

재난 대응 System-of-Systems 시뮬레이션을 위한 멀티소스 로그 통합 및 이벤트 기반 시각화 분석

(Multi-Source Log Integration and Event-Based Visualization Analysis for Disaster Response System-of-Systems Simulation)

송지영*

(Jiyoung Song)

요약

재난 대응 시스템은 소방, 구조, 의료 및 지휘 체계와 같이 독립적으로 운영되는 여러 구성 시스템이 공동 임무를 수행하는 대표적인 System-of-Systems(SoS)이다. 이러한 시스템에서는 화재 발생, 화재 심화 및 화재 진압과 같은 단계적 상황 변화에 따라 agent 배치, 환자 구조 및 치료 완료 상태, 공간 탐색 상태가 동시에 변화하므로 단일 결과 지표만으로 전체 동작을 충분히 분석하기 어렵다. 본 연구는 기존 SoS-simulation-engine을 확장하여 event log, agent state log, patient state log 및 spatial exploration log를 공통 시간축으로 통합하고, 재난 이벤트 단계에 따른 상태 변화를 시각화하여 분석하는 구조를 제안한다. 사례 연구에서는 각 재난 단계별 자원 배치, 구조 환자 수, 치료 완료 환자 수 및 미탐색 tile 수의 시간적 변화를 시각화하였고, 동일 시나리오를 서로 다른 난수 seed로 30회 반복 실행하여 평균, 표준편차 및 95% 신뢰구간을 제시하였다. 반복 실행 결과 최종 구조율은 평균 97.00%였고, 최종 치료 완료율은 평균 24.00%, 최종 미탐색 tile 수는 평균 28.97개로 나타났다. 본 연구는 새로운 자원 배치 정책이나 재구성 알고리즘의 성능을 비교하는 것이 아니라, 재난 대응 SoS의 이질적인 runtime state를 이벤트 단계 중심으로 통합·시각화하고 정량적으로 분석할 수 있는 기반을 제공한다.

■ 중심어 : 시스템 오브 시스템즈 ; 재난 대응 시뮬레이션 ; 멀티소스 로그 통합 ; 이벤트 기반 시각화 ; 스마트 인포메이션

Abstract

Disaster response systems are representative System-of-Systems (SoS), in which independently operated constituent systems such as firefighting, rescue, medical transport, and command units collaborate to accomplish a common mission. Staged changes such as fire outbreak, fire escalation, and fire suppression simultaneously affect agent deployment, patient rescue and treatment-completion states, and spatial exploration progress, making it difficult to understand overall system behavior using a single outcome metric. This study extends an existing SoS simulation engine by integrating event, agent-state, patient-state, and spatial-exploration logs along a common time axis and visualizing state changes according to disaster event phases. In a case study, a representative run was visualized by phase, and the same scenario was repeated 30 times using different random seeds to report means, standard deviations, and 95% confidence intervals. Across the repeated runs, the mean final rescue rate was 97.00%, the mean treatment-completion rate was 24.00%, and the mean number of final unvisited tiles was 28.97. The study does not compare resource-allocation policies or reconfiguration algorithms; rather, it provides a basis for integrating heterogeneous runtime states of SoS and analyzing them as event-aware smart information.

■ keywords : System-of-Systems ; disaster response simulation ; multi-source log integration ; event-based visualization ; smart information

I. 서론

지능형 미디어 네트워크와 스마트 콘텐츠 기술은 다양한 데이터 소스를 통합하고 사용자가 이해하기

쉬운 형태로 정보를 제공하는 방향으로 발전하고 있다. 특히 재난 대응에서는 구조 인력 배치, 구급차 이송 상태, 환자의 구조 및 치료 진행, 미탐색 영역 등 다양한 데이터가 시간에 따라 동시에 변화하므로, 이를 공통 시간 축에서 통합하고 시각화하는 기

* 중신회원, 한남대학교 컴퓨터공학과
이 논문은 2025학년도 한남대학교 학술연구비 지원에 의하여 연구되었음

접수일자 : 2026년 06월 25일
수정일자 : 2026년 07월 15일

재제확정일 : 2026년 07월 16일
교신저자 : 송지영 e-mail : jysong@hnu.kr

술이 중요하다.

