SAFETY ANALYSIS METHOD BASED ON A PARALLEL STATE TRANSITION DIAGRAM FOR EMBEDDED SYSTEMS

Zoohaye KIM, Yutaka MATSUBARA and Hiroaki TAKADA
Graduate School of Information Science
Nagoya University, Nagoya, Japan
{zoo, yutaka, hiro}@ertl.jp

ABSTRACT

In order to exhaustively analyze the effects of failures in safety-critical embedded systems, we have studied safety analysis methods based on state transition diagrams. However, the analytical worksheets and guidewords used in these methods are not suitable for analyzing parallel state transition diagrams, which represent the behavior of systems whose functions work in parallel. We propose a method whereby, if the severity of a deviation on a state transition diagram can be determined regardless of the other state transition diagrams, the total number of deviations to be analyzed can be reduced. Moreover, we show that techniques for containing the effects of deviations (e.g., memory protection) can limit their analytical area. Thus, we perform a Safety Analysis method based on a Parallel State Transition Diagram (SAPSTD). To clarify its effectiveness, we apply a conventional method and SAPSTD to the specifications of an example embedded system and compare the results of an evaluation.

KEY WORDS
State Transition Diagram, Safety Analysis, UML, HAZOP, SHARD, SMHA.

1 Introduction

The complexity of safety-critical embedded systems—such as aeroplane and automotive control systems—has increased rapidly. Thus, maintaining and improving their integrity becomes a serious concern, and to this end several international standards for functional safety—such as IEC61508 [1] and ISO26262 [2]—have recently been published. A thorough analysis of a system's safety is important to clarify the effects of failures and deviations. During the development of safety-critical systems, it is desirable to find failures and deviations that could cause serious harm as early as possible. For this reason, safety managers and system developers must perform a safety analysis based on the system specifications, and consider safety measures for reducing the severity of a failure to an acceptable level.

For many older embedded systems, it was relatively easy to perform a safety analysis, because software was not used in their safety-related functions. Even if software was used, its size would have been small. In contrast, modern safety-related functions in a large number of embedded systems are reliant on complex software. Thus, it is difficult to comprehensively analyze the behavior of a system affected by hardware failures and software defects. Unfortunately, no standard analysis method for safety-related systems and software has been established.

State transition diagrams (STDs) and state transition tables have been used in approximately 80% of embedded systems development projects in Japan [3]. Therefore, safety analysis methods based on STDs can be applied to numerous embedded systems. These methods can also support software engineers (who are not safety experts) in the application of safety analysis.

Lano *et al.* proposed a HAZards and OPerability (HAZOP)-based safety analysis method for the Unified Modeling Language (UML) [5], whereby guidewords for analyzing state charts were used to derive deviations. However, the analysis flow and analytical worksheet, which are necessary for practical analysis, were not explained. We have studied a technique based on STDs called the Safety Analysis method based on State Transition Diagrams (SASTD) [6]. In [6], we proposed that guidewords be used alongside the analysis flow and an analytical worksheet for the safety analysis of STDs. The results of a case study in [6] showed that we could analyze STDs more efficiently with SASTD than using Failure Mode and Effective Analysis (FMEA) [7]. Moreover, we extended SASTD to a Safety Analysis method based on Hierarchical State Transition Diagrams (SAHSTD) [8] in order to analyze more complicated, hierarchical STDs (HSTDs).

In most embedded systems, several functionalities, such as interrupt handling or error checking, work simultaneously. In general, the STDs of such systems are modeled in a parallel STD (PSTD), in which system state transitions are independent and parallel in orthogonal STDs. It is difficult to analyze these parallel systems with conventional methods for safety analysis, because the guidewords and worksheets are not suitable for a PSTD.

This paper presents a Safety Analysis method for a PSTD (SAPSTD) and proposes a new worksheet. The worksheet supports not only safety specialists, but also software engineers in their analysis of the PSTDs modeling embedded systems and software. To confirm the applicability and effectiveness of SAPSTD, we prepare a PSTD and a compositional STD (CSTD), which combines inde-

pendent STDs, according to the system specification of an electric pot. We apply SAPSTD and SASTD to the PSTD and the CSTD, respectively, and consequently confirm that all deviations derived by SASTD can also be derived by SAPSTD. Furthermore, we show that the number of deviations to be analyzed can be reduced in SAPSTD by limiting their analytical area.

This paper is organized as follows: Section 2 summarizes some related work, before Section 3 explains SAPSTD in detail. Section 4 presents a case study for SAPSTD, and we compare the results of SAPSTD and SASTD, and discuss the effectiveness of the proposed method, in Section 5. Finally, Section 6 concludes the paper.

