ISSN 2383-630X(Print) / ISSN 2383-6296(Online)
Journal of KIISE, Vol. 48, No. 1, pp. 107-118, 2021. 1
https://doi.org/10.5626/JOK.2021.48.1.107

가상물리시스템의 산출물간 통합 관계분석을 위한 추적성 분석

(A Traceability Analysis for Integrated Relationship Analysis of Development/Safety Artifacts of Cyber Physical Systems)

정 세 진 † 김 의 섭 † 유 준 범 ††
(Sejin Jung) (Eui-Sub Kim) (Junbeom Yoo)

요 약 안전성이 중요한 가상물리시스템의 개발은 추적성 분석을 통해 산출물의 관계에 대해 확인하는 것이 필요하다. 가상물리시스템은 상호운용성, 동적 상황 변경, 이종 연결성 등의 특징이 있어 개발 산출물과 안전성/위험 분석 요소를 통합해 분석할 필요가 있다. 하지만 추적성 관계를 일괄적으로 연결하고 분석하는 것으로는 각 요소별 추적성 및 그 관계의 특징을 모두 표현하기에 한계가 있다. 본 논문에서는 가상물리시스템의 개발 산출물 및 안전성/위험 분석 요소의 통합적인 관계분석을 위한 추적성 분석 방법 및 가상물리시스템의 특징에 따른 추적성 세부 관계를 제안한다. 본 논문에서는 이를 위해 가상물리시스템의 각 산출물 요소에 대한 추상화 모델을 제안하고 각 요소별로 추적성을 분석하며 세부 관계를 부여한다. 본 논문에서 제안하는 방법을 통해 가상물리시스템의 각 산출물 요소의 추적성 관계를 분석하고 연결할 수 있으며, 통합 관계를 확인할 수 있다.

키워드: 추적성 분석, 가상물리시스템, 안전성 분석 요소 모델, 개발 산출물 모델

Abstract A cyber-physical system (CPS), that is to be used a safety important system, needs to analyze the traceability of development artifacts. The traceability analysis of the CPS should be performed integrating development artifacts and safety/hazard analysis elements because CPS has several features such as heterogeneity, dynamic reconfiguration, and interoperability. However, there is a limitation in terms of expressing all traceability relationships by identically connecting and analyzing the traceability between development artifacts and safety analysis elements. This paper proposes an analysis method and relationships of traceability for CPS. The proposed method uses an abstract model for development artifacts and safety analysis elements that are defined in this paper. The traceability relationships define the relations between elements of the model. The proposed method makes it possible to analyze integrated relationships from development artifacts and safety/hazard analysis elements. The case study shows integrated relationships according to each element of several artifacts.

Keywords: traceability analysis, cyber-physical system, safety analysis element model, development artifact model

· 이 논문은 2017년도 정부(과학기술정보통신부)의 재원으로 한국연구재단-차세대정보컴퓨팅기술개발사업(NRF-2017M3C4A7066479)의 지원을 받아 수행한 연구임

† 학생회원 : 건국대학교 컴퓨터공학과 학생
jsjj0728@konkuk.ac.kr
atang34@naver.com
†† 정 회 원 : 건국대학교 컴퓨터공학과 교수(Konkuk Univ.)
jbyoo@konkuk.ac.kr
(Corresponding author임)

논문접수 : 2020년 10월 13일
(Received 13 October 2020)
논문수정 : 2020년 11월 18일
(Revised 18 November 2020)
심사완료 : 2020년 11월 23일
(Accepted 23 November 2020)

정보과학회논문지 제48권 제1호(2021. 1)

1. 서 론

소프트웨어 시스템 개발은 요구사항 분석부터 디자인, 구현 및 테스팅에 이르는 개발 프로세스에 따라 진행된다[1]. 안전성이 중요한 시스템의 경우 IEC 61508[2], ISO 26262[3] 등과 같은 국제 표준에 따라 안전생명주기가 적용되어 위험 분석을 통해 시스템/소프트웨어의 잠재적인 위험을 식별하고 이를 완화하기 위한 안전 요구사항을 할당하는 과정들이 적용된다. 안전성이 중요한 시스템의 개발에서는 요구사항이 디자인 및 구현에 잘 반영되었는지 확인하고, 위험 분석을 통해 도출한 안전 요구사항이 실제 구현되었는지에 대해 확인이 필요하며 추적성 분석을 통해 해당 관계들을 확인하는 것이 필수적이다[4].

추적성 분석은 predecessor-successor 관계에 있는 둘 이상의 산출물 사이의 관계를 분석하여 연결하는 프로세스이다[5,6]. 일반적으로 소프트웨어/시스템 개발에서 요구사항과 계층관계에 있는 다른 요구사항이나 설계, 디자인, 구현 등과 같은 다양한 산출물의 관계를 분석하는 것이라 할 수 있다. 추적성 분석은 주로 설계의 변경으로 인한 영향을 분석하거나, 요구사항에 대한 설계 및 구현여부의 분석, 또는 소프트웨어 유지보수에 사용된다[7].

가상물리시스템은 디지털 환경과 물리적 세계를 통합하는 개념의 시스템으로 센서, 네트워크를 통해서 실제의 물리 환경을 제어하는 시스템이다[8]. 이러한 가상물리시스템 또한 적용되는 도메인이나 안전 중요도에 따라 안전필수 시스템으로 분류된다. 따라서 안전성이 중요한 가상물리시스템의 개발은 개발 생명주기 및 안전생명주기를 따라 진행되며 개발 산출물과 안전성 분석 요소를 통합해 추적성 분석을 수행해야 한다. 하지만 가상물리시스템은 상호운용성, 동적 상황 변경, 이종 연결성 등의 특징을 가지고 있어 개발 산출물 및 안전성 분석 요소를 일괄적으로 연결하고 분석하는 것으로는 그 추적성 관계를 모두 표현하기에 한계가 있다. 특히 개발 산출물에 표현되는 동적 상황 변경, 상호 연결성에 대해 안전성 분석 요소와의 세부 관계를 확인하여 각 요소별로 추적성을 연결할 필요가 있으며 연결 요소들 및 관계에 대한 통합적인 분석이 필요하다[9].

본 논문에서는 가상물리시스템의 개발 및 안전성/위험 분석 산출물의 통합적인 관계분석을 위한 추적성 분석 방법 및 추적성 세부관계를 제안한다. 이를 위해 가상물리시스템의 개발 산출물 및 안전성/위험 분석 요소에 대한 추상화 모델을 정의하였다. 산출물 모델은 개발 산출물을 위한 모델과 안전성/위험 분석을 위한 모델로 나누어 구성되며, 추적성 분석은 제안한 모델의 각 요소들 사이의 추적성을 연결하고 그 관계에 따라 세부 관계를 부여하며 수행된다. 추적성 세부 관계는 가상물리시스템의 특징에 따라 정의된다. 본 논문에서 제안하는 방법을 통해 가상물리시스템의 소프트웨어 및 시스템 개발의 각 산출물과 안전성/위험 분석 결과의 데이터를 모델링하고 요소별 추적성을 분석할 수 있다. 또한 추적성 연결에 세부 관계를 부여함으로써 가상물리시스템의 특성에 따른 산출물의 관계를 통합적으로 분석할 수 있다. 본 논문에서 제안하는 모델 및 추적성 분석에 따라 군집주행 시스템(platoon system)[10]을 대상으로 수행한 사례 연구에 대해 소개한다.

