



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2022년07월22일
(11) 등록번호 10-2423988
(24) 등록일자 2022년07월19일

(51) 국제특허분류(Int. Cl.)
G06F 11/34 (2006.01) G06F 9/38 (2006.01)
G06F 9/448 (2018.01)
(52) CPC특허분류
G06F 11/3419 (2013.01)
G06F 9/3897 (2013.01)
(21) 출원번호 10-2020-0178603
(22) 출원일자 2020년12월18일
심사청구일자 2020년12월18일
(65) 공개번호 10-2022-0073592
(43) 공개일자 2022년06월03일
(30) 우선권주장
1020200160575 2020년11월26일 대한민국(KR)
(56) 선행기술조사문헌
“RINGA: Design and verification of finite state machine for self-adaptive software at runtime”, Euijong Lee 외 4명, Information and Software Technology 93 (2018), 2017.09.20.
“모델검증을 활용한 자가-적응 소프트웨어 검증 및 전략 추출 방법”, 이의중 외 2명, 한국정보과학회 학술발표논문집, 2015.12.31.
US20200042369 A1
KR1020020095723 A

(73) 특허권자
세종대학교 산학협력단
서울특별시 광진구 능동로 209 (군자동, 세종대학교)
(72) 발명자
김영갑
서울특별시 광진구 능동로 209, 세종대학교 대양AI센터 701호
이의중
서울특별시 광진구 능동로 209, 세종대학교 대양AI센터 732호
서영덕
서울특별시 광진구 능동로 209, 세종대학교 대양AI센터 706호
(74) 대리인
특허법인이상

전체 청구항 수 : 총 10 항

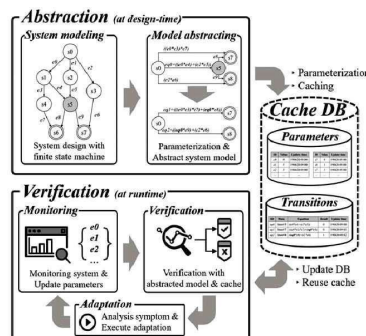
심사관 : 김계준

(54) 발명의 명칭 사물인터넷 모델 검증 방법 및 장치

(57) 요약

본 발명은 유한 상태 기계(Finite State Machine; FSM)에 포함된 복수의 상태(state)로부터 생성된 제 1 사물인터넷 모델을 기초로 제 2 사물인터넷 모델을 검증하는 방법으로서, 유한 상태 기계(Finite State Machine; FSM) 형태로 제 1 사물인터넷 모델을 생성하는 단계; 상기 제 1 사물인터넷 모델을 변수화(parameterization)하는 단계; 상기 제 1 사물인터넷 모델에 기초하여 특정 상태에 도달하는 경로를 추출하는 단계; 상기 경로를 캐시 데이터베이스(cache database)에 저장하는 단계; 상기 캐시 데이터베이스에 저장된 경로에 기반하여 제 2 사물인터넷 모델을 검증하는 단계를 포함하는, 사물인터넷 모델 검증 방법을 개시한다.

대표도



(52) CPC특허분류
G06F 9/4498 (2018.02)

이 발명을 지원한 국가연구개발사업

과제고유번호	1711103210
과제번호	2019-0-00231-002
부처명	과학기술정보통신부
과제관리(전문)기관명	정보통신기획평가원
연구사업명	정보보호핵심원천기술개발(R&D)
연구과제명	공공 인프라 안전을 위한 인공지능 기반 영상보안 기술 및 시스템 개발
기 여 율	1/1
과제수행기관명	세종대학교 산학협력단
연구기간	2020.01.01 ~ 2020.12.31

공지예외적용 : 있음

명세서

청구범위

청구항 1

유한 상태 기계(Finite State Machine; FSM)에 포함된 복수의 상태(state)로부터 생성된 제 1 사물인터넷 모델을 기초로 제 2 사물인터넷 모델을 검증하는 방법으로서,

유한 상태 기계(Finite State Machine; FSM) 형태로 제 1 사물인터넷 모델을 생성하는 단계;

상기 제 1 사물인터넷 모델을 변수화(parameterization)하는 단계;

상기 제 1 사물인터넷 모델에 기초하여 특정 상태에 도달하는 경로를 추출하는 단계;

상기 경로를 캐시 데이터베이스(cache database)에 저장하는 단계; 및

상기 캐시 데이터베이스에 저장된 경로에 기반하여 제 2 사물인터넷 모델을 검증하는 단계를 포함하는, 사물인터넷 모델 검증 방법.

청구항 2

청구항 1에 있어서,

유한 상태 기계(Finite State Machine; FSM) 형태는,

사물인터넷 모델을 복수의 노드(node) 및 노드를 연결하는 �지(edge)로 구성하는 형태인, 사물인터넷 모델 검증 방법.

청구항 3

청구항 1에 있어서,

상기 제 1 사물인터넷 모델에 기초하여 특정 상태에 도달하는 경로를 추출하는 단계는,

사용자가 미리 설정한 초기 상태에서 특정 상태에 도달하는 경로를 생성하는 단계; 및

상기 생성된 경로를 수식화하는 단계를 포함하는, 사물인터넷 모델 검증 방법.

청구항 4

청구항 3에 있어서,

상기 경로를 캐시 데이터베이스(cache database)에 저장하는 단계는,

상기 초기 상태에서 특정 상태에 도달하는 경로에서 중복되는 경로를 간소화하여 수식화하는 단계; 및

상기 간소화하여 수식화된 경로를 캐시 데이터베이스에 저장하는 단계를 포함하는, 사물인터넷 모델 검증 방법.

청구항 5

청구항 1에 있어서,

상기 캐시 데이터베이스에 저장된 경로에 기반하여 제 2 사물인터넷 모델을 검증하는 단계는,

상기 제 2 사물인터넷 모델의 구성 요소를 변수화하는 단계;

상기 제 2 사물인터넷 모델의 구성 요소를 제 1 사물인터넷 모델에 입력하는 단계; 및

입력 후 제 2 사물인터넷 모델의 실행 시간과 제 1 사물인터넷 모델의 실행 시간을 비교하여 검증하는 단계를 포함하는, 사물인터넷 모델 검증 방법.