2 Related Work

The HAZOP analysis method [9] was initially developed to analyze chemical process systems using guidewords. However, the standard guidewords used in HAZOP are not suitable for software analysis. Pumfrey *et al.* proposed a HAZOP-like method, called Software Hazard Analysis and Resolution in Design (SHARD) [10], in which the guidewords were refined so that HAZOP could be applied to software. Nonetheless, these new guidewords—such as *Omission* (A service is not provided), *Commission* (A service is provided when it is not required), *Early* (A service is provided early in relation to an expected time frame), *Late* (A service is provided late in relation to an expected time frame), *Value* (There are two types of value error to be considered - Detectable/Coarse and Undetectable/Subtle)—are unsuitable for the analysis of STDs. It is difficult to analyze the deviations of properties to be satisfied in a state by using guidewords defined in [10].

Analysis methods based on UML [4] have been proposed in [5] and [11]. These used guidewords to analyze the deviations in statechart diagrams, which are one of the UMLs. Do Hoang *et al.* applied this method to a robot control system in [11]. However, neither [5] nor [11] gave details of the analysis flow and analytical worksheet for the method.

We have also studied a safety analysis based on STDs [6]. This proposed not only guidewords, but also the analysis flow and an analytical worksheet for the safety analysis of STDs. In SAHSTD [8], we extended SASTD to enable its application to an HSTD. The results of a case study in [8] showed that we could analyze STDs using SAHSTD more efficiently than with SASTD. However, the worksheet used in SASTD and SAHSTD is not suitable for PSTDs, and so this paper presents a new method for their analysis.

Incorrect control actions, rather than component failures in a system, are the focus of STAMP-Based Process Analysis (STPA) [12]. This allows hazards due to unsafe or unintended interactions among the components in a system to be identified. However, it is difficult to use this method to analyze the deviations and state transitions within each component, because STPA only focuses on the flow of information between components.

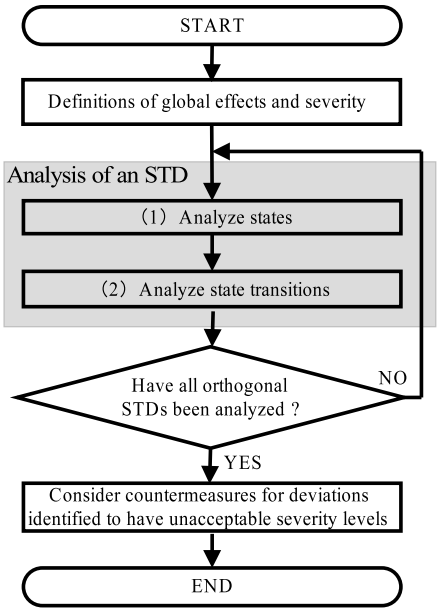

Figure 1. Overall safety analysis flow.

State Machine Hazard Analysis (SMHA) [13] was developed to search for paths from an initial system state to an unacceptable state. In this method, unacceptable states and state transitions due to hardware failures and software bugs must be exhaustively added to the STD of the system. In contrast, the SAPSTD proposed in this paper supports the derivation of unacceptable states and deviations based on the PSTD of normal system operations.

3 Safety Analysis Method for Parallel State Transition Diagrams

3.1 A parallel state transition diagram targeting

From the definition of an STD in Mealy Machine [4], STDs describe the state and the state transitions in a system. Information in the STDs and the properties of each state are available during the analysis. To select methods starting in a lower state, we can study a state transition from an STD in an upper level to an STD in a lower level using the Reset transition and History transition [4]. With SAPSTD, it is possible to analyze any case.

3.2 Safety analysis flow

Figure 1 shows the safety analysis flow of SAPSTD. Firstly, we identify known global effects and assess their severity. Global effects are classified from the perspective of user safety and the achievement of the functional objectives of the system. The severity of each deviation is defined according to the relationships among global effects. Thus, the absolute value of the severity has no meaning. Sec-

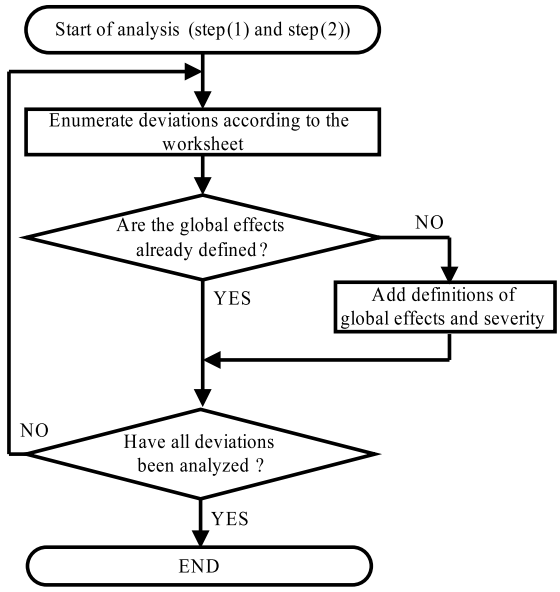

Figure 2. Analysis flow for states and state transitions.