본 논문의 구성은 다음과 같다. 먼저 2장에서는 배경지식 및 관련연구에 대해 설명하고, 3 장에서는 본 논문에서 제안하는 추적성 분석에 대해 소개한다. 4장에서는 군집주행시스템을 대상으로 수행한 사례 연구에 대해 소개하고, 5장에서 결론 및 향후 연구에 대해 설명한다.

2. 배경 지식

2.1 위험 분석(hazard analysis) 기법

위험 분석(hazard analysis)은 시스템 및 소프트웨어의 사고, 실패가 될 수 있는 잠재적인 위험 상태를 분석하는 것으로 원인 분석, 결과 분석, 그 영향을 경감시키기 위한 방법 분석 등이 포함된다[7]. 현재 사용되는 위험 분석 기법으로는 컴포넌트의 원인으로부터 그 결과의 영향을 분석하는 FMEA (failure mode and effect analysis), HAZOP (hazard and operability) 연구 등과 어떤 사고로부터 그 원인을 분석하는 FTA (fault tree analysis) 등이 있다. 또한 비교적 최근에 개발된 시스템의 상호 작용에 의한 문제를 분석하는 STPA (system-theoretic process analysis)[11]도 많이 사용된다.

FMEA는 시스템을 구성하는 item 또는 function의 failure mode (고장 모드)가 미치는 영향을 분석하고, 발생 빈도, 위험성에 따른 위험도를 수치화 하는 기법으로 표 1과 같은 worksheet을 이용해서 수행된다. 반면 FTA는 사고의 원인이 되는 사건들을 논리 게이트로 연결하

표 1 FMEA worksheet 표 예제[7]
Table 1 Example of FMEA worksheet

Failure Model and Effects Analysis								
System: Subsystem: Component:								
Item	Failure mode	Cuase	Intermediate effect	System effect	Method of detection	Hazard	Risk	Recomm action

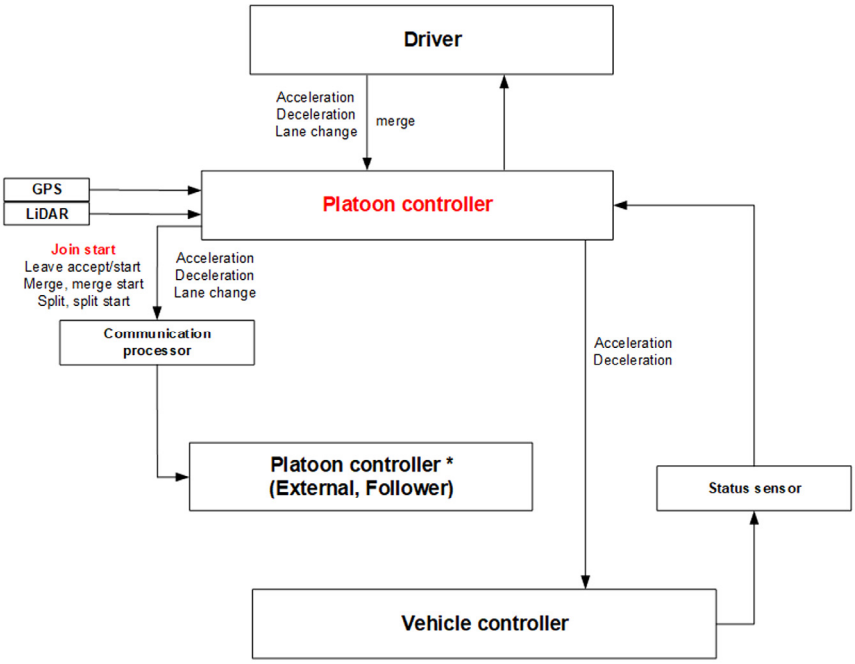

그림 1 군집주행 시스템에 대한 STPA control structure
Fig. 1 An STPA control structure of platoon system

여 분석하는 방법을 취한다. STPA는 이러한 전통적인 기법과 달리 STAMP (system theoretic accident model and process) 모델을 기반으로 컴포넌트간의 상호작용, 사람, 관리 요소들이 만든 오류와 같은 문제에 초점을 둔 분석 방법이다[11]. STPA는 제어 구조(control structure)라 불리는 시스템 모델을 작성하고 각 컴포넌트의 제어 명령(control action)에서 생길 수 있는 위험을 식별하고 분석한다. 그림 1은 STPA의 control structure에 대한 예제 그림으로 controller와 controller의 제어를 받는 controlled process 그리고 사이의 제어 명령에 대해 구조적으로 작성한다. STPA에서는 이 때 각 제어 명령에 대해 'providing causes hazard,' 'not providing causes hazard,' 'too soon/too late,' 'stopped too early, too late' 와 같은 가이드 워드를 적용하여 hazardous한 상황을 분석한다.

2.2 관련 연구

추적성 분석은 predecessor-successor 또는 master-subordinate 관계를 갖는 둘 이상의 개발 산출물 사이의 관계를 분석하여 연결하는 프로세스이다[5]. 안전필수 시스템에서는 개발 산출물 외에 안전 요구사항 등 연관 관계가 존재하는 요소들의 관계 분석에도 적용될 수 있다. 추적성 분석 결과는 변경이 발생할 때 영향 범위를 확인하거나 테스팅 범위를 확인하는데 사용될 수 있으며, 또한 국제 표준에 따른 안전성 증명의 한 자료로 사용될 수도 있다. 이러한 추적성 분석은 요구사항분석부터 시작하는 개발 산출물 간의 관계 이외에도 연결들이 나타날 수 있어 여러 연구가 진행되었다[4,12].

[5]에서는 ARINC-653 표준을 기반으로 개발된 실시간 운영체제 소프트웨어 개발 산출물들의 추적성을 분석하고 그 세부 관계를 정의하였다. 해당 논문에서는 표준으로부터 정의된 요구사항 및 구현의 결과물을 분석하며 필요한 표준의 준수 및 구현을 초점으로 두고 추적성을 분석하였다. [4]에서는 안전필수 시스템의 개발 과정에서 필수적으로 도출되고 반영되어야 하는 안전 요구사항에 대해 산출물 간의 추적성을 분석하였다. 특히 안전성의 증명을 위한 safety case의 evidence를 위해 안전 요구사항 및 개발 산출물 간의 같은 development phase, 서로 다른 development phase로 추적성을 분석하였다.

[12]에서는 IEC 61508 표준을 바탕으로 소프트웨어기반 시스템 개발에 개발 산출물 및 안전성 분석 프로세스의 데이터에 대한 개념적인 추적성 모델을 제안하였다. 해당 모델은 소프트웨어 개발 프로세스의 단계와 안전성 평가 단계의 각 활동에 대해 산출되는 데이터를 정의하고 각 데이터들 간의 연관성을 표현한다. 각 단계별로 산출되는 여러 결과물 중 추적성 분석에 필요한

요소와 그 관계를 정의하였다. 하지만 각 산출물의 연결에서 타입에 대한 고려가 모호한 점이 있다. [13]에서는 safety case를 이용한 safety argument를 위해 추적성 분석을 통해 분석한 정보를 사용하는 방법을 제안한다. 이를 위해 [12]에서 제안한 개념 모델을 메타모델로 작성하고 이를 바탕으로 safety argumentation에 필요한 정보를 추출한다. 메타모델은 [12]에서 제안한 모델에 relation type이 포함된 모델로 작성된다.