재난 대응 시스템은 대표적인 시스템 오브 시스템(System-of-Systems, 이하 SoS)이다. SoS는 독립적으로 운영 가능한 여러 구성 시스템(Constituent System, 이하 CS)이 자체적인 운영 능력과 관리적 독립성을 유지하면서 공동 임무를 수행하는 복합 시스템이다[1,2]. 화재 대응 SoS에서는 firefighter, ambulance, hospital, bridgehead 등이 개별 CS로 모델링될 수 있으며, 이들의 상호작용을 통해 환자 탐색, 구조, 이송 및 치료가 이루어진다. SoS는 지리적 분산, 진화적 개발 및 창발적 행동과 같은 특성을 가지므로 상태 변화와 상호작용의 분석이 어렵다[1,3,4].

SIMVA-SoS (Simulation-based Verification and Analysis for System-of-Systems)는 discrete-time simulation과 multi-agent simulation을 결합하고 실행 중 stimulus를 주입하여 SoS의 진화적 변화를 분석하는 시뮬레이션 및 통계적 검증 프레임워크이다[5]. 기존 연구는 speed, sight range, communication loss, CS 추가 및 제거와 같은 변화를 런타임에 반영하였으며[5,6], Dynamic-SoS 연구도 런타임 아키텍처 변화의 시뮬레이션 가능성을 제시하였다[7].

그러나 기존 SIMVA-SoS 연구는 주로 상호작용형 시뮬레이션과 통계적 검증에 초점을 두어, runtime log들을 공통 시간 축과 event phase 기준으로 통합하고 시각화하는 분석 기능은 충분히 다루지 않았다. 최근 재난·대피 시뮬레이션에서도 agent 행동, 공간 상태, 실제 데이터, 실시간 정보 및 디지털 트윈 환경을 함께 활용하려는 연구가 확대되고 있으며[8-14], 이는 이질적인 상태 정보를 시간적 맥락과 통합하여 분석할 필요성을 보여준다.

본 연구는 기존 SoS-simulation-engine을 확장하여 event log, agent state log, patient state log 및 spatial exploration log를 수집하고, 이를 frame과 event phase를 기준으로 통합·시각화하는 구조를 제안한다. 이를 통해 특정 상태 변화가 어느 재난 단계에서 발생했는지를 추적할 수 있다. 구체적으로 FIRE_OUTBREAK, FIRE_ESCALATION 및

FIRE_SUPPRESSION으로 구성된 단계적 재난 event phase를 적용하고, agent 수, 환자 구조 및 치료 완료 상태, 미탐색 영역을 frame 단위로 기록한다. 또한 각 시각화에 event phase의 경계와 구간을 표시하고, 동일 시나리오를 30회 반복 실행하여 평균, 표준편차 및 95% 신뢰구간을 제시한다. 본 연구는 새로운 자원 배치 정책이나 재구성 알고리즘의 성능 비교보다는, 재난 대응 SoS의 heterogeneous runtime state를 event-aware smart information으로 변환하는 분석 기반을 제공하는 데 초점을 둔다.

본 논문의 구성은 다음과 같다. II장에서는 관련 연구를 검토하고, III장에서는 제안 구조와 실험 설정 및 결과를 제시한다. IV장에서는 기존 연구 대비 차별성과 학술적 의미 및 한계를 논의하며, V장에서 결론과 향후 연구 방향을 제시한다.

II. 관련 연구

1. System-of-Systems 시뮬레이션

SoS는 여러 독립적인 CS가 공동 목표를 달성하기 위해 상호작용하는 복합 시스템이다. Nielsen 외 [1]는 SoS의 기본 개념, model-based technique 및 연구 방향을 정리하였고, Baldwin과 Sauter[2]는 SoS 특성의 모델링 문제를 다루었다. Zeigler와 Muzy[3]는 SoS에서 emergence를 이해하기 위한 modeling and simulation 관점을 제시하였으며, Mittal과 Risco-Martín[4]은 복잡 적응 시스템에 대한 시뮬레이션 기반 접근을 정리하였다. 이러한 연구는 SoS를 단순한 구성 요소의 집합이 아니라 시간에 따라 변화하고 상호작용 결과로 창발적 현상을 생성하는 시스템으로 다루어야 함을 보여준다.

기존 SIMVA-SoS는 evolutionary development를 표현하기 위해 discrete-time simulation과 multi-agent simulation 구조를 적용하였다[5]. 또한 재난 대응 SoS 사례 연구에서는 firefighter, ambulance, hospital 등의 CS가 재난 대응 과정에서 수행하는 행동을 simulation-based analysis로 분석하였다[6]. 본 연구는 이 기존 simulator와 재난 대응 domain

model을 기반으로 하지만, statistical verification이나 정책 비교 대신 runtime log의 통합과 event-aware visualization에 초점을 둔다.