청구항 6

유한 상태 기계(Finite State Machine; FSM)에 포함된 복수의 상태(state)로부터 생성된 제 1 사물인터넷 모델을 기초로 제 2 사물인터넷 모델을 검증하는 장치로서,

프로세서; 및

상기 프로세서를 통해 실현되는 적어도 하나의 명령을 저장하는 메모리를 포함하고,

상기 적어도 하나의 명령은,

유한 상태 기계(Finite State Machine; FSM) 형태로 제 1 사물인터넷 모델을 생성하도록 하는 명령;

상기 제 1 사물인터넷 모델을 변수화(parameterization)하도록 하는 명령;

상기 제 1 사물인터넷 모델에 기초하여 특정 상태에 도달하는 경로를 추출하도록 하는 명령;

상기 경로를 캐시 데이터베이스(cache database)에 저장하도록 하는 명령; 및

상기 캐시 데이터베이스에 저장된 경로에 기반하여 제 2 사물인터넷 모델을 검증하도록 하는 명령을 포함하는, 사물인터넷 모델 검증 장치.

청구항 7

청구항 6에 있어서,

유한 상태 기계(Finite State Machine; FSM) 형태는,

사물인터넷 모델을 복수의 노드(node) 및 노드를 연결하는 �지(edge)로 구성하는 형태인, 사물인터넷 모델 검증 장치.

청구항 8

청구항 6에 있어서,

상기 제 1 사물인터넷 모델에 기초하여 특정 상태에 도달하는 경로를 추출하도록 하는 명령은,

사용자가 미리 설정한 초기 상태에서 특정 상태에 도달하는 경로를 생성하도록 하는 명령; 및

상기 생성된 경로를 수식화하도록 하는 명령을 포함하는, 사물인터넷 모델 검증 장치.

청구항 9

청구항 8에 있어서,

상기 경로를 캐시 데이터베이스(cache database)에 저장하도록 하는 명령은,

상기 초기 상태에서 특정 상태에 도달하는 경로에서 중복되는 경로를 간소화하여 수식화하도록 하는 명령; 및

상기 간소화하여 수식화된 경로를 캐시 데이터베이스에 저장하도록 하는 명령을 포함하는, 사물인터넷 모델 검증 장치.

청구항 10

청구항 6에 있어서,

상기 캐시 데이터베이스에 저장된 경로에 기반하여 제 2 사물인터넷 모델을 검증하도록 하는 명령은,

상기 제 2 사물인터넷 모델의 구성 요소를 변수화하도록 하는 명령;

상기 제 2 사물인터넷 모델의 구성 요소를 제 1 사물인터넷 모델에 입력하도록 하는 명령; 및

입력 후 제 2 사물인터넷 모델의 실행 시간과 제 1 사물인터넷 모델의 실행 시간을 비교하여 검증하도록 하는 명령을 포함하는, 사물인터넷 모델 검증 장치.

발명의 설명

기술 분야

[0001] 본 발명은 사물인터넷 모델을 검증하는 방법 및 장치에 관한 것으로, 더욱 상세하게는 유한 상태 기계(Finite State Machine; FSM)를 활용하여 사물인터넷 모델을 검증하는 방법 및 장치에 관한 것이다.

배경 기술

[0002] 사물인터넷 모델을 검증하는 것은 시스템을 검증하는 형식적인 방법으로써 모델-검증(model-checking)은 소프트웨어 엔지니어링, 로봇 공학, 소프트웨어 검증, 사물인터넷과 같은 여러 소프트웨어 분야에 적용된다. 현재 소프트웨어 시스템은 가변적이고 역동적이며 불확실한 환경에서 운영되고 있으므로 소프트웨어 시스템의 운영 환경을 예측하기 어려운 문제가 있다.

[0003] 따라서, 소프트웨어의 검증은 소프트웨어의 견고성과 신뢰성을 확인하기 위해 소프트웨어를 설계할 때뿐만 아니라 소프트웨어의 실행시간(runtime)에 대해서도 수행되어야 한다. 특히, 사물인터넷 시스템은 개체(entity), 사용자 및 정보 리소스를 서비스와 상호 연결하므로 사물인터넷 시스템은 다양한 실행시간 환경을 생성할 수 있으므로 사물인터넷 시스템의 실행시간에 대한 검증이 필요하다.

[0004] 한편, 사물인터넷 어플리케이션 또는 시스템을 검증하기 위해서는 성능이 낮은 컴퓨팅 환경에서도 가능한 검증 기술이 필요하고, 모델-검증(model-checking) 기술은 모델링된 사물인터넷 시스템을 검증하기 위해 유용하지만 상태(state) 폭발 등의 문제가 있어 사물인터넷 환경에 적용하기 어려운 문제가 있다. 또한, 기존 추상화(abstraction) 기반 사물인터넷 모델을 검증하는 기술은 사물인터넷 모델이 복잡해지는 경우 추상화 및 검증 시간이 급격하게 증가하는 문제가 있다.

[0005] 결과적으로, 이러한 사물인터넷 시스템은 사물인터넷 모델을 구성하는 요소의 변화로 인해 빈번하게 검증이 필요하다. 또한, 자원이 제한된 사물인터넷 장치에 대한 효율적인 검증 방법 또한 필요하다.

발명의 내용

해결하려는 과제

[0006] 상기와 같은 문제점을 해결하기 위한 본 발명의 목적은 저사양 컴퓨팅 환경에서도 사물인터넷 모델을 모델-검증(model-checking) 기술을 통해 검증하는 방법을 제공하는 데 있다.

[0007] 상기와 같은 문제점을 해결하기 위한 본 발명의 다른 목적은 사물인터넷 모델을 검증하여 실행시간을 검증하는 방법을 제공하는 데 있다.

과제의 해결 수단

[0008] 상기 목적을 달성하기 위한 본 발명의 일 실시예에 따른 사물인터넷 모델 검증 방법은, 유한 상태 기계(Finite State Machine; FSM)에 포함된 복수의 상태(state)로부터 생성된 제 1 사물인터넷 모델을 기초로 제 2 사물인터넷 모델을 검증하는 방법으로서, 유한 상태 기계(Finite State Machine; FSM) 형태로 제 1 사물인터넷 모델을 생성하는 단계; 상기 제 1 사물인터넷 모델을 변수화(parameterization)하는 단계; 상기 제 1 사물인터넷 모델에 기초하여 특정 상태에 도달하는 경로를 추출하는 단계; 상기 경로를 캐시 데이터베이스(cache database)에 저장하는 단계; 및 상기 캐시 데이터베이스에 저장된 경로에 기반하여 제 2 사물인터넷 모델을 검증하는 단계를 포함할 수 있다.

[0009] 여기서, 유한 상태 기계(Finite State Machine; FSM) 형태는, 사물인터넷 모델을 복수의 노드(node) 및 노드를 연결하는 엣지(edge)로 구성하는 형태일 수 있다.