[14]는 가상물리시스템의 모델링, 시뮬레이션 등을 포함한 도구의 셋을 제안하였다. 추적성 분석은 요구사항, 디자인 모델 간의 추적성 분석을 제공하며 multi-model 등에 대한 고려를 포함하며 의미적 디자인 관계를 기반으로 추적성을 분석한다. [15]에서는 요구사항 추적성 분석과 관련된 여러 도구들 및 기법에 대해 소개하고 있다. [15]에 따르면 요구사항 추적성 분석을 효과적으로 수행하고 시각화 하는 다양한 모델, 도구들이 개발되어 사용되고 있다.

이처럼 여러 기존의 연구들이 존재하지만 가상물리시스템의 특징을 반영하여 추적성을 분석하고 세부 관계를 정의하는 등에 대한 연구는 아직 부족한 상황이다. 또는 개념적인 모델로만 구성되어 있어 가상물리시스템을 모두 반영하기엔 차이가 존재한다. 본 논문에서는 이에 대해 가상물리시스템의 효과적인 추적성 분석을 위해 개발 산출물 및 안전성 분석 요소의 추상화 모델을 제안하고 모델의 각 요소별 추적성 분석의 세부 관계를 정의하여 개발 산출물 요소 및 안전성 분석 요소의 관계를 통합적으로 확인할 수 있도록 한다.

3. 가상물리시스템 산출물의 추적성 분석

가상물리시스템의 개발은 개발프로세스 및 안전 중요도에 따라 안전생명주기가 적용된다. 따라서 가상물리시스템의 산출물에는 개발과 안전성/위험 분석 결과가 모두 포함되며 전체적인 관계를 통합적으로 분석할 필요가 있다. 본 논문에서는 이에 따라 각 산출물 간의 통합적인 추적성 관계분석을 위해 산출물의 요소를 추상화한 모델과 추적성 분석 및 가상물리시스템의 특징에 따른 추적성 세부 관계를 제안한다. 3.1장에서는 추적성 분석을 위한 산출물 요소의 추상화 모델을 설명하고 3.2장에서는 제안하는 모델을 활용한 추적성 분석 및 세부 관계에 대해 설명한다.

3.1 개발 및 안전성 분석 산출물 요소의 추상화 모델

가상물리시스템을 이용한 안전 시스템 개발에 적용되는 안전생명주기에 따른 주요 산출물은 안전성/위험 분석과 이를 경감하기 위한 안전 요구사항이 있다. 본 논문에서는 우선 추적성 분석의 대상이 될 수 있는 개발 산출물의 관계에 대해 확인하였다. 그림 2는 개발 프로세스와 기능안전성 표준의 안전 생명주기와의 각 과정별 연관 관계를 나타낸 그림으로 각 생명주기 내부에서의 추적성 분석과 생명주기 사이의 추적성 분석이 필요한 부분을 분석하였다. 특히 안전성 분석의 3 번 단계인 hazard and risk analysis를 통해 도출한 안전 요구사항을 할당하고 개발하는 과정에서 개발 산출물과의 밀접한 추적성 분석이 필요하다.

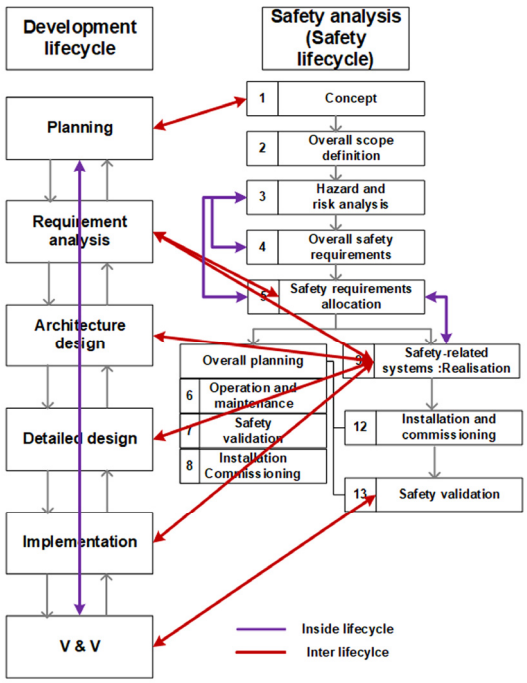

그림 2 추적성 분석을 위한 개발 생명주기와 안전 생명주기의 연관 관계 확인

Fig. 2 Relationships between development process and safety lifecycle for traceability analysis

추적성 분석을 통합적으로 표현하기 위해서는 우선 분석의 대상이 될 수 있는 요소들을 표현할 필요가 있다. 본 논문에서는 이를 위해 개발 및 안전성/위험 분석 산출물을 추상화 한 모델인 산출물 요소 모델(DEM: development element model)과 안전성 분석 요소 모델(SAEM: safety analysis element model)을 정의하였다. DEM은 계층 구조를 갖는 3 단계로 구성하여 추적성 분석에 필요한 요소들을 모델로 나타낼 수 있다. 안전성 분석 요소는 현재 위험 분석에 주로 적용되는 4가지 기법인 FMEA, STPA, FTA, ETA를 대상으로 모델을 정의하여 개발 대상이 되는 시스템의 안전성/위험 분석의 결과를 모델링 할 수 있도록 구성하였다.

요구사항 명세서의 구조는 일반적으로 IEEE 830 *"IEEE Recommended Practice for Software Requirements*

Specifications [16]"과 같은 표준에 따라 문서 구조, 목차 구조, 기능 명세로 구성된다. 이 중 기능 명세가 작성하는 요구사항의 가장 작은 단위라 할 수 있으며, 실제 추적성 분석에 연결되는 직접적인 단위라 할 수 있다. 마찬가지로 요구사항 이후의 개발 단계도 기능 단위로 유사하게 표현할 수 있다. 본 논문에서는 문서 구조, 목차 구조, 기능 명세를 각각 phase structure, artifact item structure, element로 정의하였고, 각각의 항목에 고유 id를 부여해 추적성을 분석하는 기본 단위로 하였다.

Definition 1 (Phase Structure). A phase structure defined as a PhS = <id, Pt, AI, E, D>
- Id: a unique id of the PhS
- Pt: a type of system/software development phases, Let {PFR, SRS, SDS(A), SDS(D), I, TP} be the types that are 'preliminary functional requirements, 'software requirements specification,' 'software design specification (architectural),' 'software design specification (detailed),' 'implementation,' 'testing plan' respectively (That is ∀ Pt ∈ {PFR, SRS, SDS(A), SDS(D), I, TP, SA})
- AI: a set of artifact item structures that are constructed phase structure, {ai_1, ai_2, …}
- E: a set of Elements in PhS, {e_1, e_2, …} (optional)
- D: a simple explanation of current phases, {d_1, d_2, …}

Definition 2 (Artifact Item Structure). An artifact item structure is defined as AI = <id, name, D, AI, E>, where
- id: a unique of the item structure
- name: a subject/name of chapter in the structured artifact that is phase structure
- D: a simple explanation of process structure (optional), {d_1, d_2, …}
- AI: a set of artifact item structure that are constructed hierarchically, {ai_1, ai_2, …}
- E: a set of Elements that are included in AI, {e_1, e_2, …}

Definition 1은 phase structure에 대한 설명으로 소프트웨어/시스템 개발에서 요구사항 분석, 디자인, 구현 등과 같은 개발 생명주기의 한 단계 (phase) 전체에 대한 산출물을 표현하기 위한 모델이다. Phase structure는 개발 phase의 타입을 나타내는 Pt와 상세 설명을 위한 D 그리고 하위 아이템을 구성하는 artifact item structure 모델 및 element 모델을 갖도록 구성된다. 다음으로 Artifact item structure는 각 개발 산출물 내부의 목차구조를 모델링하기 위한 모델로 Definition 2는 artifact item structure에 대한 설명이다. Artifact item structure의 구성 요소는 각 챕터 수준의 제목을 정의하는 name과 구체적인 설명인 D가 포함된다. 또한 각 챕터의 수준이 여러 단계로 구성될 수 있기 때문에 재귀적인 구조로 구성된다. Artifact item structure 모델을 이용해 한 development phase 구성을 모델링 하여 개발 문서의 전체적인 구조와 목차 등의 형태를 표현한다.