2. Interactive Simulation과 Runtime State Observation

SIMVA-SoS의 주요 특징 중 하나는 interactive simulation이다. 사용자는 시뮬레이션 실행 중 interrupt를 통해 새로운 이벤트를 주입할 수 있고, simulation engine은 이후 tick에서 해당 변화를 반영한다[5]. Dynamic-SoS는 runtime에서 SoS 아키텍처가 변화하는 상황을 시뮬레이션하는 접근법을 제안하였다[7].

본 연구는 이러한 runtime 변화와 상태 관찰의 필요성을 계승하지만, 목적은 다르다. 본 연구에서는 FIRE_OUTBREAK, FIRE_ESCALATION 및 FIRE_SUPPRESSION을 event phase로 정의하고, 각 phase에서 생성되는 agent, patient 및 spatial state를 동일한 frame 축에 정렬한다. 따라서 특정 event의 발생 여부뿐 아니라, 그 event 이후 여러 runtime state가 어떤 형태로 변화했는지를 함께 분석할 수 있다.

3. 시뮬레이션 로그 기반 시각화 분석

Multi-agent simulation은 복잡한 사회적·공학적인 시스템에서 개별 agent의 행동과 상호작용이 전체 시스템 수준의 결과로 어떻게 연결되는지를 분석하는 데 활용된다. Drogoul 외[15]는 multi-agent simulation을 창발적 과정 분석 도구로 제시하였고, Robinson[16]은 시뮬레이션에서 실험 목적, 모델 구성, 검증 및 결과 분석의 중요성을 강조하였다.

최근 재난 및 대피 시뮬레이션 연구는 실제적 공간 정보와 agent-based model, 반복 실행 및 시각화를 결합하는 방향으로 발전하고 있다. Wahlqvist 외[8]는 WUI-NITY platform을 통해 wildland-urban interface 화재 대피를 simulation하는 환경을 제시하였고, Gwynne 외[9]는 실제 community wildfire evacuation drill 데이터를 이용하여 model benchmarking을 수행하였다. Wang 외[10]는 2024년 CrowdEgress multi-agent platform을 제안하여

비상 대피 상황의 agent interaction과 공간 구조를 모사하였다. Cassidy 외[11]는 first responder training을 위한 dynamic data-driven digital twin testbed를 제시하여 재난 대응 환경에서 시뮬레이션 상태와 통신 조건을 함께 분석할 수 있음을 보였다.

또한 Borgwardt 외[12]는 wildfire information을 통합한 evacuation planning을 연구하였고, Sun 외[13]는 대규모 GPS 데이터를 활용하여 wildfire evacuation decision의 공간적 차이를 분석하였다. Ramadan[14]은 cellular automata 기반 wildfire simulation과 autonomous response를 결합하였다. 이러한 최근 연구들은 재난 대응 시뮬레이션에서 시간, 공간, agent 및 외부 정보가 서로 다른 형태로 생성되며, 이를 통합하여 해석 가능한 형태로 제공하는 것이 중요함을 보여준다. 본 연구는 실제 센서·영상 데이터 통합까지 다루지는 않지만, SoS simulator 내부에서 생성되는 heterogeneous runtime logs를 event-aware smart information으로 재구성한다는 점에서 연구 흐름과 연결된다.

III. 본 론

1. 연구 설계 및 시스템 개요

본 연구는 기존 SoS-simulation-engine을 확장하여 재난 대응 SoS 시뮬레이션 멀티소스 로그를 수집하고, 이를 이벤트 단계별로 시각화하여 분석하는 구조를 제안한다. 제안하는 구조는 그림 1과 같다.

Simulation Engine은 frame 단위로 재난 대응 SoS 시뮬레이션을 수행한다. Scenario Event Logger는 현재 frame에서 발생한 재난 이벤트를 기록한다. Agent State Logger는 firefighter와 ambulance 수를 기록한다. Patient State Logger는 전체 환자 수, 구조된 환자 수, 병원에 도착해 생존한 환자 수를 기록한다. Spatial Exploration Logger는 미탐색 tile 수를 기록한다. Log Integrator는 각 로그를 frame과 eventName을 기준으로 통합하고, Visualization Analyzer는 통합 로그를 그래프로 변환한다. 본 연구의 로그 수집 항목은 표 1과 같다.