[0010] 한편, 상기 제 1 사물인터넷 모델에 기초하여 특정 상태에 도달하는 경로를 추출하는 단계는, 사용자가 미리 설정한 초기 상태에서 특정 상태에 도달하는 경로를 생성하는 단계; 및 상기 생성된 경로를 수식화하는 단계를 포함할 수 있다.

[0011] 여기서, 상기 경로를 캐시 데이터베이스(cache database)에 저장하는 단계는, 상기 초기 상태에서 특정 상태에 도달하는 경로에서 중복되는 경로를 간소화하여 수식화하는 단계; 및 상기 간소화하여 수식화된 경로를 캐시 데이터베이스에 저장하는 단계를 포함할 수 있다.

[0012] 또한, 상기 캐시 데이터베이스에 저장된 경로에 기반하여 제 2 사물인터넷 모델을 검증하는 단계는, 상기 제 2 사물인터넷 모델의 구성 요소를 변수화하는 단계; 상기 제 2 사물인터넷 모델의 구성 요소를 제 1 사물인터넷 모델에 입력하는 단계; 및 입력 후 제 2 사물인터넷 모델의 실행 시간과 제 1 사물인터넷 모델의 실행 시간을

비교하여 검증하는 단계를 포함할 수 있다.

- [0014] 상기와 같은 목적을 달성하기 위한 본 발명의 다른 실시예에 따른 사물인터넷 모델 검증 장치는 유한 상태 기계(Finite State Machine; FSM)에 포함된 복수의 상태(state)로부터 생성된 제 1 사물인터넷 모델을 기초로 제 2 사물인터넷 모델을 검증하는 장치로서, 프로세서; 및 상기 프로세서를 통해 실현되는 적어도 하나의 명령을 저장하는 메모리를 포함하고, 상기 적어도 하나의 명령은, 유한 상태 기계(Finite State Machine; FSM) 형태로 제 1 사물인터넷 모델을 생성하도록 하는 명령; 상기 제 1 사물인터넷 모델을 변수화(parameterization)하도록 하는 명령; 상기 제 1 사물인터넷 모델에 기초하여 특정 상태에 도달하는 경로를 추출하도록 하는 명령; 상기 경로를 캐시 데이터베이스(cache database)에 저장하도록 하는 명령; 및 상기 캐시 데이터베이스에 저장된 경로에 기반하여 제 2 사물인터넷 모델을 검증하도록 하는 명령을 포함할 수 있다.
- [0015] 여기서, 유한 상태 기계(Finite State Machine; FSM) 형태는, 사물인터넷 모델을 복수의 노드(node) 및 노드를 연결하는 엣지(edge)로 구성하는 형태일 수 있다.
- [0016] 한편, 상기 제 1 사물인터넷 모델에 기초하여 특정 상태에 도달하는 경로를 추출하도록 하는 명령은, 사용자가 미리 설정한 초기 상태에서 특정 상태에 도달하는 경로를 생성하도록 하는 명령; 및 상기 생성된 경로를 수식화하도록 하는 명령을 포함할 수 있다.
- [0017] 여기서, 상기 경로를 캐시 데이터베이스(cache database)에 저장하도록 하는 명령은, 상기 초기 상태에서 특정 상태에 도달하는 경로에서 중복되는 경로를 간소화하여 수식화하도록 하는 명령; 및 상기 간소화하여 수식화된 경로를 캐시 데이터베이스에 저장하도록 하는 명령을 포함할 수 있다.
- [0018] 또한, 상기 캐시 데이터베이스에 저장된 경로에 기반하여 제 2 사물인터넷 모델을 검증하도록 하는 명령은, 상기 제 2 사물인터넷 모델의 구성 요소를 변수화하도록 하는 명령; 상기 제 2 사물인터넷 모델의 구성 요소를 제 1 사물인터넷 모델에 입력하도록 하는 명령; 및 입력 후 제 2 사물인터넷 모델의 실행 시간과 제 1 사물인터넷 모델의 실행 시간을 비교하여 검증하도록 하는 명령을 포함할 수 있다.

발명의 효과

- [0019] 본 발명의 일 실시예에 따르면 기존 추상화 기반 사물인터넷 모델 검증 방법보다 추상화에 걸리는 시간이 단축될 수 있다.
- [0020] 본 발명의 다른 실시예에 따르면 기존 사물인터넷 모델 검증하는 방법들에 비해 검증 시간이 단축될 수 있다.

도면의 간단한 설명

- [0021] 도 1은 본 발명의 일 실시예에 따른 사물인터넷 모델 검증 방법의 개념도이다.
- 도 2는 본 발명의 일 실시예에 따른 사물인터넷 모델 검증 방법의 동작 순서도이다.
- 도 3a는 사물인터넷 모델을 생성하는 과정을 설명하는 제 1 예시도이다.
- 도 3b는 사물인터넷 모델을 생성하는 과정을 설명하는 제 2 예시도이다.
- 도 4는 사물인터넷 모델을 변수화하는 과정을 설명하는 도면이다.
- 도 5는 특정 상태에 도달하는 경로를 추출하는 과정을 설명하는 도면이다.
- 도 6은 특정 상태에 도달하는 경로를 추출하는데 활용되는 제 1 알고리즘이다.
- 도 7은 특정 상태에 도달하는 경로를 추출하는데 활용되는 제 2 알고리즘이다.
- 도 8은 경로를 캐시 데이터베이스에 저장하는 과정을 설명하는 도면이다.
- 도 9는 실행시간을 비교하여 검증하는 과정의 개념도이다.
- 도 10은 실행시간을 비교하여 검증하는데 활용되는 알고리즘이다.
- 도 11은 종래기술과 본 발명을 비교한 제 1 예시도이다.
- 도 12는 종래기술과 본 발명을 비교한 제 2 예시도이다.
- 도 13은 종래기술과 본 발명을 비교한 제 3 예시도이다.

도 14는 종래기술과 본 발명을 비교한 제 4 예시도이다.

도 15는 본 발명의 다른 실시예에 따른 사물인터넷 모델 검증 장치의 블록 구성도이다.