ondly, we analyze the states and state transitions in PSTDs. Analysis of an STD is composed of (1) Analyze states and (2) Analyze state transitions as shown in Figure 1. We can analyze STDs of the PSTD in no particular order. Finally, we consider countermeasures for any deviations identified to have an unacceptable severity.

Figure 2 presents the analysis flow for (1) states and (2) state transitions. During the analysis in Figure 2, we derive a deviation and its global effects, then check whether the corresponding global effects of the deviation have been defined. If they have not, we add a definition of the global effects and determine the severity of the deviation. We repeat this analysis flow for all states and state transitions in the PSTD. The above analysis is performed with a worksheet, the details of which are given in Section 3.3. In analyzing the states of the PSTD, deviations are derived with a combination of a guideword and an attribute, as shown in Table 1. In this phase, deviations from the correct properties required in that state, and from the correct conditions for triggering a state transition, are derived.

In the analysis of state transitions, deviations are again derived with a combination of a guideword and an attribute from Table 2. In this phase, deviations from intended state transitions and from the correct actions to be performed during state transitions are derived. In the analysis of an STD, we need only analyze states and state transitions in one STD. However, to comprehensively analyze a PSTD, we must also consider states and state transitions in other orthogonal STDs. The details of this are explained in the next section. Although we have determined guidewords and attributes that can be applied to various STDs, these can be modified depending on the target system.

Table 1. Guidewords and attributes for the derivation of state deviations.

Guideword	Attribute	Interpretation
Value	-	The value of the property to be met in the state is invalid.
More	transition	The state transition occurs without the event happening.

Table 2. Guidewords and attributes for the derivation of state transition deviations.

Guideword	Attribute	Interpretation
No	transition	Although the event happens, the state does not transition. The required actions are not executed.
Incorrect		The destination state is not the expected one. Unintended actions are executed.
Early		The state transition occurs earlier than required. The required actions are executed.
Late		The state transition occurs later than required. The required actions are executed.
More	action	The state transition occurs correctly. However, an unintended action is executed in addition to the required actions.
Incorrect		The state transition occurs correctly. However, an unintended action is executed instead of a required action.
Missing		The state transition occurs correctly. However, one of the required actions is not executed.

3.3 Worksheet for STDs

We now analyze the nine items that compose a deviation. The first is an ID number for identifying the deviation. The second item consists of the specifications that are subject to the analysis or the description of the design. The analysis of states must meet all the properties required for state transition. This item also contains the specifications for state transition analysis or a description of the design. The third item describes the contents of the deviation with attributes and guidewords, and the fourth item identifies the effects of the deviation on the internal system. The fifth item describes the *state/state transition* of other orthogonal STDs. We enumerate the orthogonal *states/state transitions* and consider the global effects of each one. If there are three or more orthogonal STDs, we should add columns for each orthogonal STD. Global effects are analyzed in terms of the local effects of the entire system due to deviations or

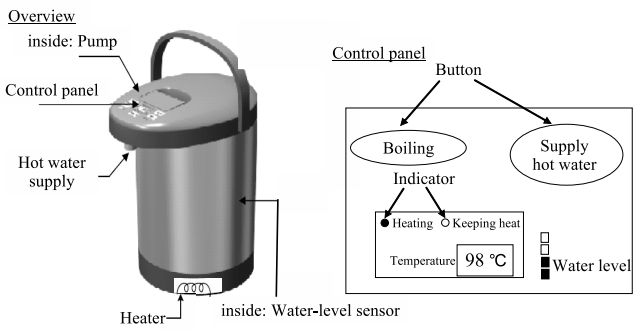

Figure 3. Overview of the electric pot and its control panel.

the effects on the system user. Item six contains a description of these effects, and item seven is a value indicating their severity. To determine this value, it is necessary to define whether the severity is acceptable to the system, the level of safety required in the system, and any cost constraints applicable to the system. The eighth item states whether the cause of the deviation is related to hardware or software, and the ninth item describes countermeasures to reduce high severity or prevent the deviation occurring. Countermeasures can apply to both software and hardware. It is difficult to explicitly calculate the probability of a failure. In this paper, we enumerate all deviations, regardless of the probability of occurrence, and only use the severity when we can determine whether the occurrence of a failure can be tolerated.