Definition 3은 DEM의 가장 작은 단위를 표현하기 위한 정의로 본 논문에서는 element라 한다. Element는 각 산출물의 구체적인 명세를 작성하기 위한 구성이다. 예를 들어 요구사항 명세 및 디자인 단계에서는 각 세부 기능의 요구사항/디자인 명세가 대상이 되고, 테스팅 계획에서는 각 기능별 테스트 케이스가 그 대상이 될 수 있다. Element는 표, 다이어그램, 텍스트와 같은 작성되는 타입과 대상의 이름을 나타내는 name, 그리고 실제 명세를 나타내는 D로 구성되어 있다. 추적성 분석에서는 element로 정의된 산출물 요소들이 하나의 노드 형태로 이용된다.

그림 5는 본 논문에서 정의한 phase structure (PhS), artifact item structure (AI) 및 element를 이용하여 개발산출물을 구조에 맞게 모델링한 예시이다. 그림 5에 나타난 바와 같이 PhS 수준에서는 제목인 'SRS for platoon controller of platooning system'과 구조를 나타내는 AI를 가지고 있으며 각 챕터구조에 따라 element를 구성 요소로 가진다.

Definition 3 (Element). An element is a leaf node of artifacts, process of software/system development phases. It is defined as a tuple of E = <id, type, name, D, E>
- id: a unique id of the element
- type: a type of the elements, {table, diagram, texts}
- name: a representative subject/function name of the E
- D: a detailed contents of node elements, it is determined by the type of the E, {d_1, d_2, …}
- E: a set of Elements that are included in E hierarchically, {e_1, e_2, …}

가상물리시스템의 안전생명주기와 관련된 결과물들을 모델링하기 위한 SAEM은 전체 구조와 위험 분석 두 종류로 구성된다. 위험분석의 결과는 개발 산출물과는 다른 형태로 존재하기 때문에 본 논문에서는 대표적으로 사용되는 4 종류의 위험 분석 기법인 FTA (fault tree analysis), ETA (event tree analysis), STPA (system-theoretic process analysis), FMEA (failure mode and effect analysis)를 위한 모델을 정의하였다. Definition 4는 수행된 안전분석의 전체 구조를 모델링하기 위한 정의로 위험 분석의 결과를 정의하는 H와 안전 요구사항을 정의하는 SR로 구성된다. 안전요구사항은 각 위험분석의 결과를 통해 도출되는 내용들로 해당 안전 요구사항들이 개발 산출물과의 추적성 분석에 중요한 요소가 된다. H는 각 위험분석 기법을 사용한 결과에 대한 모델로 본 논문의 Definition 5~8에 정의된

FMEA, STPA, FTA, ETA에 대한 모델을 의미한다.

Definition 4 (Safety Analysis Structure). A safety analysis structure is defined as a tuple of SAS = <id, H, SR>
- id: a unique id of safety analysis structure
- H: a hazard analysis technique that are applied to system {provided that ∀ H ∈ {*FMEA, STPA, ETA, FTA*}, {h_1, h_2, …},
- SR: a set of safety requirements derived by safety/hazard analysis, {sr_1, sr_2, …}

Definition 5 (FMEA). An FMEA model, it is defined as a tuple of FMEA = <id, TC, I, FM, E, CA, CH>
- id: a unique id of FMEA set
- TC: a target component/system of the hazard analysis
- I: a set of item/function lists, {i_1, i_2, …}
- FM: a set of failure modes, {fm_1, fm_2, …}
- E: a set of effects/hazards, {h_1, h_2, …}
- CA: a set of causes, {c_1, c_2, …}
- CH: a set of chains for FMEA construction, I × FM × E × CA, {ch_1, ch_2, …}
  - CH = <i_n, fm_n, $e_{1..n}$, cA_n>

FMEA[7]는 분석에 사용되는 worksheet table에 따라 다양한 변형이 존재하지만 기본적으로는 대상 item, 고장 모드(failure mode), 영향(effect), 원인(cause)로 구성되며 경우에 따라 추가 위험(hazard), 추천 대응(recommend action) 등이 포함된다. 본 논문에서는 기본적인 항목들을 기준으로 추적성을 분석하기 위해 FMEA의 항목을 Definition 5로 정의하였다. 각 요소는 item/function, failure mode, effect, cause 및 chain을 각각 표현하는 I, FM, E, CA, CH로 구성되며 chain을 통해 표 1과 같은 FMEA worksheet table의 한 row를 연결해 표현할 수 있다.

Definition 6 (FTA). An FTA model, it is defined as a tuple of FTA = <id, M, TC, E, e_0, L>
- id: a unique id of the FTA set
- TC: a target component/system of the hazard analysis
- M: a set of minimal cut-sets from fault tree, {m_1, m_2, ...}
- E: a set of event nodes of fault trees, {e_1, e_2, ...}
- e_0: a top accident event node of fault trees
- L: a set of links in fault trees, <E_s, gate, E_t>, (gate ∈ {AND, OR, XOR})

Definition 7 (ETA). An ETA model, it is defined as a tuple of ETA = <id, TC, E_I, E_P, O, L>
- id: a unique id of the FTA set
- TC: a target component/system of the hazard analysis
- E_I: An initial event of ETA
- E_P: A set of pivotal events of ETA, {e_1, e_2, …}
  - E_p is defined as a tuple of <D, Boolean>, where D is description of event and Boolean is a condition of event occurs (T/F)
- O: A set of outcomes of ETA (outcome of ETA), {o_1, o_2, …}
- L: A set of links between initial event, pivotal events, and outcome, E_I × E_P × O, <eI, ep1, ep2, …, epn, on>, {l1, l2, …}

FTA와 ETA는 모두 기본적으로 tree 형태의 모델을 작성하며 분석이 진행되지만 FTA는 결과로부터 그 원인을 분석하고 ETA는 초기 사건으로부터 pivotal event로 인한 그 경과에 따른 과정의 결과를 분석한다[7]. 따라서 FTA 와 ETA의 모델링에는 tree 구조 및 각각의 event를 모델링하는 것이 필요하며 Definition 6과 Definition 7을 통해 각각 정의하였다. Definition 6은 FTA에 대한 설명으로 FTA는 top-event의 발생 관계에 대해 basic event들을 논리 게이트로 연결하여 분석하기 때문에 해당 event 및 연결 구조를 각각 E와 L로 정의하고 top-event를 e_0로 하였다. Definition 6에서 M은 top-event가 발생하기 위한 최소한의 이벤트의 조합을 나타내는 minimal-cut set으로 failure event들의 셋이라 할 수 있다. 또한 tree를 구성하는 연결인 L은 사건-논리 게이트-사건의 연속으로 구성되어 fault tree를 모델링할 수 있다. ETA는 초기 사건(initial event)로부터 이를 방지하기 위한 이벤트들(pivotal event)을 이진 트리로 연결하는 기법으로 이를 표현하기 위해 E_I, E_P 및 결과를 나타내는 O로 구성되며, 마찬가지로 트리 구조 연결을 표현하는 L을 구성 요소로 가지고 있다.