발명을 실시하기 위한 구체적인 내용

- [0022] 본 발명은 다양한 변경을 가할 수 있고 여러 가지 실시예를 가질 수 있는 바, 특정 실시예들을 도면에 예시하고 상세한 설명에 상세하게 설명하고자 한다. 그러나, 이는 본 발명을 특정한 실시 형태에 대해 한정하려는 것이 아니며, 본 발명의 사상 및 기술 범위에 포함되는 모든 변경, 균등물 내지 대체물을 포함하는 것으로 이해되어야 한다. 각 도면을 설명하면서 유사한 참조부호를 유사한 구성요소에 대해 사용하였다.
- [0023] 제1, 제2, A, B 등의 용어는 다양한 구성요소들을 설명하는 데 사용될 수 있지만, 상기 구성요소들은 상기 용어들에 의해 한정되어서는 안 된다. 상기 용어들은 하나의 구성요소를 다른 구성요소로부터 구별하는 목적으로만 사용된다. 예를 들어, 본 발명의 권리 범위를 벗어나지 않으면서 제1 구성요소는 제2 구성요소로 명명될 수 있고, 유사하게 제2 구성요소도 제1 구성요소로 명명될 수 있다. "및/또는"이라는 용어는 복수의 관련된 기재된 항목들의 조합 또는 복수의 관련된 기재된 항목들 중의 어느 항목을 포함한다.
- [0024] 어떤 구성요소가 다른 구성요소에 "연결되어" 있다거나 "접속되어" 있다고 언급된 때에는, 그 다른 구성요소에 직접적으로 연결되어 있거나 또는 접속되어 있을 수도 있지만, 중간에 다른 구성요소가 존재할 수도 있다고 이해되어야 할 것이다. 반면에, 어떤 구성요소가 다른 구성요소에 "직접 연결되어" 있다거나 "직접 접속되어" 있다고 언급된 때에는, 중간에 다른 구성요소가 존재하지 않는 것으로 이해되어야 할 것이다.
- [0025] 본 출원에서 사용한 용어는 단지 특정한 실시예를 설명하기 위해 사용된 것으로, 본 발명을 한정하려는 의도가 아니다. 단수의 표현은 문맥상 명백하게 다르게 뜻하지 않는 한, 복수의 표현을 포함한다. 본 출원에서, "포함하다" 또는 "가지다" 등의 용어는 명세서상에 기재된 특징, 숫자, 단계, 동작, 구성요소, 부품 또는 이들을 조합한 것이 존재함을 지정하려는 것이지, 하나 또는 그 이상의 다른 특징들이나 숫자, 단계, 동작, 구성요소, 부품 또는 이들을 조합한 것들의 존재 또는 부가 가능성을 미리 배제하지 않는 것으로 이해되어야 한다.
- [0026] 다르게 정의되지 않는 한, 기술적이거나 과학적인 용어를 포함해서 여기서 사용되는 모든 용어들은 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자에 의해 일반적으로 이해되는 것과 동일한 의미를 가지고 있다. 일반적으로 사용되는 사전에 정의되어 있는 것과 같은 용어들은 관련 기술의 문맥 상 가지는 의미와 일치하는 의미를 가지는 것으로 해석되어야 하며, 본 출원에서 명백하게 정의하지 않는 한, 이상적이거나 과도하게 형식적인 의미로 해석되지 않는다.
- [0028] 이하, 본 발명에 따른 바람직한 실시예를 첨부된 도면을 참조하여 상세하게 설명한다.
- [0029] 도 1은 본 발명의 일 실시예에 따른 사물인터넷 모델 검증 방법의 개념도이다.
- [0030] 도 1을 참고하면, 본 발명의 사물인터넷 검증 방법은 추상화(abstraction) 단계 및 검증(verification) 단계로써 2단계로 분류될 수 있다.
- [0031] 여기서, 추상화 단계는 사물인터넷 모델을 생성하고, 생성된 사물인터넷 모델을 추상화하는 단계를 의미할 수 있다. 특히, 사물인터넷 모델은 유한 상태 기계(Finite State Machine; FSM) 형태로 모델링될 수 있다. 또한, 사물인터넷 모델은 추상화 단계를 통해 캐시 데이터베이스(Database; DB)를 생성하고, 생성된 캐시 데이터베이스는 이후 검증 단계에서 활용될 수 있다.
- [0032] 한편, 검증 단계는 모니터링(monitring) 단계, 검증(verification) 단계 및 적응(adaptaion) 단계, 총 3 단계로 분류될 수 있고, 이는 반복적으로 수행될 수 있다.
- [0033] 여기서, 모니터링 단계는 검증에 필요한 다양한 정보를 실시간으로 수집하는 단계를 의미할 수 있고, 검증 단계는 캐시 데이터베이스에 저장된 사물인터넷 모델(생성된 사물인터넷 모델)과 실시간 데이터에 기반한 현재 사물인터넷 모델을 비교하여 검증하는 단계를 의미할 수 있고, 적응 단계는 검증 단계에서 검증된 결과를 실행하는 단계를 의미할 수 있다.
- [0034] 본 발명의 사물인터넷 모델 검증 방법은 추후 구체적으로 서술한다.
- [0036] 도 2는 본 발명의 일 실시예에 따른 사물인터넷 모델 검증 방법의 동작 순서도이다.
- [0037] 도 2를 참고하면, 본 발명의 일 실시예에 따른 사물인터넷 모델 검증 방법은, 유한 상태 기계(Finite State Machine; FSM)에 포함된 복수의 상태(state)로부터 생성된 제 1 사물인터넷 모델을 기초로 제 2 사물인터넷 모

모델을 검증하는 방법으로서, 유한 상태 기계(Finite State Machine; FSM) 형태로 제 1 사물인터넷 모델을 생성하는 단계(S110)를 포함할 수 있다.

[0038] 여기서, 유한 상태 기계(Finite State Machine; FSM) 형태는, 사물인터넷 모델을 복수의 노드(node) 및 노드를 연결하는 엣지(edge)로 구성하는 형태일 수 있다.

[0039] 또한, 본 발명은 상기 제 1 사물인터넷 모델을 변수화(parameterization)하는 단계(S120)를 포함할 수 있다.

[0040] 또한, 본 발명은 상기 제 1 사물인터넷 모델에 기초하여 특정 상태에 도달하는 경로를 추출하는 단계(S130)를 포함할 수 있다.

[0041] 여기서, 상기 제 1 사물인터넷 모델에 기초하여 특정 상태에 도달하는 경로를 추출하는 단계는, 사용자가 미리 설정한 초기 상태에서 특정 상태에 도달하는 경로를 생성하는 단계; 및 상기 생성된 경로를 수식화하는 단계를 포함할 수 있다.

[0042] 또한, 본 발명은 상기 경로를 캐시 데이터베이스(cache database)에 저장하는 단계(S140)를 포함할 수 있다.

