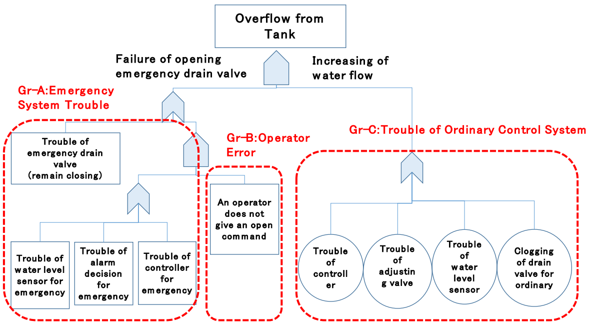
Overflow from Tank
Failure of opening emergency drain valve
Increasing of water flow
Gr–A:Emergency System Trouble
Gr–B:Operator Error
Gr–C:Trouble of Ordinary Control System
Trouble of emergency drain valve (remain closing)
Trouble of water level sensor for emergency
Trouble of alarm decision for emergency
Trouble of controller for emergency
An operator does not give an open command
Trouble of controller
Trouble of adjusting valve
Trouble of water level sensor
Clogging of drain valve for ordinary

Fig. 7 Fault tree classification of overflow.

3.2 Inverted two wheel vehicle control

Figure 8 is a target system for another case study in which the inverted two-wheel vehicle (LEGO-EV3) is designed based on self-standing PID controller and human remote control using joystick and wireless communication. Human can control LEGO-EV3 to go forward, backward or rotating. Due to the wireless communication, a certain delay and small variation of transmission time exist.

We assume the accident as ‘Overturning of LEGO’, and, the hazard as ‘Large body angle state in which LEGO cannot be controlled by PID controller’. Then, the safety constraint becomes ‘Angle and angular velocity of body and wheel are kept under a certain threshold’. The control structure diagram can be expressed by Fig. 9. Here, we assume 4 remote control actions from the operator to the controller of LEGO, 4 control actions from the controller to LEGO actuators which are the same as the previous ones, and, 1 control action for self-standing by the PID controller. Also, in Fig. 9, we assume two kinds of disturbances, ‘expectation from audiences’ and ‘road or wind condition variation’. Since this system is used for technology demonstration in some exhibition, ‘expectation from audiences’ could be pressure on the operator and may induce mis-operation.

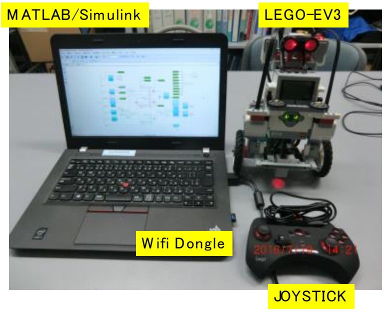

Fig. 8 Overall LEGO-EV3 control system.

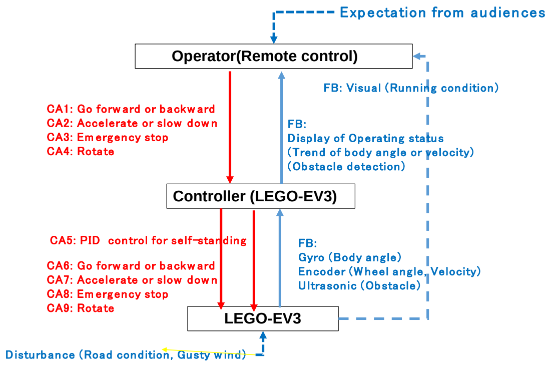
Expectation from audiences
Operator(Remote control)
FB: Visual (Running condition)
CA1: Go forward or backward
CA2: Accelerate or slow down
CA3: Emergency stop
CA4: Rotate
FB:
Display of Operating status
(Trend of body angle or velocity)
(Obstacle detection)
Controller (LEGO-EV3)
CA5: PID control for self-standing
CA6: Go forward or backward
CA7: Accelerate or slow down
CA8: Emergency stop
CA9: Rotate
FB:
Gyro (Body angle)
Encoder (Wheel angle, Velocity)
Ultrasonic (Obstacle)
LEGO-EV3
Disturbance (Road condition, Gusty wind)

Fig. 9 Safety control structure of LEGO-EV3.

Based on the above control structure, the UCA table can be created. Typical examples for CA-1(remote instruction) and CA-5(PID control instruction) are shown in Table 2. In this table, a typical hazard scenario are shown below:

1) Hazard scenario for UCA-1P: ‘Overturn occurred by instruction to go backward in the middle of instruction to go forward’
   Component safety constraint: ‘Limitation of the change ratio of acceleration (Jerk limitation)’
2) Hazard scenario for UCA-5P: ‘Overturn occurred by unexpected disturbance such as road condition change, obstacles, or gusty wind’
   Component safety constraint: Implementation of new control algorithm such as rule based non-linear controller overriding linear PID feedback.

The derivation of component safety constraints is a good example which suggests that STPA is very useful as a top down design approach. Comparing bottom up design improvements, this approach will reduce design costs, especially, in the complex embedded system design, since it prevents a return work often observed in the system architecture design.