4 Case Study

4.1 Specification and STD for the electric pot

The Society of Embedded Software Skill Acquisition for Managers and Engineers (SESSAME) [14] has distributed a requirement specification for the fictitious electric pot depicted in Figure 3. The pot is designed to achieve both the functional requirements for heating water and the operational safety requirements for users. This section describes a safety analysis case study for the electric pot using SAP-STD.

Figure 4 presents the system level specification of the electric pot, in which the state is specified as a two-level HSTD, i.e., upper level STD and lower level STD. As shown in Figure 5, the possible upper-level states of the electric pot are *Idle*, *Running*, and *Error*. In this case study, we focus solely on the pot's behavior in the *Running* state. Figure 4 illustrates that the behavior of the electric pot in the *Running* state can be specified in two ways. Thus, we prepare two specifications in order to compare the results of our safety analysis. One is the PSTD specification, shown as Case 1 in Figure 4, in which the state of the electric pot transitions in parallel between *Supplying hot water* and *Heating* (see Figure 6). The other specification is for the

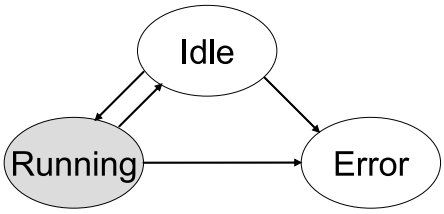

Figure 5. Upper-level STD for the electric pot.

CSTD, shown as Case 2 in Figure 4. As shown in Figure 7, the state of the electric pot does not transition in parallel in the CSTD.

To heat water, the user turns on the power, pours water into the pot, sets the target temperature, and presses the *Boiling* button. A heater installed in the electric pot then heats the water to boiling point. While heating the water, the system state of the electric pot is *Running*. The *Heating* indicator is lit and the *Keeping heat* indicator is off.

When the water reaches boiling point, the state of the electric pot transitions to *Removing chlorine*. After 3 min, the state of the electric pot transitions from *Removing chlorine* to *Keeping heat*. When the electric pot is in the *Keeping heat* state, the *Heating* indicator switches off and the *Keeping heat* indicator lights up.

While the user holds down the *Supply hot water* button, a pump installed in the electric pot operates and water comes out. In the PSTD, the state of the electric pot transitions from *Not-supplying hot water* to *Supplying hot water*. When the user releases the *Supply hot water* button, the pump stops. At this point, the state of the electric pot transitions from *Supplying hot water* to *Not-supplying hot water*. The necessary properties for each state are shown in Table 3. As explained above, in the PSTD, the state of the electric pot transitions in parallel. In contrast, the electric pot is in only one state in the CSTD, and does not transition in parallel.

Table 3. Necessary properties for each state.

Hot water mode	Pump
Supplying hot water state	open
Not-supplying hot water state	closed

Heating mode	*Boiling* indicator	*Keeping heat* indicator	Heater
Heating state	on	off	on
Removing chlorine state	on	off	on
Keeping heat state	off	on	on

Specification for STD of upper level
(H.1) The initial state is the *Idle*.
(H.2) In the *Idle* state, when a user pours water into the pot and closes the lid, the state transitions to *Running*.
(H.3) In the *Idle* state, if the system detects an error, the state transitions to *Error*.
(H.4) In the *Running* state, if the system detects that no water remains in the pot, the state transitions to *Idle*.
(H.5) In the *Running* state, if the system detects an error, the state transitions to *Error*.

Specification for STD of lower level: *Running* state
Case 1: Specification for the PSTD
- State transitions in the *Heating* mode:
(A.1) The initial state is *Heating*.
(A.2) In the *Heating* state, when the water temperature reaches boiling point, the state transitions to *Removing chlorine*.
(A.3) In the *Removing chlorine* state, the water is boiled to remove chlorine for 3 min. After this period, the state transitions to *Keeping heat*.
(A.4) In the *Keeping heat* state, when the *Boiling* button is pressed, the state transitions to *Heating*.
- State transitions of the *Supplying hot water* mode:
(B.1) The initial state is *Not-supplying hot water*.
(B.2) In the *Not-supplying hot water* state, if the user holds down the *Supply hot water* button, the state transitions to *Supplying hot water*.
(B.3) In the *Supplying hot water* state, if the user releases the *Supply hot water* button, the state transitions to *Not-supplying hot water*.
Case 2: Specification for the CSTD
(C.1) The initial state is *Heating and Not-Supplying hot water*.
(C.2) In the *Heating and Supplying hot water* state, when the water temperature reaches boiling point, the state transitions to *Removing chlorine and Supplying hot water*.
(C.3) In the *Removing chlorine and Supplying hot water* state, the water is boiled to remove chlorine for 3 min. After this period, the state transitions to *Keeping heat and Supplying hot water*.
(C.4) In the *Keeping heat and Supplying hot water* state, if the *Boiling* button is pressed, the state transitions to *Heating and Supplying hot water*.
(C.5) In the *Heating and Not-supplying hot water* state, when the water temperature reaches boiling point, the state transitions to *Removing chlorine and Not-supplying hot water*.
(C.6) In the *Removing chlorine and Not-supplying hot water* state, the water is boiled to remove chlorine for 3 min. After this period, the state transitions to *Keeping heat and Not-supplying hot water*.
(C.7) In the *Keeping heat and Not-supplying hot water* state, if the *Boiling* button is pressed, the state transitions to *Heating and Not-supplying hot water*.