[0043] 여기서, 상기 경로를 캐시 데이터베이스(cache database)에 저장하는 단계는, 상기 초기 상태에서 특정 상태에 도달하는 경로에서 중복되는 경로를 간소화하여 수식화하는 단계; 및 상기 간소화하여 수식화된 경로를 캐시 데이터베이스에 저장하는 단계를 포함할 수 있다.

[0044] 또한, 본 발명은 상기 캐시 데이터베이스에 저장된 경로에 기반하여 제 2 사물인터넷 모델을 검증하는 단계(S150)를 포함할 수 있다.

[0045] 여기서, 상기 캐시 데이터베이스에 저장된 경로에 기반하여 제 2 사물인터넷 모델을 검증하는 단계는, 상기 제 2 사물인터넷 모델의 구성 요소를 변수화하는 단계; 상기 제 2 사물인터넷 모델의 구성 요소를 제 1 사물인터넷 모델에 입력하는 단계; 및 입력 후 제 2 사물인터넷 모델의 실행 시간과 제 1 사물인터넷 모델의 실행 시간을 비교하여 검증하는 단계를 포함할 수 있다.

[0047] 도 3a는 사물인터넷 모델을 생성하는 과정을 설명하는 제 1 예시도이고, 도 3b는 사물인터넷 모델을 생성하는 과정을 설명하는 제 2 예시도이다.

[0048] 사물인터넷 모델을 생성하는 과정은 전이 시스템(transition system)으로 사물인터넷 모델을 생성하는 단계일 수 있다. 한편, 사물인터넷 모델은 도 3a 및 도 3b와 같이 정의될 수 있다. 다만, 사물인터넷 모델의 정의는 이에 한정되지 않는다.

[0049]

[0050] 도 4는 사물인터넷 모델을 변수화하는 과정을 설명하는 도면이다.

[0051] 본 발명에 따르면, 생성된 사물인터넷 모델을 검증하는데 필요한 요소로 변수화(parameterization)할 수 있다. 도 4를 참고하면, 생성된 사물인터넷 모델은 s0(상태0), s1, s2, s3, s4, s5, s6 및 s7을 포함하는 노드(node) 및 노드와 노드를 연결하는 e0, e1, e2, e3, e4, e5, e6, e7, e8 및 e9을 포함하는 엣지(edge)로써 표현될 수 있다.

[0052] 한편, 본 발명에 따르면 생성된 사물인터넷 모델을 추후 현재 사물인터넷 모델을 검증하는데 필요한 요소로써 변수화할 수 있다. 예를 들어, 도 4의 생성된 사물인터넷 모델에서 검증에 필요한 요소 e0 내지 e9를 추출하여 변수화할 수 있다.

[0054] 도 5는 특정 상태에 도달하는 경로를 추출하는 과정을 설명하는 도면이다.

[0055] 본 발명에 따르면 생성된 사물인터넷 모델을 변수화하고, 변수화된 값에 기반하여 생성된 사물인터넷 모델을 추상화(abstraction)할 수 있다. 즉, 추상화는 생성된 사물인터넷 모델에서 특정 상태에 도달하는 경로를 추출하고, 추출된 경로를 수식화하는 것을 의미할 수 있다.

[0056] 예를 들어, 도 5를 참고하면, 사용자가 설정한 초기 상태(s0)에서 특정 상태(s3 또는 s4)에 도달하는 경로를 추출하면 다음과 같다. 먼저, s0 에서 s3에 도달하는 경로는 e0 및 e2를 거치는 경로1 또는 e0, e1 및 e5를 거치는 경로2로 구성될 수 있다. 이를 수식화하면, $(e0 * e2) + (e0 * e1 * e5)$ 또는 $(e0 \& e2) \parallel (e0 \& e1 \& e5)$ 와 같다. 한편, s0에서 s4에 도달하는 경로를 수식화하면, $(e0 * e1 * e6)$ 또는 $(e0 \& e1 \& e6)$ 와 같다.

[0058] 도 6은 특정 상태에 도달하는 경로를 추출하는데 활용되는 제 1 알고리즘이고, 도 7은 특정 상태에 도달하는 경

로를 추출하는데 활용되는 제 2 알고리즘이다.