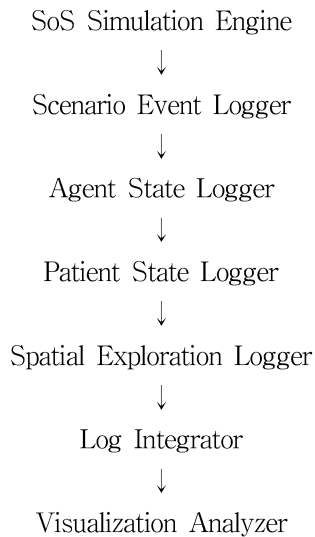


그림 1. 멀티소스 로그 통합 및 이벤트 기반 시각화 분석 구조

표 1. 시뮬레이션 로그 항목

구분	로그 항목	설명
Event Log	frame, eventName, opMode	이벤트 발생 시점과 운영 모드
Agent Log	firefighterCount, ambulanceCount	투입 agent 수
Patient Log	patientCount, rescuedPatientCount, treatmentPatientCount	전체 환자 수, 구조된 환자 수, 생존한 환자 수
Spatial Log	unvisitedTileCount	미탐색 tile 수

2. 단계적 화재 대응 이벤트 모델

본 연구에서는 재난 대응 SoS 시나리오를 FIRE_OUTBREAK, FIRE_ESCALATION 및 FIRE_SUPPRESSION의 세 단계로 구성하였다. 표 2에서 t1과 t3의 mode가 모두 C로 표시된 것은 의도된 설정이다. FIRE_OUTBREAK는 초기 화재 발생으로 Critical(C) 상태이며, FIRE_ESCALATION에서는 위험 수준 증가에 따라 Highly Critical(HC) 상태로 전환된다. FIRE_SUPPRESSION에서는 화재 위험이 완화되지만 재난 대응 임무가 계속 진행 중이므로 Low-critical 상태로 전환하지 않고 C 상태로 복귀하도록 설정하였다. 따라서 operational mode의 변화는 C→HC→C이다.

FIRE_OUTBREAK 단계에서는 firefighter 5명과 ambulance 두 대가 동작한다. FIRE_ESCALATION 단계에서는 심화 상황을 표현하기 위해 firefighter가 7명으로 증가하며 ambulance는

표 2. 단계적 화재 대응 이벤트 모델

Stage	Event	Mode	구간	설명
t1	FIRE OUTBREAK	C	frame 100-299	화재 발생 및 초기 대응 단계
t2	FIRE ESCALATION	HC	frame 300-599	화재 심화 및 자원 요구 증가 단계
t3	FIRE SUPPRESSION	C	frame 600-1,000	화재 진압 및 안정화 단계

2대를 유지한다. FIRE_SUPPRESSION 단계에서는 firefighter가 다시 5명으로 감소하고 ambulance는 1대로 감소한다. 이 자원 변화는 멀티소스 로그 통합 및 event-aware visualization의 동작을 관찰하기 위해 사전에 정의한 고정 시나리오이다.

3. 실험 환경, 초기 설정 및 행동 알고리즘
실험의 재현 가능성을 확보하기 위해 공간 구성, 초기 agent 수와 위치, 환자 배치, 이벤트 시점, 이동·탐색·구조·이송 규칙 및 반복 실험 절차를 명시하였다. 시뮬레이션 공간은 35×35 grid로 구성되어 총 1,225개의 tile을 포함한다. Firefighter와 ambulance는 네 모서리 (0,0), (34,0), (34,34), (0,34)에 순환 배치된다. Hospital 역시 네 모서리에 위치하며, bridgehead는 map의 1/8 및 7/8 지점에 배치된다. Patient 20명은 bridgehead 영역을 제외한 내부 영역에 난수 기반으로 배치된다.

Firefighter는 기본 sight range 5를 사용하여 주변 환자를 관찰하고, 관찰한 tile을 visited 상태로 표시한다. 기억된 환자가 존재하는 경우 환자를 선택하여 이동하며, 중증 환자가 있으면 이를 우선한다. 환자가 없으면 shuffled queue에서 아직 방문하지 않은 tile을 선택하여 탐색을 계속한다. Grid 이동은 Manhattan 방식이며 x축과 y축 모두 이동이 가능한 경우 두 방향 중 하나를 난수로 선택한다. Firefighter의 기본 move delay는 2 frame이다.

Ambulance는 가장 가까운 bridgehead를 순차적으로 방문한다. Bridgehead에 환자가 존재하면 serious 환자를 wounded 환자보다 우선 선택하고, 선택된 환자를 가장 가까운 hospital로 이송한다.

Hospital의 capacity는 10명이며, patient가 hospital에 도착하여 treatment counter를 완료하면 코드의 treatmentPatientCount가 증가한다.