Figure 4. System specification for the electric pot.

4.2 Identification of global effects and assessment of severity

The global effects of deviations in the electric pot system can be identified, enabling us to assess their severity. Table 4 lists the results of this procedure. In this case study, we consider global effects from the perspective of the state of the water and the safety of the user. In terms of user safety, the worst situation is that boiling water is emitted at an unexpected time, because this may result in scalding. Thus, we assess the severity of the global effect as the maximum value of 9. Even if users do not get scalded, the global effect of unboiled water being emitted at an unexpected time should also be avoided, and so the severity of this effect is assigned a value of 8. We continue to evaluate the severity of different situations until, in the case of no deviations, the severity is defined as 1. As the main objective of this phase is to clarify the relative relationships among global effects and determine which are acceptable, the absolute value of the severity has no meaning.

4.3 Assumptions in safety analysis

One of the most important safety requirements for the electric pot is to ensure that the user does not get injured whenever a system malfunction occurs. Thus, we determine that global effects with a severity of 6 or less in Table 4 are acceptable. If a deviation with a severity of 7 or more is found during the safety analysis, we insert countermeasures to mitigate the effects of the deviation and reduce its severity to a value of 6 or less. To simplify the safety analysis, we assume the following.

- Operation errors by the user are not considered in order to limit the analytical area.

Table 4. Definition of global effects and severity.

Global effects		Definitions	Severity
Safety of user	State of hot water		
Scalded	boiled	Boiling water comes out at an unexpected time	9
Safe	unboiled	Unboiled water comes out at an unexpected time	8
Safe	unboiled	Unboiled water comes out	7
Safe	unboiled	Boiling functionality does not work	6
Safe	boiled or unboiled	No water comes out	5
Safe	unboiled or not kept warm	Users are informed that an error has occurred	4
Safe	kept warm	Water is kept warm but chlorine may remain in the water	3
Safe	boiled or kept warm	Everything working correctly except the indicator signal	2
Safe	boiled or kept warm	Everything working correctly	1

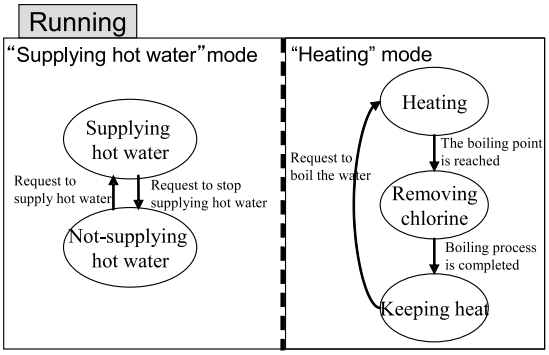

Figure 6. Lower-level PSTD (Case 1) for the electric pot.

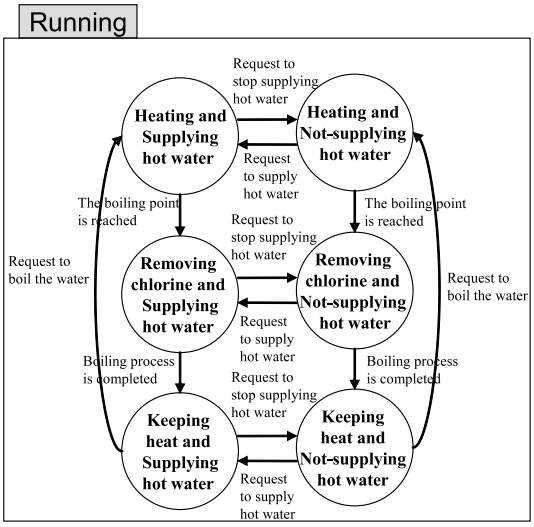

Figure 7. Lower-level CSTD (Case 2) for the electric pot.