Definition 8 (STPA). An STPA model, it is defined as a tuple of STPA = <id, TC, L, CS, UCA, SC, LS>
- id: a unique id of STPA set
- TC: a target component/system of the hazard analysis
- L: a set of losses/accidents/hazards at system level
- CS: a control structure of TC, CS = <Con, A, S, CA, F, Lcs>
  - Con: a set of controller or controlled process in con trol structure, {Con_1, Con_2, …}
  - A: a set of actuators in control structure, {a1, a2, …}
  - S: a set of sensors in control structure, {s_1, s_2, …}
  - CA: a set of control actions in control structure, {ca_1, ca_2, …}
  - F: a set of feedback in control structure, {f_1, f_2, …}
  - Lcs:
    - a set of links of control structure, Con × CA × F × A × S × Con
    - <Con_s, CA, A, Con_t> // <Con_s, S, F, Con_t>
- UCA: a set of unsafe control actions that are defined by <Con, CS.CA_i, type, description>, {uca1, uca2, …}
- L_s: a set of loss scenarios,

마지막으로 STPA는 시스템을 control structure로

모델링하고 control structure에서 각 controller 간의 hazardous한 control인 unsafe control action을 통해 분석하는 기법으로 control structure와 unsafe control action 및 그 scenario를 모델링 하는 것이 필요하다. Definition 8은 이러한 STPA 결과를 모델링하기 위한 것으로 control structure (CS), unsafe control action (UCA), loss/accident (L), loss scenario (L_S)로 구성된다. 이를 통해 STPA의 결과에 대한 주요 요소들을 모델링해 추적성 분석을 수행할 수 있다. 각각의 위험 분석 모델의 요소들은 기본적인 구분을 위해 고유한 구분자를 갖는다.

이처럼 3.1절에서는 가상물리시스템의 개발 프로세스 및 안전생명주기의 산출물에 대한 모델을 정의하였다. DEM은 element라 하는 기본 요소 단위로 기능/디자인 등의 명세를 모델링하여 추적성 분석에 이용할 수 있도록 정의되었으며 SAEM 4 종류의 위험 분석 기법의 중요 항목들 및 안전생명주기의 안전 요구사항 도출과 관련된 항목을 모델링하도록 정의하였다. 개발 산출물 요소 모델과 달리 안전성 분석 요소 모델은 모든 항목이 추적성 연결의 대상이 될 수 있다. 다음 장에서는 추적성 분석 및 세부 관계에 대해서 정의하고 설명한다.

3.2 산출물 요소별 세부 관계에 따른 추적성 분석

본 논문에서는 추적성 분석을 개발 산출물 및 안전성/위험 분석 요소에 대해 표 2와 같이 3 분류로 나누어 수행한다. 표 2는 본 논문에서 제안하는 추적성 분석의 종류 및 세부 관계에 대한 표로 $E_s \rightarrow E_t$, $SAS_s \rightarrow E_t$ 그리고 $SAS_s \rightarrow SAS_t$ 로 표현된 connection은 각각 DEM의 element간의 추적성 분석, DEM-SAEM의 element-item 간의 추적성 분석 및 SAEM의 item 간의 추적성 분석을 의미한다. 추적성 분석 과정에서 각 연결별로 표 2에 정의된 세부 관계에 따라 연결되는 산출물의 특징을 확인하고 추적성 세부 관계를 부여한다. 그림 3은 본 논문의 추적성 분석에 대한 전체 구조 예시에 대한 그림으로 Definition 1~3에 정의된 DEM과 Definition 4~8

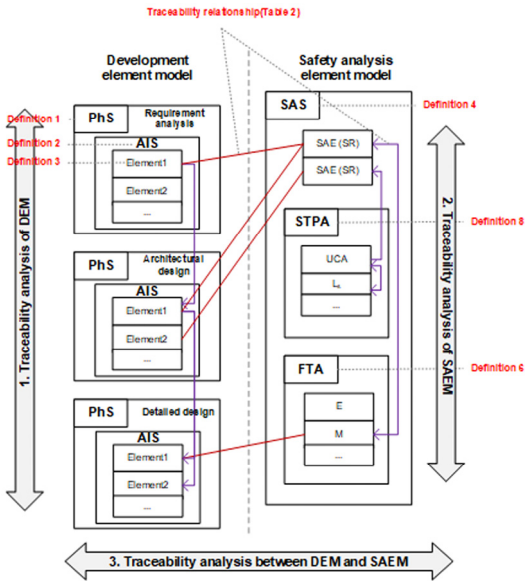

그림 3 추적성 분석 전체 구조

Fig. 3 Overall structure of traceability analysis

표 2 가상물리시스템의 산출물간 추적성 분석의 세부 관계

Table 2 Traceability relationships of development and safety analysis artifacts for CPS

Connection	Relationship	Description
$E_s \rightarrow E_t$	Development procedure	Relationship between development products according to development progress/hierarchy
	Multi-mode	Relationship of Items defined by dividing the specification for the case where one function operates in multiple modes
	Interaction point	Relationship of Items where the function operates by interactions of external components
	Multi-connection	Relationship when one specification is divided into multiple items
	Self-reference	Items of traceability between elements inside one PhS
$SAS_s \rightarrow E_t$	Safety requirements assignment	Item relationship between the safety requirement and the function of the requirement/design specification to which the safety requirement is assigned
	Failure/ effect/hazard point	Item relationship between the SAS item and the function of the requirement/design specification to which the failure/effect/hazard occurs
	Cause point	Item relationship between the SAS item and the function of the requirement/design specification which relates to causes of hazards
$SAS_s \rightarrow SAS_t$	Safety requirements	A connection between items of hazard/failure/effect and safety requirement to mitigate the impact
	Cause-consequence	Cause-consequence relationships between several SAS items that are not analyzed in one hazard analysis technique

을 통해 정의된 SAEM을 이용해 개발 산출물 및 안전성 분석 요소를 각각 모델링하고 3 분류로 추적성을 분석한다.

우선 개발 산출물 간의 추적성 분석은 DEM의 Definition 3에 해당하는 element 단위의 관계에 따라 진행된다. 기본적으로 개발 프로세스에 따라 요구사항이 디자인 및 구현 단계를 거쳐 모두 반영되었는지 확인하는 추적성 분석을 기본으로 하며 본 논문에서는 이를 development procedure 관계로 정의하였다. [17]에서는 가상물리시스템의 복잡도와 관련된 중요한 특징으로 heterogeneity, interoperability, connectivity, dynamic reconfiguration, software-intensiveness 등으로 정리하였다. 본 논문에서는 이에 대해 개발 산출물에서 해당 특징들이 나타나는 세부 관계로써 multi-mode 관계, interaction point 관계, multi-connection 관계를 추가로 정의하였다.

첫 번째로 추적성 분석 과정에서 각 element들 중 외부 혹은 다른 가상물리시스템과의 interaction을 기반으로 동작하는 element는 interaction point 관계를, 다음으로 여러 모드를 기반으로 동적 변경과 관련된 element는 multi-mode 관계로 추가 설정하여 이용한다. Multi-connection은 하나의 element가 개발이 진행되며 기능의 분화, 모드의 다양성 등으로 여러 element로 연결될 때 부여한다. 위와 같은 DEM의 추적성 분석에서 세부 관계 분석을 통해 가상물리시스템의 특징이 반영되는 요소들을 표현할 수 있고, SAEM과의 추적성 분석을 통해 가상물리시스템의 특징과 관련된 요소와 안전성 분석과의 관계를 통합적으로 확인하는데 이용된다.