로그는 10 frame 간격으로 기록하며 시뮬레이션은 frame 1000에서 종료한다. Grid의 이론적 초기 unvisited tile 수는 1,225개이지만, firefighter가 초기 update 과정에서 주변 tile을 visited로 표시한 후 첫 로그가 저장되므로 첫 기록값은 1,225보다 작을 수 있다. 대표 실행의 첫 기록(frame 10)에서 unvisited tile 수는 1,147개였다. 동일 시나리오에는 서로 다른 random seed를 사용하여 30회 반복 실행하였고, 각 실행에서 final rescued patients, final rescue rate, final treatment-completed patients, treatment completion rate 및 final unvisited tiles를 수집하였다.

표 3. 실험 환경 및 주요 파라미터

범주	파라미터	설정값
Space	Grid size / total tiles	35×35 / 1,225 tiles
Agent	Initial firefighters	5
Agent	Initial ambulances	2
Patient	Initial patients	20
Hospital	Number / capacity per hospital	4 / 10
Firefighter	Default sight range	5
Firefighter	Move delay	2 frames
Event	FIRE_OUTBREAK	frame 100
Event	FIRE_ESCALATION	frame 300
Event	FIRE_SUPPRESSION	frame 600
Time	Simulation termination	frame 1000
Logging	Logging interval	10 frames
Repetition	Number of repeated runs	30

4. 이벤트 기반 시각화 실험 결과

그림 2부터 그림 4는 동일한 대표 실행에서 생성된 agent, patient 및 spatial log를 공통 frame 축에 표시한 결과이다. 각 그림에는 FIRE_OUTBREAK가 시작되는 frame 100, FIRE_ESCALATION이 시작되는 frame 300, FIRE_SUPPRESSION이 시작되는 frame 600을 세로 점선으로 표시하고, 각 event phase 구간을 배경 음영으로 구분하였다.

가. Agent 배치 변화

그림 2는 frame에 따른 firefighter와 ambulance와 같은 agent의 수 변화를 보여준다. FIRE_

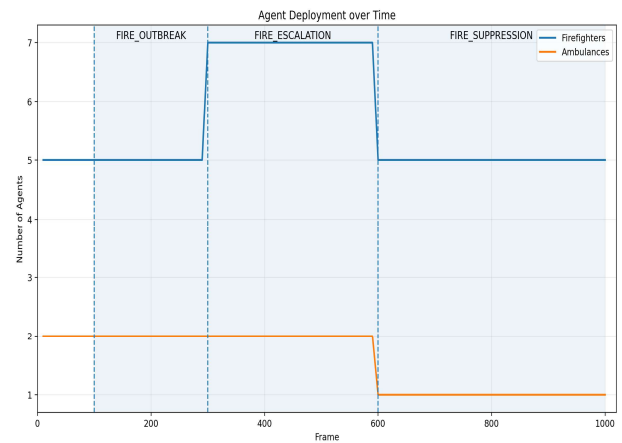


그림 2. Agent 배치 변화

OUTBREAK 구간에서는 firefighter 5명, ambulance 2대가 유지되고, FIRE_ESCALATION 단계에서는 firefighter 수가 7명으로 증가한다. FIRE_SUPPRESSION 단계에서는 firefighter가 5명으로 감소하고 ambulance도 한대로 감소한다. 이 그림은 agent deployment 변화와 event phase 경계를 동일 축에서 직접 연결한다.

나. 환자 구조 및 생존 상황 진행

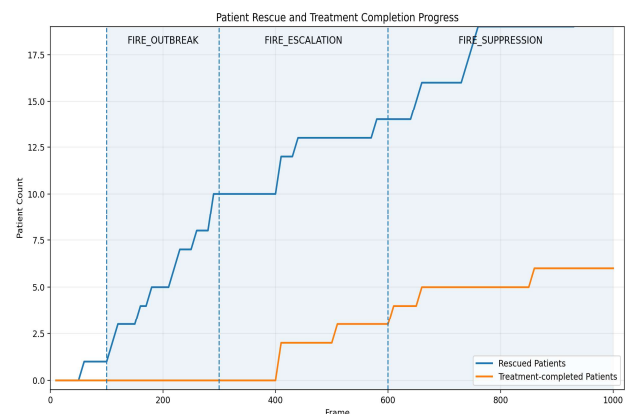


그림 3 환자 구조 및 치료 완료 진행

그림 3은 rescued patient와 treatment-completed patient의 30회 반복 실험의 평균 누적 변화를 보여준다. FIRE_OUTBREAK 구간의 최대 구조 환자는 9명, FIRE_ESCALATION 구간은 15명, FIRE_SUPPRESSION 구간은 18명이었다. 치료 완료 환자는 각 구간에서 최대 0명, 2명, 3명으로 증가하였다. 구조 환자 수와 치료 완료 환자 수 사이의 시간적 차이는 환자가 발견된 후 구조 거점 이동, ambulance 이송 및

hospital treatment라는 후속 단계를 거쳐야 하기 때문이다. 이 결과는 환자 상태의 진행을 하나의 최종값이 아니라 event phase별 누적 변화로 분석할 수 있음을 보여준다.