- Multiple failures at the same time are not considered, because their probability of occurrence seems to be sufficiently low.
- In a state transition from an upper level to a lower level, we analyze states in the lower level using *Reset transition*, regardless of the previous state.
- The analytical area of deviations derived by the guideword *Incorrect transition* is limited.

By considering *Incorrect transitions*, deviations related to incorrect state transitions from an analytical target can be derived. The final assumption means that the *Incorrect transitions* analyzed are limited to those having a transitional relationship with the state, regardless of the direction. For example, in the analysis of deviations from the *Heating and Supplying hot water* state, the transitions represented by dotted arrows in Figure 8 are analyzed for incorrect transitions, whereas those represented by bold white arrows are not.

4.4 Safety analysis for PSTD and CSTD

To confirm the effectiveness of SAPSTD, we analyze the PSTDs shown in Figure 6 using SAPSTD, and the CSTD shown in Figure 7 by SASTD. Moreover, we compare the number of deviations to be analyzed.

Some partial state transition deviation results from the SAPSTD are presented in Table 5. ID1 represents a functional specification for (A.2) in Figure 4, i.e., in the *Heating* state, a transition to the *Removing chlorine* state occurs when the water reaches boiling point. By applying the *No transition* guideword, we can derive the deviation "The state does not transition to the *Removing chlorine* state." The system remains in the *Heating* state; thus, all of the water in the pot may boil away. Under requirement pot-330-13 in [14], the water level sensor in the electric pot detects this situation and forcibly transitions the system to the *Idle* state. When this deviation in *Heating* mode occurs, the system state in the orthogonal STD, i.e., the *Hot water* mode shown in Figure 6, is either *Supplying hot water* or *Not-supplying hot water*. In this case, the global effects and their severity are the same regardless of the system state in the orthogonal STD. The severity of the global effect when

Table 5. Partial results of SAPSTD for state transition deviations in heating mode.

ID	Item	Deviation	Local effects	State/state transition in orthogonal STD	Global effects	Severity	Causes	Countermeasures
1	(A.2) In the *Heating* state, when the water temperature reaches boiling point, the state transitions to *Removing chlorine*.	No transition The state does not transition to *Removing chlorine*.	If there is no water left in the pot, the state transitions to *Idle* (in pot-330-13 requirements). The deviations are detected and the user is informed (Safe Failure).	*Supplying hot water* (pump:open) *Not-supplying hot water* (pump:closed)	Water is not boiling	4	N/A	N/A
7		Incorrect action In the *Heating* state, when the water temperature reaches boiling point, the state transitions to *Removing chlorine*(only deviation is that the heater is off).	Unboiled water comes out	*Supplying hot water* (pump:open)	-Safe -Unboiled water comes out at an appropriate time	7	-Heater failure -Memory failure in heater	-Duplicate heater -Memory protection by MMU（Memory Protection Unit）or MPU（Memory Management Unit）
				Not-supplying hot water (pump:closed)	-Safe -Water is not boiling	4	N/A	N/A

Table 6. Partial results of SAPSTD for state deviations.

ID	Item	Deviation	Local effects	State/state transition in orthogonal STD	Global effects	Severity	Causes	Countermeasures
3	*Not-supplying hot water* state (pump:closed)	Value pump:open	Water comes out at an unexpected time	-*Heating* -*Removing chlorine* -*Keeping heat*	-Scalded -Boiled water comes out	9	-Pump failure -Memory failure in pump	-Duplicate pump -Memory protection by MMU or MPU

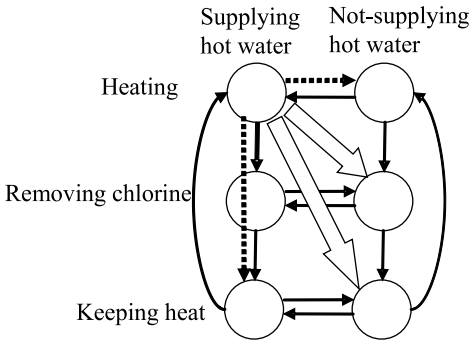

Figure 8. Analytical area of deviations due to incorrect transitions (we consider transitions denoted by dotted arrows and neglect those denoted by white arrows).