SAEM 간의 추적성 분석은 각 위험 분석 기법의 모델을 구성하는 item 간의 연관성 분석으로 진행된다. 기본적으로는 도출된 hazardous한 event에 대해 도출한 safety requirement와의 연결성에 대한 추적성 분석이 가능하며 분석 대상에 따라 원인-결과의 연결 관계가 나타날 수 있다. 마지막으로 개발 산출물과 안전성 분석 요소 간의 추적성 분석은 안전 요구사항 할당에 대한 연결성과 failure/effect/hazard 및 cause point에 대한 추적성 분석이 가능하다. 안전요구사항 할당에 대한 연결성은 위험분석을 통해 도출한 안전 요구사항이 실제 할당되어 구현되는 요구사항/디자인의 명세와의 추적성이다[3]. Failure/effect/hazard 및 cause point는 위험분석에서 확인한 failure 또는 그 원인과 관련된 내용들이 발생하는/나타나는 명세의 지점을 연결하는 추적성이다. 개발 산출물과 안전성 분석 요소 간의 추적성을 분석할 때 multi-mode, multi-connection으로 정의된 산출물 과의 추적성에 대해 고려해야 한다.

본 논문에서는 추가로 추적성 분석의 결과를 통합하여 확인하기 위해 Definition 9 traceability graph를 정의하였다. Traceability graph는 일반적인 graph와 같이 노드(N)와 연결(L)로 구성된다. 각 노드는 개발 산출물 모델의 element와 안전성/위험 분석 모델에서의 한 item을 나타내며 연결은 추적성 분석을 통해 연결한 관계에 따라 구성되며 rel_type은 표 2에서 정의한 추적성 세부 관계를 연결 타입으로 부여한다.

Definition 9 (Traceability Graph). A traceability graph is defined as a tuple of TG = <N, L>, where
- N: a set of nodes, $\{n_1, n_2, \cdots\}$
  N can be defined as tuples of elements and unique id. That is nm = <id, E_t>, (Provided that $\forall\ E_t \in E_m$ or SAS_m.item)
- L: a set of links(edges) of nodes that are L = $\{l_1, l_2, \cdots\}$
  L can be defined as tuples of nodes with traceability relationship types. That is L_m = <n_s, n_t, rel_type>, (provided that $n_s.E_t.id \neq n_t.E_t.id$)
  rel_type: a type of traceability relationships of nodes. That is a meaning of links (edges).

이처럼 본 논문에서는 가상물리시스템의 개발 산출물과 4 종류의 기법을 사용한 안전성/위험 분석의 요소를 모델링하고 추적성을 분석하는 방법을 제안하였다. 이를 통해 가상물리시스템의 개발 산출물 및 안전성 분석 요소들 간의 관계를 분석하고 통합적으로 표현하여 확인할 수 있다. 본 논문에서는 다음 장에서 군집주행 시스템(platoon system)을 대상으로 한 사례 연구를 소개한다.

4. 사례 연구

본 논문에서는 군집주행 시스템을 대상으로 하여 사례 연구를 수행하고 본 논문에서 제안하는 추상화 모델 및 추적성 분석을 활용해 가상물리시스템의 추적성 분석을 효과적으로 수행할 수 있음을 보였다. 군집주행 시스템은 여러대의 차량이 하나의 군집을 이루어 주행함으로써 운전의 피로도를 줄이고 도로를 효율적으로 사용하여 주행할 수 있도록 하는 시스템으로 가상물리시스템의 일종이다. 또한 자동차 분야에 사용되는 시스템으로 안전성이 중요하다고 할 수 있다.

4.1 군집주행 시스템 구조

그림 4는 군집주행 시스템의 시스템 구조이다. 군집주행 시스템은 군집 컨트롤러(platoon controller)와 통신 모듈(communication processor) 및 센서 모듈(sensor processor)로 구성되며 시스템 외부의 요소로 군집 주행을 구성하기 위한 외부 센서, 네트워크, 및 차량이 존재 한다. 또한 군집주행 시스템은 여러대의 차량이 군집을 이루어 주행하기 때문에 같은 platoon system을 갖는 여러대의 차량이 통신을 하며 주행하며 주행을 컨트롤하는 leader와 군집에 속해 주행하는 follower 모드의

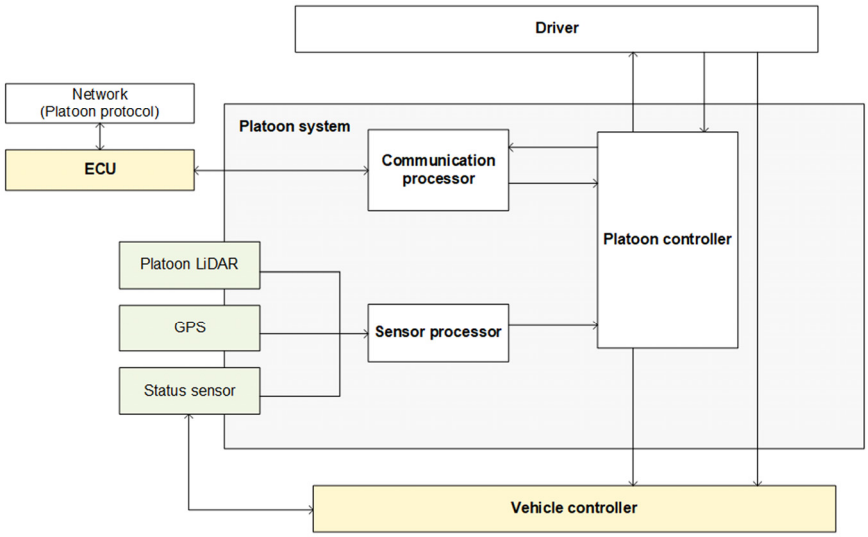

그림 4 군집주행 시스템 구조

Fig. 4 System-level architecture for platooning system

두 가지 모드를 가지고 있다. 군집 주행의 주요 기능으로는 군집 구성, 분할, 합류, 이탈, 긴급 속도 변경, 차선 변경 등이 있다.

4.2 군집주행 시스템의 산출물 모델 및 추적성 분석 결과

본 논문에서는 시스템 수준의 계획 및 기능 확인에 해당하는 preliminary functional requirement 와 platoon controller 의 소프트웨어 요구사항 명세서, 구조 및 상세 디자인으로 구성된 산출물을 DEM으로 모델링하였다. 안전성 분석은 STPA, FMEA 및 FTA를 일부 수행하여 각각 위험 분석 모델 및 SAEM으로 모델링 하였다. 그림 5는 platoon controller의 요구사항명세서의 모델에 대한 예시로 Definition 1~3의 모델을 활용해 작성되며, 요구사항명세서를 나타내는 phase structure 부터 목차 구조에 대한 artifact item structure 그리고 명세내용을 모델링한 element의 계층구조로 구성된다. 요구사항 명세 이외이도 상위 레벨로 시스템 계획 및 사전 기능 분석서(preliminary functional requirements), 디자인 명세서를 같은 형태로 모델링하였다.