다. 공간 탐색 진행

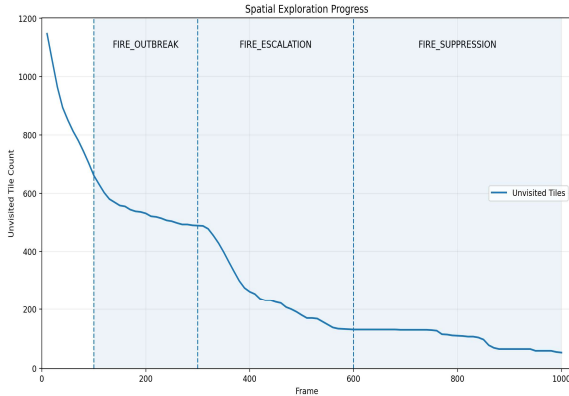


그림 4 공간 탐색 진행

그림 4는 unvisited tile 수의 변화를 보여준다. 대표 실행에서 FIRE_OUTBREAK 구간의 최소 미탐색 tile 수는 385개, FIRE_ESCALATION 구간은 185개, FIRE_SUPPRESSION 구간은 1개였다. Firefighter의 탐색이 진행되면서 미탐색 영역이 지속적으로 감소하며, event phase 경계와 함께 표현함으로써 공간 탐색 진행을 단계별로 추적할 수 있다.

5. 이벤트 단계별 멀티소스 로그 요약

표 4. 이벤트 단계별 멀티소스 로그 요약

범주	분석지표	FIRE_OUTBREAK	FIRE_ESCALATION	FIRE_SUPPRESSION
Resource	Avg. Firefighters	5	7	5
	Avg. Ambulances	2	2	1
Patient Outcome	Max Rescued Patients	9	15	18
	Max Rescue Rate	45%	75%	90%
	Max Treatment-completed Patients	0	2	3
	Max Treatment Completion Rate	0%	10%	15%
Spatial Progress	Min Unvisited Tiles	385	185	1

표 4는 대표 실행의 멀티소스 로그를 Resource, Patient Outcome 및 Spatial Progress

의 세 범주로 구성한 결과이다. 이 표는 각 이벤트 단계에서 평균 agent 수, 최대 구조 환자 수, 최대 치료 환자 수, 최소 미탐색 tile 수, 구조율 및 생존율을 요약하며, 하나의 단계적 화재 대응 시나리오에서 서로 다른 로그 source가 event phase 기준으로 어떻게 통합되는지를 보여준다.

6. 동일 시나리오 30회 반복 실험

표 5. 동일 시나리오 30회 반복실험 결과

분석지표	평균	표준편차	95% 신뢰구간
Final Rescued Patients	19.40	0.81	[19.11, 19.69]
Final Rescue Rate	97.00%	4.07%	[95.54%, 98.46%]
Final Treatment-completed Patients	4.80	1.00	[4.44, 5.16]
Treatment Completion Rate	24.00%	4.98%	[22.22%, 25.78%]
Final Unvisited Tiles	28.97	21.72	[21.19, 36.74]

Agent-based simulation의 비결정성을 고려하고 단일 실행 결과에 대한 의존성을 줄이기 위해 동일한 단계적 화재 대응 시나리오를 서로 다른 random seed로 30회 반복 실행하였다. 각 실행에서 최종 구조 환자 수, 구조율, 치료 완료 환자 수, 치료 완료율 및 최종 미탐색 tile 수를 수집하고 평균, 표준편차 및 95% 신뢰구간을 산출하였다. 본 반복실험의 목적은 다른 정책과의 성능 비교가 아니라, 동일 시나리오에서 제안한 로그 수집 및 분석 구조가 반복적으로 정량 지표를 산출할 수 있음을 확인하는 것이다. 30회 반복 결과, 평균 최종 구조 환자 수는 19.40명, 평균 구조율은 97.00%였으며 구조율의 95% 신뢰구간은 95.54%에서 98.46%였다. 평균 치료 완료 환자 수는 4.80명, 평균 치료 완료율은 24.00%였고, 평균 최종 미탐색 tile 수는 28.97개였다.

