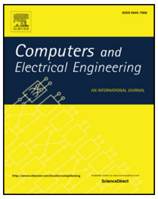
Computers and Electrical Engineering 71 (2018) 162–172

Contents lists available at ScienceDirect

Computers and Electrical Engineering

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



Petri Nets-based method to elicit component-interaction related safety requirements in safety-critical systems

Danjiang Zhu[a], Huobin Tan[b,*], Shuzhen Yao[a]

[a] School of Computer Science and Engineering, Beihang University, Beijing, China
[b] School of Software, Beihang University, Beijing, China

ARTICLE INFO

Keywords:
System safety
Software safety requirements
Requirements elicitation
Petri nets
Dysfunctional interaction
STAMP

ABSTRACT

System component-interaction has been critical for system safety, accompanied by the ever-growing complexity of safety-critical systems. As a novel causality model, Systems-Theoretic Accident Modeling and Process (STAMP) has been well used to obtain component-interaction related safety requirements. However, the original process model used in STAMP can't specify component-interactions clearly, which limits the component-interaction related requirement elicitation. Petri nets, which are effective tools to model complex systems, can help analyze component-interactions and make the safety requirement elicitation with STAMP effectively. This paper extends traditional Petri Nets, Control Logical Petri Net (CLPN), to model and analyze component-interactions in the control process. Then four kinds of basic dysfunctional interactions that can lead to system hazards are described with CLPN. Additionally, several rules are listed to guide dysfunctional interactions analysis with CLPN. Based on the studies above, an integrated approach eliciting safety requirement is proposed. The experimental results illustrate the feasibility and availability.

1. Introduction

The ever growing complexity and coupling of safety-critical systems have led the dysfunctional interaction among system components to be an important causal factor of accidents [1,2]. System component-interaction related safety analysis is attracting widespread interest in various areas. And the Component-Interaction related Software Safety Requirements (CISSRs), which act as an important part of software safety requirements, are considered to be essential for system safety [3,4].

In software engineering, Feature-Oriented Domain Analysis (FODA) [5] treats dysfunctional interactions as the feature interaction problem to obtain unexpected hazardous interactions [6,7]. However, most feature-oriented methods require formal feature interaction models, and the combination of these methods and accident causality models used in CISSRs elicitation remains to be further researched.

Safety analysis is one of the most important ways to elicit software safety requirements [8], including CISSRs. According to the accident causality model, the obtained causal factors could be translated into CISSRs. Unfortunately, most traditional failure-based safety analysis approaches, such as Fault Tress Analysis (FTA), Failure Modes and Effect Analysis (FMEA), are based on event-chain accident models, which assume that accidents are caused by component failures [1,9], but they do not account for component-interaction accidents and do not work well for CISSRs elicitation [10]. The reason of this problem is that dysfunctional interactions, which serve as an important accident causal factor for software safety requirements elicitation [2], could occur among non-failing

[*] Corresponding author.
E-mail address: thbin@buaa.edu.cn (H. Tan).

https://doi.org/10.1016/j.compeleceng.2018.07.019
Received 16 May 2018; Received in revised form 24 July 2018; Accepted 24 July 2018

components and can't be simply explained using failure event. In order to address this issue, Leveson developed a new causality model named Systems-Theoretic Accident Modeling and Processes (STAMP) [4]. In STAMP, accidents such as component-interaction accidents and component failure accidents, are conceived as the results of inadequate control or enforcement of safety-related constraints on the behavior of components in system life cycle. So accidents result from dysfunctional interactions among components can be eliminated by enforcing appropriate component-interaction constraints on the control actions of components in the control structure of STAMP. Based on STAMP, System Theoretic Process Analysis (STPA) method [11] is developed to identify a wider range of hazard causal factors and derive more complete software safety requirements in various areas [9,12]. However, STPA's application is ad-hoc with no rigorous procedures to guide the analysis, and the process model of system control structure used by STPA is too simple to specify the component-interaction, that limit the CISSRs elicitation with STAMP.