- [0059] 도 6 및 도 7을 참고하면, 생성된 사물인터넷 모델에 기초하여 특정 상태에 도달하는 경로를 알 수 있다. 즉, 상기 알고리즘에 따라 사용자가 미리 설정한 초기 상태에서 특정 상태에 도달하는 경로를 생성하고, 생성된 경로를 수식화할 수 있다.
- [0061] 도 8은 경로를 캐시 데이터베이스에 저장하는 과정을 설명하는 도면이다.
- [0062] 본 발명에 따르면, 특정 상태에 도달하는 경로를 수식화하고, 수식화된 경로에 있어서 중복되는 경로를 간소화하고, 간소화된 경로를 캐시 데이터베이스(DB)에 저장할 수 있다. 예를 들어, 도 8을 참고하여 초기 상태 s0에서 특정 상태 s6 또는 s7에 도달하는 경로를 살펴보면, s0에서 직접적으로 s6 또는 s7에 도달하는 경로 1 및 s0에서 s5를 거쳐서 s6 또는 s7에 도달하는 경로2가 있을 수 있다.
- [0063] 경로 1을 수식화하면, s0에서 직접적으로 s6에 도달하는 경로는, $(e0 * e3) * e7$ 로 수식화될 수 있고, s0에서 직접적으로 s7에 도달하는 경로는, $(e2 * e6)$ 로 수식화될 수 있다. 경로2를 수식화하면, s0에서 s5에 도달하는 경로는, $((e0 * e4) + (e1 * e5) = cq0)$ 로 수식화되고, s5에서 s6에 도달하는 경로는 e8, s5에서 s7에 도달하는 경로는 e9로 수식화될 수 있다.
- [0064] 한편, 이를 캐시 데이터베이스에 저장하기 전에 간소화하면 다음과 같다. s0에서 s6에 도달하는 경로는, $(e0 * e3) * e7 + (cq0 * e8)$ 로 간소화하여 수식화될 수 있고, s0에서 s7에 도달하는 경로는, $(cq0 * e9) + (e2 * e6)$ 로 간소화하여 수식화될 수 있다. 즉, 본 발명에 따르면 s0에서 s5를 거쳐서 s6 또는 s7에 도달하는 경로2를 간소화하여 나타낼 수 있고, 이와 같이 간소화된 경로를 캐시 데이터베이스에 저장할 수 있다.
- [0066] 도 9는 실행시간을 비교하여 검증하는 과정의 개념도이다.
- [0067] 한편, 도 9는 본 발명에 따라 생성된 사물인터넷 모델과 검증하고자 하는 사물인터넷 모델을 검증하는 과정을 나타내고 있다. 구체적으로는, 검증하고자 하는 사물인터넷 모델을 검증하는 과정은 검증하고자 하는 사물인터넷 모델(이하, 검증 대상 사물인터넷 모델)을 모니터링하고, 검증 대상 사물인터넷 모델을 변수화하고, 변수화된 검증 대상 사물인터넷 모델을 추상화하고, 추상화된 검증 대상 사물인터넷 모델의 변수를 본 발명에 따라 생성된 사물인터넷 모델의 경로에 포함된 변수로써 입력하여 실행시간을 비교함으로써 검증할 수 있다.
- [0069] 도 10은 실행시간을 비교하여 검증하는데 활용되는 알고리즘이다.
- [0070] 도 10을 참고하면, 본 발명에 따라 생성된 사물인터넷 모델과 검증 대상 사물인터넷 모델의 실행시간 비교를 통해 검증을 수행할 수 있다.
- [0072] 도 11은 종래기술과 본 발명을 비교한 제 1 예시도이고, 도 12는 종래기술과 본 발명을 비교한 제 2 예시도이고, 도 13은 종래기술과 본 발명을 비교한 제 3 예시도이고, 도 14는 종래기술과 본 발명을 비교한 제 4 예시도이다.
- [0073] 도 11 및 도 12를 참고하면, 종래기술에 따른 추상화 기반 사물인터넷 모델 검증 방법(RINGA)에 비해 본 발명의 사물인터넷 모델 검증 방법은 추상화에 걸리는 시간이 단축된 것을 알 수 있다.
- [0074] 한편, 도 13 및 도 14를 참고하면, 종래기술에 따른 사물인터넷 모델 검증 방법에 비해, 본 발명의 사물인터넷 모델 검증 방법이 사물인터넷 모델을 검증하는데 소요되는 시간이 단축되는 것을 알 수 있다.
- [0075]
- [0076] 도 15는 본 발명의 다른 실시예에 따른 사물인터넷 모델 검증 장치의 블록 구성도이다.
- [0077] 도 15를 참조하면, 본 발명의 일 실시예에 따른 사물인터넷 모델 검증 장치(100)는 프로세서(110) 및 프로세서를 통해 실행되는 적어도 하나의 명령 및 명령 수행의 결과를 저장하는 메모리(120) 및 네트워크와 연결되어 통신을 수행하는 송수신 장치(130)를 포함할 수 있다.
- [0078] 사물인터넷 모델 검증 장치(100)는 또한, 입력 인터페이스 장치(140), 출력 인터페이스 장치(150), 저장 장치(160) 등을 더 포함할 수 있다. 사물인터넷 모델 검증 장치(100)에 포함된 각각의 구성 요소들은 버스(Bus)(170)에 의해 연결되어 서로 통신을 수행할 수 있다.
- [0079] 프로세서(110)는 메모리(120) 및 저장 장치(160) 중에서 적어도 하나에 저장된 프로그램 명령(program command)을 실행할 수 있다. 프로세서(110)는 중앙 처리 장치(central processing unit, CPU), 그래픽 처리 장치(graphics processing unit, GPU), 또는 본 발명의 실시예에 따른 방법들이 수행되는 전용의 프로세서를 의미할

수 있다. 메모리(120) 및 저장 장치(160) 각각은 휘발성 저장 매체 및 비휘발성 저장 매체 중에서 적어도 하나로 구성될 수 있다. 예를 들어, 메모리(120)는 읽기 전용 메모리(read only memory, ROM) 및 랜덤 액세스 메모리(random access memory, RAM) 중에서 적어도 하나로 구성될 수 있다.

[0080] 저장 장치(160)는 유한 상태 기계 형태로 생성된 사물인터넷 모델, 생성된 사물인터넷 모델을 변수화하였을 때의 변수 값, 특정 상태에 도달하는 경로의 수식, 검증 대상 사물인터넷 모델, 검증 대상 사물인터넷 모델을 변수화하였을 때의 변수 값을 저장할 수 있다.

[0081] 여기서, 적어도 하나의 명령은, 유한 상태 기계(Finite State Machine; FSM) 형태로 제 1 사물인터넷 모델을 생성하도록 하는 명령; 상기 제 1 사물인터넷 모델을 변수화(parameterization)하도록 하는 명령; 상기 제 1 사물인터넷 모델에 기초하여 특정 상태에 도달하는 경로를 추출하도록 하는 명령; 상기 경로를 캐시 데이터베이스(cache database)에 저장하도록 하는 명령; 및 상기 캐시 데이터베이스에 저장된 경로에 기반하여 제 2 사물인터넷 모델을 검증하도록 하는 명령을 포함할 수 있다.

[0082] 여기서, 유한 상태 기계(Finite State Machine; FSM) 형태는, 사물인터넷 모델을 복수의 노드(node) 및 노드를 연결하는 엣지(edge)로 구성하는 형태일 수 있다.

[0083] 한편, 상기 제 1 사물인터넷 모델에 기초하여 특정 상태에 도달하는 경로를 추출하도록 하는 명령은, 사용자가 미리 설정한 초기 상태에서 특정 상태에 도달하는 경로를 생성하도록 하는 명령; 및 상기 생성된 경로를 수식화하도록 하는 명령을 포함할 수 있다.

[0084] 여기서, 상기 경로를 캐시 데이터베이스(cache database)에 저장하도록 하는 명령은, 상기 초기 상태에서 특정 상태에 도달하는 경로에서 중복되는 경로를 간소화하여 수식화하도록 하는 명령; 및 상기 간소화하여 수식화된 경로를 캐시 데이터베이스에 저장하도록 하는 명령을 포함할 수 있다.

[0085] 또한, 상기 캐시 데이터베이스에 저장된 경로에 기반하여 제 2 사물인터넷 모델을 검증하도록 하는 명령은, 상기 제 2 사물인터넷 모델의 구성 요소를 변수화하도록 하는 명령; 상기 제 2 사물인터넷 모델의 구성 요소를 제 1 사물인터넷 모델에 입력하도록 하는 명령; 및 입력 후 제 2 사물인터넷 모델의 실행 시간과 제 1 사물인터넷 모델의 실행 시간을 비교하여 검증하도록 하는 명령을 포함할 수 있다.