the "Water is not boiling" is 4; thus, we do not need to perform further analysis of this deviation. As we can determine the severity of the deviation from only one global effect, the number of deviations to be analyzed is one. Under ID7, by applying the *Incorrect transition* guideword, we can derive the deviation whereby "In the *Heating* state, when the water temperature reaches boiling point, the state transitions to *Removing chlorine* (only deviation is that the heater is off)." Thus, unboiled water may come out of the pot. The system state in the orthogonal STD is either *Supplying hot water* or *Not-supplying hot water*. In the former case, the pump is open and unboiled water is emitted when the user expects hot water. The global effect is that the user does not get scalded, and is thus safe. The severity of *Unboiled water comes out* is 7; the cause of this deviation is "Heater failure" or a "Memory failure in heater," as the heater is off, and the countermeasure for this situation is a "Duplicate heater" or "Memory protection by MMU or MPU" (Memory Management/Protection Unit). In the *Not-supplying hot water* state, the pump is closed, and so no water comes out. The global effect is, once again, that the user does not get scalded. The severity of "Water is not boiled" is 4, so we see that the combination of global effects and severity levels differs according to the system state in the orthogonal STD. In this case, we can determine the severity of the deviation from two global effects; thus, there are two deviations to be analyzed.

Partial state deviation results from SAPSTD are presented in Table 6. ID3 represents a functional specification for "Pump is closed in *Not-supplying hot water* state." By applying the guideword *Value*, the "Pump is open" devia-

Table 7. Partial results of the analysis of deviations in the CSTD.

ID	Item	Deviation	Local effects	Global effects	Severity	Causes	Countermeasures
56	(C.5) In the *Heating and Not-supplying hot water* state, when the water temperature reaches boiling point, the state transitions to *Removing chlorine and Not-supplying hot water*.	Incorrect action In the *Heating and Not-supplying hot water* state, when the water temperature reaches boiling point, the state transitions to *Removing chlorine and Not-supplying hot water*.(boiling indicator, keeping heat indicator, and heater are working correctly; only deviation is that the pump is open)	Boiled water comes out at an unexpected time.	-Scalded -Boiled water comes out	9	-Pump failure -Memory failure in pump	-Duplicate pump -Memory protection by MMU or MPU

Table 8. Comparison of the number of deviations analyzed by the two methods.

	SASTD for CSTD		SAPSTD for PSTD			
	Number of deviations	Number of deviations with countermeasures	Number of deviations	Number of deviations with countermeasures	Number of deviations whose analysis can be omitted	Fewest number of deviations to be analyzed
Analysis of states	36	5	36	10	24	12
Analysis of state transitions in *Heating* mode	78	9	54	6	21	33
Analysis of state transitions in *Supplying hot water* mode	66	8	20	2	4	16
Total	180	22	110	18	49	61

tion can be derived. If the pump is open, water may come out at an unexpected time. When this deviation occurs, the system state in the orthogonal STD, i.e., the *Heating* mode in Figure 6, is either *Heating*, *Removing chlorine* or *Keeping heat*. In this case, the global effects and their severity are the same regardless of the system state and the state transition in the orthogonal STD. The severity of the global effect of boiling water coming out at an unexpected time is 9, thus we do not need to analyze this deviation further. At first, there are three deviations to be analyzed (states: *Heating*, *Removing chlorine* or *Keeping heat*). We need only analyze the fewest deviations, regardless of states; in this case, there is only one deviation to be analyzed. Thus, the analysis is more effective.

Partial results from the CSTD analysis are presented in Table 7. ID56 represents a functional specification for (C.5) "In the *Heating and Not-supplying hot water* state, when the water temperature reaches boiling point, the state transitions to *Removing chlorine and Not-supplying hot water*." By applying the *Incorrect action* guideword, the deviation shown in Table 7 is derived. When the "Pump is open" deviation is the only one to occur in the *Heating and Not-supplying hot water* state, the heater in the system shown in Figure 7 works correctly; thus, the local effect is "Boiling water comes out at an unexpected time." In this case, the severity of the global effect of *Boiling water comes out at an unexpected time* is 9. The cause is either "Pump failure" or "Memory failure in pump," and the countermeasures are to provide a "Duplicate pump" or supply "Memory protection by MMU or MPU." In this case, we can determine the severity of the deviation from only one global effect; thus, the number of deviations to be analyzed is one.

5 Discussion

The number of deviations analyzed by SASTD for the CSTD and by SAPSTD for the PSTD are compared in Table 8. For the CSTD, there 36 state deviations and 144 state transition deviations, giving a total of 180 deviations to be analyzed by SASTD. In comparison, for the PSTD, it was necessary to analyze 36 state deviations and just 74 state transition deviations, giving a total of 110 deviations for SAPSTD. Because there are fewer states and state transitions in the PSTD than the CSTD, the analytical area of the PSTD is also narrower than that of the CSTD. Thus, using SAPSTD, a reduction of 70 deviations was achieved.