IV. 논 의

1. 기존 SIMVA-SoS 대비 차별점

기존 SIMVA-SoS는 runtime stimulus injection과 statistical verification을 중심으로 SoS의 evolutionary development를 표현하였다. 반면 본 연구는 statistical verification을 다루지 않고, 시뮬레이션

로그의 수집, 통합, 시각화 분석에 초점을 둔다.

기존 연구에서는 speed, sight range, communication, add CS, remove CS와 같은 stimulus를 개별적으로 적용하고, 그 결과를 분석하였다. 본 연구는 화재 발생, 화재 심화, 화재 진압이라는 단계적 이벤트를 기준으로 agent 수, 환자 상태, 공간 탐색 상태를 함께 분석하였다. 이 점에서 본 연구는 기존 simulator의 표현 결과를 사용자 친화적인 시각 분석 콘텐츠로 확장한 연구라고 볼 수 있다.

2. 학술적 의미

본 연구의 첫 번째 학술적 의미는 heterogeneous simulation logs를 단순 실행 기록으로 저장하는 데 그치지 않고 event-aware smart information으로 재구성했다는 점이다. 서로 다른 단위와 의미를 갖는 agent, patient 및 spatial state를 공통 frame과 event phase 기준으로 정렬함으로써 시간적 연관성을 분석할 수 있도록 하였다.

두 번째로, 재난 단계의 경계와 지속 구간을 그래프에 직접 표시함으로써 특정 상태 변화가 어느 재난 단계에서 발생했는지를 직관적으로 추적할 수 있다. 이는 단순한 line chart 추가가 아니라, 이벤트 의미와 runtime state를 같은 시각적 맥락에 배치하여 분석이 가능한 information structure로 변환한다는 의미를 가진다.

따라서 본 연구의 기여는 새로운 자원 배치 정책이나 재구성 알고리즘을 제안하는 데 있지 않다. 본 연구는 재난 대응 SoS의 다양한 runtime state를 event phase 중심으로 통합·시각화하고, 동일 시나리오 반복 실험을 통해 정량적으로 요약할 수 있는 분석 기반을 제공한다.

3. 한계 및 향후 개선 방향

첫째, 현재 시각화는 정적 그래프 형태이며 사용자가 event phase를 선택하여 여러 로그 view를 연동 탐색하는 interactive dashboard는 구현하지 않았다. 향후에는 특정 phase를 선택하면 resource, patient 및 spatial state가 함께 갱신되는 interactive interface로 확장할 수 있다.

둘째, CS간의 통신을 기록하는 communication log는 현재 MsgRouter의 message exchange와 직접 연결되어 있지 않아 본 연구의 분석 대상에서 제외하였다. 향후 communication count, delay 및 loss를 실제 routing event와 연동하면 event-aware analysis의 범위를 확장할 수 있다.

셋째, 본 연구의 사례 연구는 하나의 화재 대응 시나리오와 35×35 grid를 사용한다. 향후 다양한 grid size, patient distribution, event timing 및 disaster type을 적용하여 로그 통합 구조의 일반성을 검토할 필요가 있다.

넷째, 본 연구의 30회 반복실험은 동일한 단계적 시나리오의 결과 분포를 제시하지만, 실제 재난 현장 데이터와의 calibration 또는 external validation까지 수행하지 않았다. 향후 실제 evacuation drill 데이터나 first-responder training data와 연계하여 model validity를 강화할 수 있다[8,9,11].

V. 결 론

본 연구는 재난 대응 System-of-Systems 시뮬레이션에서 생성되는 event, agent, patient 및 spatial log를 공통 시간축에 통합하고, FIRE_OUTBREAK, FIRE_ESCALATION 및 FIRE_SUPPRESSION의 단계별 상태 변화를 시각적으로 분석하는 구조를 제안하였다. 단일 실행 결과에 대한 의존성을 줄이기 위해 동일 시나리오를 30회 반복 실행한 결과, 평균 최종 구조율은 97.00%, 평균 치료 완료율은 24.00%, 평균 최종 미탐색 tile 수는 28.97개로 나타났다. 반복 실험 결과는 특정 단일 실행의 우연한 경로에만 의존하지 않고 동일 시나리오에서 수집되는 지표의 분포를 함께 제시한다.