STPA 수행 결과에 대한 모델은 그림 1 및 6과 같다. 그림 1은 본 논문에서 수행한 STPA의 control structure에 대한 그림이며 그림 6은 결과 모델의 일부이다. 군집주행 시스템은 현재 모드에 따라 controller 또는 controlled process로 상태와 역할이 달라질 수 있으며 그림 1은 리더모드의 경우에 대한 control structure이다. Control action으로는 join, leave, 가속, 감속 등이 있다. 본 논문에서는 각 control action에 대해 unsafe한 상황을 분석하고 SAEM을 이용해 모델링 한 결과의 일부는 그림 6

```
<PhS id="PhS_2" Pt = "SRS">
  <D>SRS for platoon controller of platooning system</D>
  <AI id="SRS_1">
    <name>Introduction</name>
    <D>Introduction of the SRS</D>
    ...
  </AI>
  <AI id="SRS_2">
    <Sb>Specific requirements</Sb>
    <D>Software requirements specification</D>
    ...
    <AI id="SRS_3_3">
      <Sb>3.3 Functional requirements</Sb>
      <D>Specific functional requirements for platoonig
system software</D>
      <E id = "SRS_3_3_E_6">
        <type>text</type>
        <F>3.3.6 Platoon join</F>
        <D>주변의 Platoon leader에 join 요청 msg를 전송하
여 platoon 합류 여부를 승인 후에 지정된 공간으로 합류한다. 합류
후 platoon 정보를 업데이트 한다.</D>
      </E>
    </AI>
  </AI>
</PhS>
```

그림 5 Platoon controller의 개발 산출물 모델 예제

Fig. 5 Example of development element model of platoon controller

과 같다. FMEA 및 FTA의 결과도 같은 형태로 모델링되고 SAS를 이용해 안전 요구사항도 마찬가지로 작성된다.

추적성 분석은 모델링한 결과를 이용하여 DEM의 element 및 SAEM의 item 간의 추적성을 분석하였으며, 분석 과정에서 표 2에 나타난 세부관계를 부여하였다. 그림 7은 traceability graph로 표현한 추적성 분석 결과의 일부에 대한 그림으로 platoon system 및 platoon controller의 개발 산출물 모델과 FMEA, STPA, FTA

```
<STPA id="STPA_1", TC = "Platoon system">
  <L>Accident: 차량 추돌/충돌 사고로 인명 및 재산 피해</L>
  <L>Hazard: 전/후방 차량과의 안전거리 위반</L>
  <CS>
    <Con id="con1">Platoon controller</Con>
    <Con id="con2">Platoon controller (f) </Con>
    <A id="a1">Communiation processor</A>
    <CA id="CA1">Join start</CA>
    <CA id="CA2">Split end l</CA>
    <Lcs id="Lcs1">con1, CA1, a1, con2</Lcs>
    <Lcs id="Lcs2">con1, CA2, a1, con2</Lcs>
  </CS>
  <UCA id="uca1">con1, CA1, provided, 리더모드의 platoon
controller가 join 차량과 platoon 차량의 전/후방 거리가 안전거리
이하일때, join start command를 제공한다.</UCA>
  <UCA id="uca1">con1, CA2, not provided, 리더모드의 platoon
controller가  leave 진행 중 split end l command를 차량이 떠난 후
에도 제공하지 않는다.</UCA>
</STPA>
```

그림 6 Platoon system의 STPA 모델의 예제
Fig. 6 Example of STPA model for platoon system

를 통한 안전성/위험 분석의 결과를 종합하여 그 관계를 확인할 수 있다. 그림 7과 같이 개발 산출물의 연결, 안전 요구사항 할당, failure/effect/hazard point 등의 관계를 쉽게 파악하고 분석할 수 있다. 특히 안전성/위험 분석 요소와 관계된 요구사항, 디자인, 기능 명세 등의 관계를 한 번에 파악할 수 있고 지점을 연결할 수 있다.

4.3 추적성 분석 결과의 활용에 대한 고찰

이처럼 본 논문에서는 군집주행 시스템의 개발 산출물 및 안전성/위험 분석 요소를 활용한 모델링과 추적성 분석을 수행하였다. 추적성 분석을 통해 몇몇 추가 분석을 수행하고 확인할 수 있는 부분들이 존재한다. 예를 들어 그림 7에서 빨간 선으로 표시된 FMEA의 결과와 디자인 산출물 사이에 multi-mode 부분에 대한 분석이 고려되지 않은 지점을 확인할 수 있다. 군집주행 시스템의 join 관련 기능은 platoon controller의 모드에 따라 서로 다른 동작을 수행하고, follower의 수에 따라 동적상황 변경이 발생하는 기능이지만 FMEA 분석에서는 추적성 분석 결과 해당 내용들에 대한 연결이 이루어지지 않음을 알 수 있다. 표 3은 추적성 분석 및 모델을 활용해 확인할 수 있는 추가적인 분석 및 지점에 대한 예시이다. 안전/위험 분석 요소와 관련해 interaction point, multi-mode와 관련된 위험 분석 여부를 연결해 확인하거나 분석이 부족한 지점, 시작되는 지점등을 통합해 확인할 수 있다.

특히 가상물리시스템의 경우 안전성 분석과 관련해 interaction analysis를 통해 이종의 가상물리시스템이 연결되어 동작하는 요구사항을 확인하고 안전성 분석 요소를 연결해 failure/fault의 전이 혹은 전파를 확인하거나 multi-mode로 동작하는 지점에 따라 동적 상황 변경(dynamic configuration) 지점 등에 대한 분석 여부를 생각해 볼 수 있다. 예를 들어 본 논문의 사례연구에서 사용한 platoon system의 경우에는 platoon을 구성하는 follower의 숫자에 따라 platoon의 이탈, 합류, 분할과 같은 기능의 동작 및 상호작용의 발생이 동적으로 변화할 수 있으며 [9]에서 소개한 것과 같이 STPA

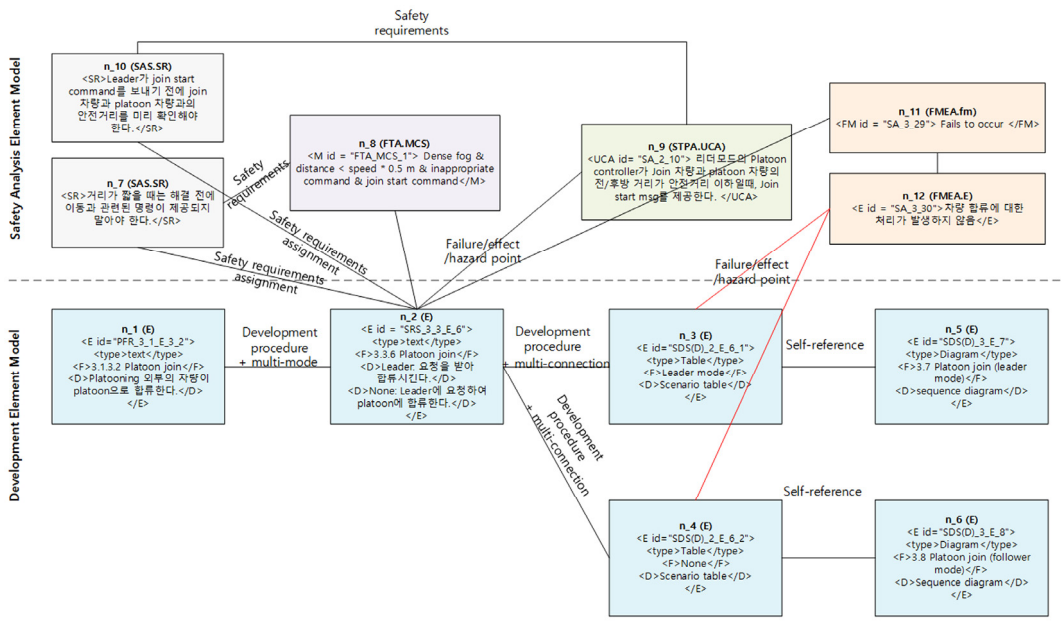

그림 7 Traceability graph로 표현된 추적성 분석 결과 예제
Fig. 7 Partial graph model of traceability analysis results