Deviations with countermeasures imply an original severity score of 7 or more. The total number of such deviations in SASTD and SAPSTD were 22 and 18, respectively. We compared the worksheets of both methods in detail, and confirmed that all deviations derived by SASTD could also be analyzed by SAPSTD. Thus, the completeness of both methods is the same.

In SAPSTD, the number of deviations can be further reduced. As mentioned in Section 4.4, if we can specify the global effects in the worst case, regardless of the states and state transitions in orthogonal STDs, the analysis of the global effects for each state and state transition in orthogonal STDs is omissible. In this case study, there were 49 such deviations in total, meaning that the only 61 deviations require analysis in SAPSTD.

As explained in Figure 8, we assumed that the analytical area of deviations derived from *Incorrect transitions* was limited. Techniques to limit the analytical area are useful for reducing the burden of the safety analysis. For example, variables for the state management and functions for device access for each STD should be implemented by different tasks. Moreover, memory protection by MMU or MPU is highly effective in preventing invalid access due to bugs in one task affecting devices managed by other tasks. Using such techniques for the limitation of the accessible area, the analytical scope of deviations can be reduced.

6 Conclusion

This paper has described the SAPSTD technique for safety analysis using PSTDs. We compared SAPSTD with the earlier SASTD method and confirmed that all deviations derived by SASTD could also be derived by SAPSTD. In addition, we have clarified that the number of deviations to be analyzed can be reduced in SAPSTD by limiting their analytical area. Therefore, SAPSTD has been effectively tested and its performance was shown to be very promising. It is desirable to compare SAPSTD with other analysis methods in order to improve the completeness of embedded systems' safety analysis. In future work, we will apply SAPSTD to a complex embedded system.

References

[1] IEC, *IEC 61508 : Functional Safety of Electrical/electronic/programmable Electronic Safety-related Systems, Part 1–7*, 2000.

[2] ISO, *ISO 26262 : Road vehicles - Functional safety -, Part 1–9*, 2011.

[3] Ministry of Economy, Trade and Industry, *Current survey of selected service industries of embedded system: Survey of Project Management Reference*, 2010. (In Japanese)

[4] D. Harel, Statecharts: A visual formalism for complex systems, *The Science of Computer Programming*, 1987, 231-274.

[5] K. Lano, D. Clark, and K. Androutsopoulos, Safety and security analysis of object-oriented models, Computer Safety, Reliability and Security: SAFECOMP, *In 21st International Conference on Computer Safety, Reliability and Security*, Catania, Italy, 2002, 82-93.

[6] Z. Kim, Y. Matsubara, and H. Takada, A safety analysis method based on state transition diagram, *IEICE TRANSACTIONS on Fundamentals of Electronics, Communications and Computer Sciences*, Vol.J95-A, No.2, 2012, 198-209. (In Japanese)

[7] D.H. Stamatis, *Failure Mode and Effect Analysis*, Milwaukee: ASQ Quality Press, 2003.

[8] Z. Kim, Y. Matsubara, and H. Takada, Safety analysis method based on hierarchical state transition diagram, *A workshop related to embedded and network technologies, Information Processing Society of Japan*, Matsushima, Japan, 2012. (In Japanese)

[9] Imperial Chemical Industries, Ltd., Chemical Industries Association. Chemical Industry Safety and Health Council, *A guide to hazard and operability studies*, The Pennsylvania State University: Chemical Industry Safety and Health Council of the Chemical Industries Association, 1977.

[10] D.J. Pumfrey, The Principled Design of Computer System Safety Analyses, *PhD Thesis*, University of York, 1999.

[11] Q.A. Do Hoang, J. Guiochet, D. Powell, and M. Kaaniche, Human-robot interactions : model-based risk analysis and safety case construction, *In 6th European Congress on Embedded Real-Time Software and Systems*, Toulouse, France, 2012.

[12] T. Ishimatsu, N. Leveson, J. Thomas, M. Katahira, Y. Miyamoto, and H. Nakao, Modeling and Hazard Analysis using STPA, *Conference of the International Association for the Advancement of Space Safety*, Huntsville, Alabama, 2010.

[13] N.G. Leveson and J.L. Stolzy, Safety Analysis Using Petri Nets, *IEEE Transactions on Software Engineering*, Vol. SE-13, No. 3, 1987, 386-397.

[14] Society of Embedded Software Skill Acquisition for Managers and Engineers, editor, *Specification of Boiling Jar (GOMA-1015) 7th edition*, Society of Embedded Software Skill Acquisition for Managers and Engineers, 2005. (In Japanese)