본 연구의 핵심 기여는 서로 다른 runtime log를 공통 frame과 event phase 기준으로 통합하고, 이벤트 경계가 표시된 시각화와 반복실험 통계 요약을 함께 제공함으로써 재난 대응 SoS의 동적 상태를 해석 가능한 smart information으로 전환하는 데 있다. 향후 연구에서는 interactive dashboard를 구현하고, communication log를 실제 MsgRouter

와 연동하며, 다양한 재난 유형과 실제 훈련 데이터터를 이용한 validation을 수행할 계획이다.

REFERENCES

- [1] C. B. Nielsen, P. G. Larsen, J. Fitzgerald, J. Woodcock, and J. Peleska, "Systems of systems engineering: Basic concepts, model-based techniques, and research directions," *ACM Computing Surveys*, vol. 48, no. 2, pp. 1-41, 2015.
- [2] W. C. Baldwin and B. Sauser, "Modeling the characteristics of system of systems," *Proc. IEEE International Conference on System of Systems Engineering*, pp. 1-6, 2009.
- [3] B. P. Zeigler and A. Muzy, "Some modeling and simulation perspectives on emergence in system-of-systems," *Spring Simulation Multi-conference*, 2016.
- [4] S. Mittal and J. L. Risco-Martín, "Simulation-based complex adaptive systems," in *Guide to Simulation-Based Disciplines*, Springer, pp. 127-150, 2017.
- [5] S. Park, Y. Shin, S. Hyun, and D.-H. Bae, "SIMVA-SoS: Simulation-based Verification and Analysis for System-of-Systems," *Proc. IEEE International Conference on System of Systems Engineering (SoSE)*, 2020.
- [6] S. Park, Z. M. Belay, and D.-H. Bae, "A simulation-based behavior analysis for MCI response system of systems," *Proc. IEEE/ACM SESoS/WDES*, pp. 2-9, 2019.
- [7] W. Manzano, V. V. Graciano Neto, and E. Y. Nakagawa, "Dynamic-SoS: An approach for the simulation of systems-of-systems dynamic architectures," *The Computer Journal*, 2018.
- [8] J. Wahlqvist, E. Ronchi, R. Lovreglio, D. Nilsson, and S. M. V. Gwynne, "The simulation of wildland-urban interface fire evacuation: The WUI-NITY platform," *Safety Science*, vol. 136, 105130, 2021.
- [9] S. M. V. Gwynne, E. Ronchi, J. Wahlqvist, A. Cuesta, and J. Gonzalez Villa, "Roxborough Park Community Wildfire Evacuation Drill: Data Collection and Model Benchmarking," *Fire Technology*, vol. 59, no. 2, pp. 879-901, 2023.
- [10] P. Wang, X. Wang, P. Luh, N. Olderman, C. Wilkie, and T. Korhonen, "CrowdEgress: A Multi-Agent Simulation Platform for Pedestrian Crowd," arXiv:2406.08190, 2024.
- [11] A. Cassity, H. Le, H. Santos, E. Priest, and J. Tao, "Dynamic Data-Driven Digital Twin Testbed for Enhanced First Responder Training and Communication," arXiv:2410.14140, 2024.
- [12] S. Borgwardt, N. Crawford, D. Horton, A. Morrison, and E. Speakman, "Evacuation Planning on Time-Expanded Networks with Integrated Wildfire Information," arXiv:2410.14500, 2024.
- [13] Y. Sun, A. Forrister, E. D. Kuligowski, R. Lovreglio, T. J. Cova, and X. Zhao, "Social Vulnerabilities and Wildfire Evacuations: A Case Study of the 2019 Kincade Fire," arXiv:2402.06639, 2024.
- [14] A. Ramadan, "Wildfire Autonomous Response and Prediction Using Cellular Automata (WARP-CA)," arXiv:2407.02613, 2024.
- [15] A. Drogoul, J. Ferber, and C. Cambier, "Multi-agent simulation as a tool for analysing emergent processes in societies," *Proc. Simulating Societies Symposium*, pp. 127-142, 1994.
- [16] S. Robinson, "Simulation: The Practice of Model Development and Use", Wiley, 2004.

저 자 소 개



송지영(중신회원)

2014년 이화여자대학교 컴퓨터공학과 학사 졸업.

2016년 한국과학기술원 전산학부 석사 졸업.

2022년 한국과학기술원 전산학부 학과 박사 졸업.

<주관심분야 : 소프트웨어공학, 소프트웨어 테스트, 소프트웨어 모델 검증, 프로젝트 관리, 시스템 오브 시스템즈, IoT, CPS>