[0087] 본 발명의 실시예에 따른 방법의 동작은 컴퓨터로 읽을 수 있는 기록매체에 컴퓨터가 읽을 수 있는 프로그램 또는 코드로서 구현하는 것이 가능하다. 컴퓨터가 읽을 수 있는 기록매체는 컴퓨터 시스템에 의해 읽혀질 수 있는 정보가 저장되는 모든 종류의 기록장치를 포함한다. 또한 컴퓨터가 읽을 수 있는 기록매체는 네트워크로 연결된 컴퓨터 시스템에 분산되어 분산 방식으로 컴퓨터로 읽을 수 있는 프로그램 또는 코드가 저장되고 실행될 수 있다.

[0088] 또한, 컴퓨터가 읽을 수 있는 기록매체는 롬(rom), 램(ram), 플래시 메모리(flash memory) 등과 같이 프로그램 명령을 저장하고 수행하도록 특별히 구성된 하드웨어 장치를 포함할 수 있다. 프로그램 명령은 컴파일러(compiler)에 의해 만들어지는 것과 같은 기계어 코드뿐만 아니라 인터프리터(interpreter) 등을 사용해서 컴퓨터에 의해 실행될 수 있는 고급 언어 코드를 포함할 수 있다.

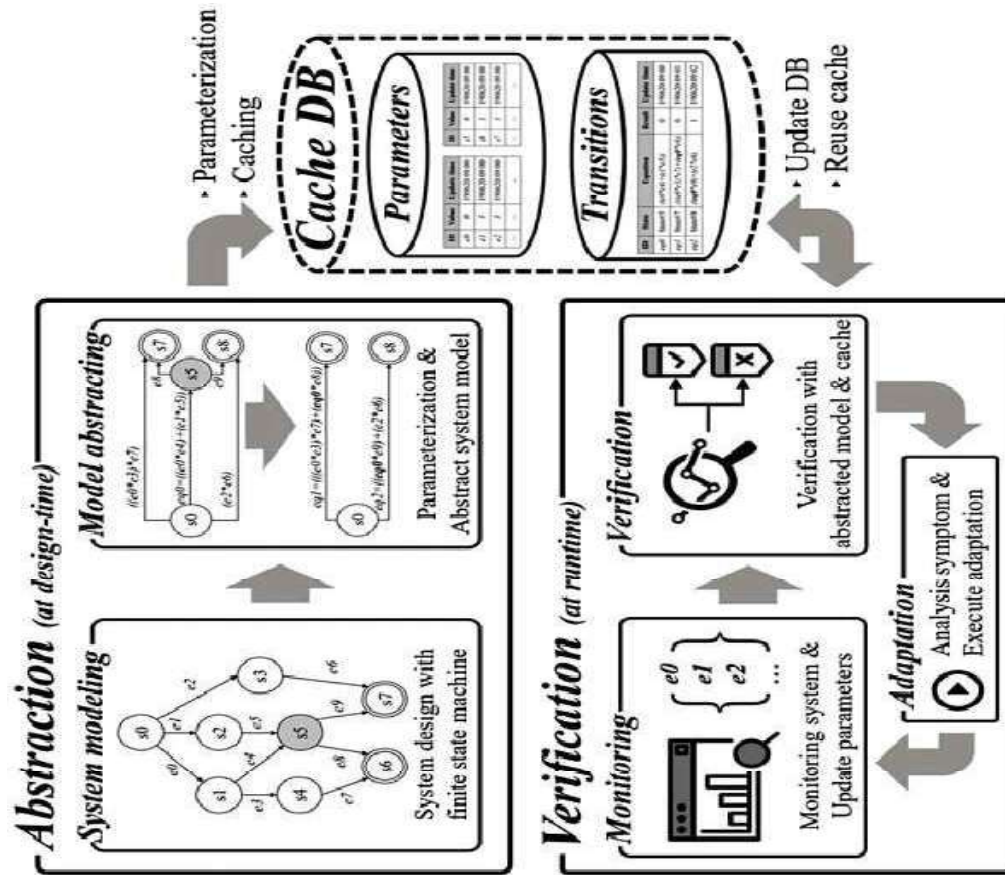
[0089] 본 발명의 일부 측면들은 장치의 문맥에서 설명되었으나, 그것은 상응하는 방법에 따른 설명 또한 나타낼 수 있고, 여기서 블록 또는 장치는 방법 단계 또는 방법 단계의 특징에 상응한다. 유사하게, 방법의 문맥에서 설명된 측면들은 또한 상응하는 블록 또는 아이템 또는 상응하는 장치의 특징으로 나타낼 수 있다. 방법 단계들의 몇몇 또는 전부는 예를 들어, 마이크로프로세서, 프로그램 가능한 컴퓨터 또는 전자 회로와 같은 하드웨어 장치에 의해(또는 이용하여) 수행될 수 있다. 몇몇의 실시예에서, 가장 중요한 방법 단계들의 하나 이상은 이와 같은 장치에 의해 수행될 수 있다.

[0090] 실시예들에서, 프로그램 가능한 로직 장치(예를 들어, 필드 프로그래머블 게이트 어레이)가 여기서 설명된 방법들의 기능의 일부 또는 전부를 수행하기 위해 사용될 수 있다. 실시예들에서, 필드 프로그래머블 게이트 어레이는 여기서 설명된 방법들 중 하나를 수행하기 위한 마이크로프로세서와 함께 작동할 수 있다. 일반적으로, 방법들은 어떤 하드웨어 장치에 의해 수행되는 것이 바람직하다.