Table 2 Unsafe control action table for LEGO-EV3

Control Action	Sender	Receiver	Not Providing Causes Hazard(N)	Providing Causes Hazard(P)	Wrong Timing or Order (T)	Stopped Too Soon or Applied Too Long(D)
CA-1 Instruction to go forward or backward	Operator (Remote Control)	Controller (EV3)	In the absence of Instruction to go backward for evading a obstacle, EV3 crash into a obstacle (UCA-1N)	Overturn is occurred by instruction to go forward and backward in wrong condition（UCA-1P)	Instruction to go forward and backward for evading a obstacle is delayed, therefore EV3 crash into a obstacle（UCA-1T)	–
CA-5 PID control instruction	Controller (EV3)	EV3	Overturn is occurred by discontinuing a control signal（UCA-5N)	Overturn is occurred by wrong instruction from controller（UCA-5P)	Overturn is occurred by control instruction at wrong timing（UCA-5T)	–

3.3 Discussions on case study results

Although the above two kinds of case studies seem to be a toy problem, they suggest us the usefulness of STAMP/STPA from the flowing viewpoints:

- Top down safety design: The component safety constraints are systematically derived from the system basic safety constraints and abstracted safety control structure model. It is useful to prevent hazards under various kinds of complex contexts in which the complex systems often used.
- Prevention of hazards in various system modes: The complex systems have a variety of operation modes, maintenance works or recovery modes from accidents. To find hazards under these various kinds of context, flexible and systematic thinking based on STAMP/STPA is necessary.
- Prevention of human error: STAMP/STPA can provide various types of human/organization error modes.

The consumer oriented complex systems or infrastructure systems like NPP should be properly operated and maintained for their long life cycles under various environmental changes such as degradation of hardware, change of use environment, change of regulation or update of sub-systems. In such cases, it is important to keep in mind the top level safety constraints and corresponding component safety constraints with traceable forms.

4 Conclusion

Perspective of next generation safety design methods are discussed through the case studies of two kinds of STAMP/STPA applications. As mentioned in Introduction, the current safety standards do not include human-machine cooperative safety control or AI technology. So, industrial people has strong interest on next generation standards or safety design methodologies which can be used for the complex embedded systems. The case studies in the present paper suggests STAMP/STPA would be one of effective methods for these safety design. However, they also show us that the STPA analysis largely depends on engineer's knowledge or skills.

To reduce the above dependence, the concept of model based systems engineering (MBSE) is attracting industrial people interest. The key issue of MBSE is how the target system is modelled. As mentioned in the present paper, many modeling methods are proposed, but, they have their own merits and demerits according to their application areas. Especially, from viewpoints of safety design, they have to describe not only normal system behaviors but also hazardous behaviors. For example, Simulink is used for developing autonomous driving car as de facto standard, but, it is difficult to use this for hazard analysis. On the other hand, STAMP/STPA is expected for hazard analysis of the complex safety critical systems, but, it too much depends on engineer skills and has not yet established as safety design standard.

From viewpoints of NPP industry engineers, MBSE is not a new idea, since it is commonly used in the system design. So, the know-how obtained there should be spread to other industrial area. Conversely, new methods like STAMP/STPA could be utilized in NPP. There are many areas in which the conventional hazard analysis methods like FTA are difficult to apply, for example, non-routine operation or maintenance works. Many incidents or accidents still remain in these works. The improvement of the overall system safety should be pursued continuously without being satisfied in the current safety analysis.

References

[1] LEVESON, N.G.: Engineering a Safer World, The MIT Press, 2012.

[2] KANEMOTO, S., and OOTOMO, S.: Application of new hazard analysis model for embedded systems, Nuclear Safety and Simulation, Vol. 6, Number 2, June 2015.

[3] IPA/SEC reports (March 2017, in Japanese): http://www.ipa.go.jp/sec/reports/20170321_2.html, http://www.ipa.go.jp/sec/reports/20170324.html

[4] THOMAS, J.: Extending and automating a system-theoretic hazard analysis for requirements generation and analysis, PhD dissertation, MIT, 2013.

[5] HOLLNAGEL, E., and GOTEMAN, O.: The functional resonance accident model. Cognitive System Engineering in Process Plant 2004, 2004.

[6] HOLLNAGEL, E., WOODS, D. D., and LEVESON, N. C.: (Eds.) Resilience engineering: Concepts and precepts. Aldershot, UK ,2006.

[7] WOOD, R.T.: Diversity Strategies to Mitigate Posturated Common Cause Failure Vulnerabilities, NPIC & HMIT, November 7-11, 2010, Las Vegas, USA.

[8] https://jp.mathworks.com/products/simulink.html

[9] LIND, M.: Modeling Goals and Functions of Complex Industrial Plants, Applied Artificial Intelligence, Vol.8, p259, 1994.

[10] GOFUKU, A. and OHARA, A.: A Systematic fault tree analysis based on multi-level flow modeling: Toward Innovative Nuclear safety and Simulation Technology (ISSNP2008), vol.2, P.11, 1994.

[11] KANEMOTO, S., and OTOMO S.: “Application of new hazard analysis model for embedded systems,” STSS/ISSN2015, Kyoto, Japan, August 25-28, 2015.

[12] KORDON, F., HUGUES, J., CANALS, A., and DOHET, A.: Embedded systems/ Analysis and Modeling with SysML, UML and AADL, ISTE and John Wiley, 2013.

[13] BJORKMAN, K., FRITS, J., VALKONEN, J., LAHTINEN, J., HELJANKO, K., NIEMELA, I., and HAMALAINEN, J.J.: Verification of Safety Logic Designs by Model Checking, NPIC&HMIT, April 5-9, 2009, Knoxville, USA