In order to identify the hazardous interaction that result from inadequate constraints of control action, interactions in the control structure of STAMP should be modeled and analyzed. Although there are lots of component interaction modeling and analyzing tools such as Component Interaction Automata and Interaction Overview Diagram in software engineering, most of these tools only specify the interaction and they always need to work together with model checkers in analyzing the interaction. Unlike those above-mentioned tools for component interaction specification, Petri Nets are well-founded models applied to simulate and analyze the behavior of concurrent systems [12,13], as well as the concurrency and uncertainty of component interactions, and the analysis of Petri nets does not depend on the model checking tools. Therefore, Petri Nets [14] could identify dysfunctional interactions through modeling and analyzing complex interactions in the control structure of STAMP, especially when there are multiple controllers control the same process. Petri Nets have been used to determine the critical software functions for such properties as safety and fault-tolerance along with FTA, FMEA and so on [15]. However, these safety analyses with Petri Nets are based on event-chain accident models and hardly applicable for dysfunctional interaction identification and CISSRs elicitation. Consequently, an efficient method based on Petri Nets is needed to analyze dysfunctional interactions and elicit related software safety requirements with STAMP.

Aimed to accomplish this goal, we presented a novel CISSRs elicitation method with an extended Petri Net named Control Logical Petri Net (CLPN). CLPN was proposed in this study to model the interactions among control actions in the control process of system control structure. Then, the interactions were analyzed in the CLPN to identify the dysfunctional interactions which lead to the coordination risks of STAMP. We investigated when and how component-interactions can lead system to hazardous states, and four kinds of dysfunctional interactions were found. Afterwards, the hazardous dysfunctional interactions elicited from the CLPN were further used to generate the safety constraints on control action as CISSRs based on STAMP.

The rest parts of this paper are organized as follows. Section 2 describes the methodology of component-interaction related safety requirement elicitation proposed in this paper. An extended Petri Net is proposed to model the component-interactions in system control structure in Section 2.1; the dysfunctional interactions of STAMP are specified with CLPN and further to be analyzed with the reachable tree of CLPN for safety requirement obtainment in Section 2.2. Experimental results with detailed analysis are discussed in Section 3. Finally, some concluding remarks along with the scope of future works are given in Section 4.

2. Methodology

According to STAMP, systems are viewed as hierarchical control structures, and control processes operate throughout the hierarchy with control actions to impose constraints on the lower levels. In the system control structure, especially when there are multiple components as controllers, constraints that are adequate for individual control actions may be inadequately coordinated for the interactions among these components and lead to system hazards. So interactions among these components should be examined to ensure the system safety. To achieve this, we proposed an extended Petri Net to describe the interactions in control process based on STAMP and elicit the dysfunctional interactions from the extended Petri Net models with the purpose of generating the related

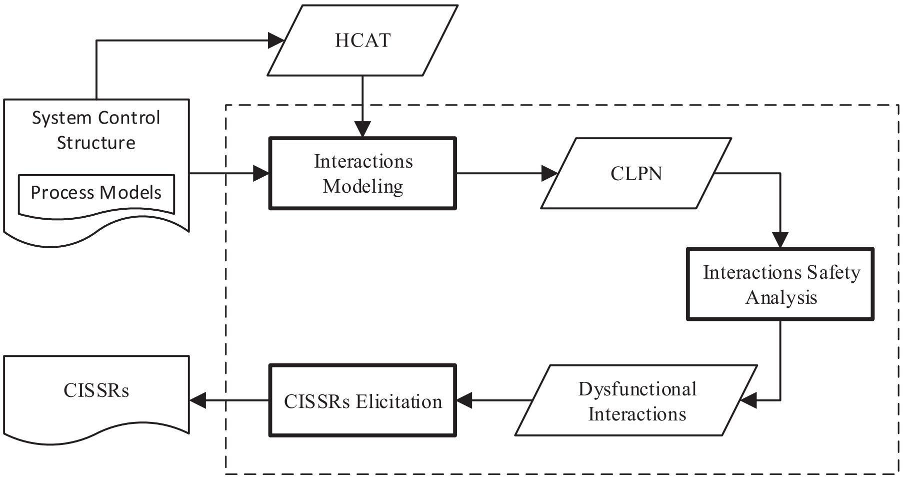


Fig. 1. The framework of CISSRs elicitation method.