표 3 추적성 분석을 통한 추가 분석에 대한 예시
Table 3 Example of additional application to traceability analysis results

Classification	Analysis Point	Description
Safety hazard analysis	Common failure point	Identifying points where multiple failures converge in the specification
	Analysis Omission	Identifying omission points of hazard analysis
	Multi-mode analysis	Identifying analysis points of considering multi-mode
	Interaction analysis	Identifying analysis points of considering components interaction
Development process	Omission	Identifying omission of specifications
	Creation	Items with additional specification for defining one function/element

분석에서 control structure를 여러 모양으로 작성할 수도 있다. 이런 점에서 제안하는 추적성 분석을 통해 interaction point 세부 관계를 갖는 명세와 연결된 안전성 분석 요소를 확인하여 해당 특징들이 고려되는지 또는 안전성 분석이 어떤 갈래로 수행되었는지 등을 확인하는데 도움이 될 수 있을 것으로 생각된다.

이처럼 본 논문에서 제안하는 모델 및 방법을 통해 추적성 분석을 필요한 노드에 맞게 수행할 수 있으며 그 결과를 효과적으로 확인하고 분석할 수 있어 안전성이 중요한 가상물리시스템의 개발 및 안전성 분석을 지원할 수 있다. 특히 추적성 분석에서 각 요소별로 가상물리시스템에서 확인이 필요한 특징에 따라 정의된 세부관계를 부여함으로써 개발 산출물과 안전/위험성 분석 요소의 통합적인 관계 분석에 도움이 될 수 있다.

5. 결론 및 향후 연구

본 논문에서는 가상물리시스템의 개발 및 안전성/위험 분석산출물의 통합적인 관계분석을 위한 추적성 분석 및 세부 관계를 정의하였다. 이를 위해 개발 산출물 및 안전성/위험 분석 요소를 각각 DEM과 SAEM으로 정의된 모델을 제안하고, 산출물 모델의 각 요소별 가상물리시스템의 특징 및 연관 관계에 따른 추적성 세부 관계를 정의하였다. 논문에서 제안하는 추적성 분석을 통해 가상물리시스템의 개발 산출물 및 안전성/위험성 분석 요소들의 여러 관계를 확인하고 수립할 수 있다. 특히 가상물리시스템의 특징적인 요소에 대한 추적성 분석 결과를 이용해 안전성 분석 및 개발에 도움이 되는 내용을 쉽게 확인할 수 있다. 또한 군집주행 시스템의 개발 산출물 및 안전성/위험 분석 결과를 대상으로 본 논문에서 제안하는 모델과 추적성 분석을 통해 효과적으로 추적성을 분석하고 그 관계를 표현할 수 있음을 확인하였다.

가상물리시스템은 여러 이종의 시스템 사이의 상호연결성이 중요한 시스템으로 서로 다른 시스템의 개발 산출물 사이의 추적성 분석 또한 중요한 역할을 할 수 있다. 향후 이 점에 대해 여러 가상물리시스템으로 구성된 SoCPSs (System of cyber-physical systems) 수준의 추적성 분석에 대해 연구할 계획이다. 또한 추가로 동적 불확실성을 갖는 가상물리시스템의 위험 분석에 도움이 되는 요소들을 추적성 분석 결과로부터 도출하여 안전성/위험 분석을 수행할 수 있는 방법에 대해서도 연구할 계획을 가지고 있다.

References

[1] I. Sommerville, *Software Engineering 10th*, Addison Wesely, Boston, 2007.

[2] Functional safety of electrical, electronic and programmable electronic (E/E/PE) safety-related systems (IEC 61508) Tech. Rep., International Electrotechnical Commission (IEC), 2010.

[3] ISO 26262, road vehicles - functional safety Tech. Rep., International Organization for Standardization (ISO), 2011.

[4] J.-S. Lee, V. Katta, E.-K. Jee, and C. Raspoting, "Means-ends and whole-part traceability analysis of safety requirements," *Journal of Systems and Software*, Vol. 83, No. 9, pp. 1612-1621, 2010.

[5] J. Kim, D.-A. Lee, J. Yoo, "A detailed relationships of traceability analysis for software development process," *2016 Korea Conference on Software Engineering (KCSE 2016)*, pp. 409-411. 2016. (in Korean).

[6] ISO/IEC/IEEE 24765, "Systems and software engineering - Vocabulary," pp. 1-418, 2010.

[7] C.A. Ericson, *Hazard analysis techniques for systemsafety*, John wiley & Sons, 2015.

[8] E. Griffor, C. Greer, D.A. Wollman, M. J. Burns, Framework for Cyber-Physical Systems: Volume 1 (NIST SP 1500-201), National Institute of Standards and Technology U.S. Department of Commerce Pubs. 2017

[9] E.-S. Kim, J. Yoo, "A Study on Application of STPA in Safety Analysis of Platoon System," *2020 Korea Conference on Software Engineering*, pp. 193-196. 2020. (in Korean)

[10] M. Amoozadeh, H. Deng, C.-N. Chuah, H. M. Zhang, D. Ghosal, "Platoon management with cooperative adaptive cruise control enabled by VANET," *Vehicular communication*, Vol. 2, No. 2, pp. 110-123, 2015.

[11] N. G. Leveson, *Engineering a safer world*, MIT

Press, 2009.
[12] V. Katta, T. Stalhane, "A conceptual model of traceability for safety systems," *Proc. of the complex systems design & management conference*, 2011.
[13] V. Katta, T. Stalhane, C. Raspotnig, "Presenting a traceability based approach for safety argumentation," *Proc. of ESREL 2013*, pp. 2037-2046, 2013.
[14] P. G. Larsen et al., "Integrated tool chain for model-based design of Cyber-Physical Systems: The INTO-CPS project," *2016 2nd International Workshop on Modelling Analysis, and Control of Complex CPS (CPS Data)*, pp. 1-6, 2016.
[15] H. Tufail, M. F. Masood, et al., "A Systematic Review of Requirement Traceability Techniques and Tools," *2017 2nd international conference on system reliability and safety (ICSRS)*, pp. 450-454, 2017.
[16] IEEE 830, IEEE Recommended Practice for Software Requirements Specifications, 1998.
[17] Victor Bolbot et al., "Vulnerabilities and safety assurance methods in Cyber-Physical Systems: A comprehensive review," *Reliability Engineering and System Safety*, Vol. 182, pp. 179-193. 2019.

정 세 진

2015년 건국대학교 컴퓨터공학과 졸업(학사). 2016년 건국대학교 컴퓨터 정보통신 공학과 졸업(석사). 2016년~현재 건국대학교 컴퓨터 정보통신공학과 박사과정. 관심분야는 소프트웨어 공학, 위험/안전성 분석

김 의 섭

2012년 건국대학교 컴퓨터공학과 졸업(학사). 2015년 건국대학교 컴퓨터 정보통신 공학과 졸업(석사). 2015년~현재 건국대학교 컴퓨터 정보통신공학과 박사과정. 관심분야는 소프트웨어 공학, 정형검증

유 준 범

2005년 KAIST 전자전산학과 전산학 전공 졸업(박사). 2008년 삼성전자주식회사 통신연구소 책임연구원. 2008년~현재 건국대학교 컴퓨터공학부 교수. 관심분야는 소프트웨어 공학, 안전성 분석, 정형 기법