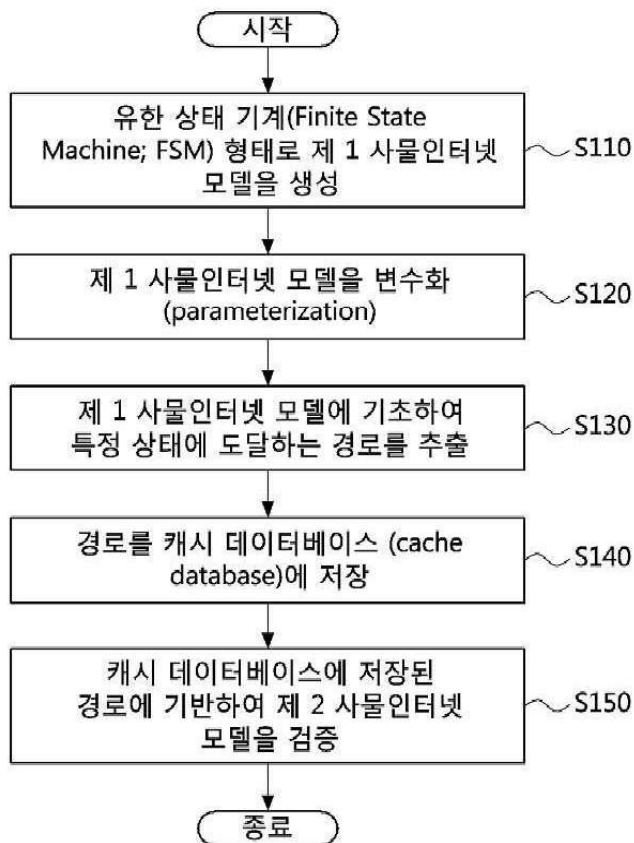
[0091] 이상 본 발명의 바람직한 실시예를 참조하여 설명하였지만, 해당 기술 분야의 숙련된 당업자는 하기의 특허 청구의 범위에 기재된 본 발명의 사상 및 영역으로부터 벗어나지 않는 범위 내에서 본 발명을 다양하게 수정 및 변경시킬 수 있음을 이해할 수 있을 것이다.

도면

도면1



도면2



도면3a

A transition system is a tuple $(S, Act, \rightarrow, I, AP, L)$, where:

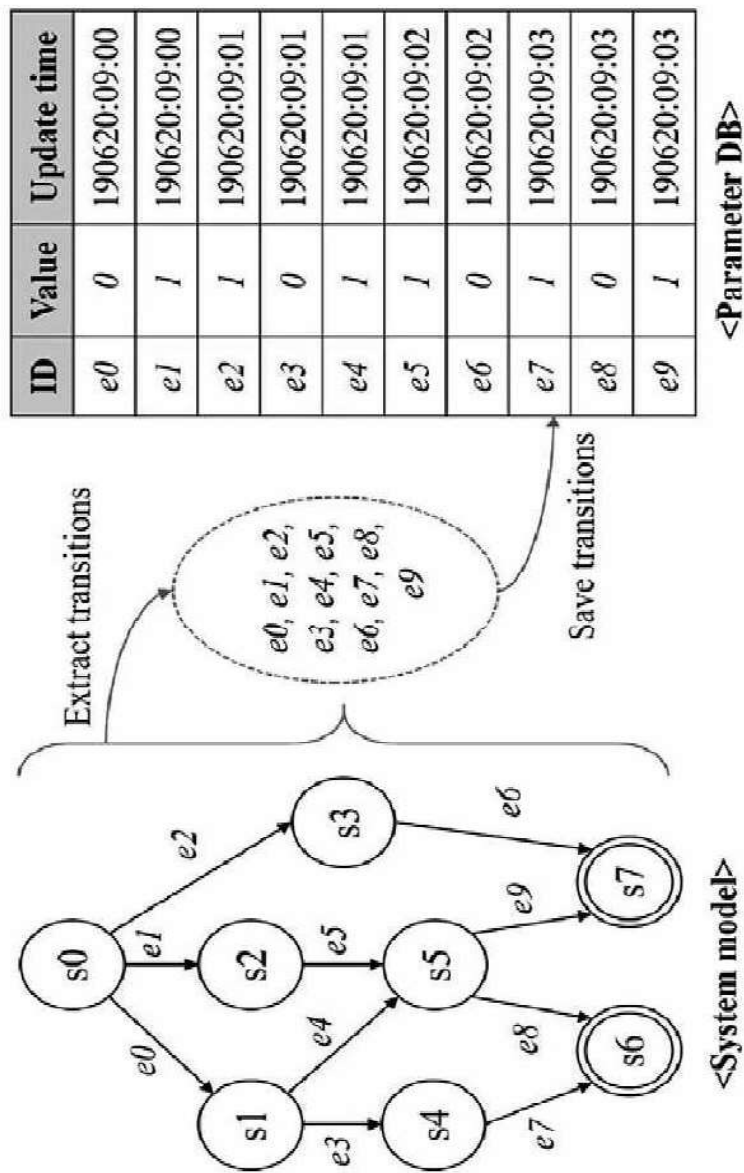
- 1) S is the set of states;
- 2) Act is the set of actions;
- 3) $\rightarrow \subseteq S \times Act \times S$ is the transition relation;
- 4) $I \subseteq S$ is the set of initial states;
- 5) AP is the set of atomic propositions;
- 6) $L : S \rightarrow 2AP$ is a power of the label function.

도면3b

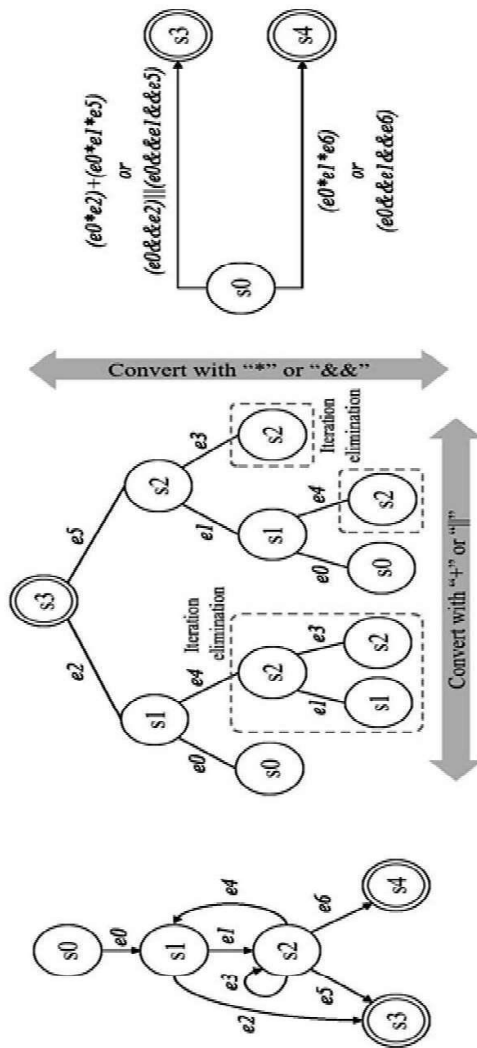
With this definition of transitions, the C-FSM can be expressed as a tuple $(S, s_0, \rightarrow, I, AP, L)$, where:

- 1) S is the set of states;
- 2) the states classified into three types $\{S_{\text{normal}}, S_{\text{cache}}, S_{\text{end}}\} \subseteq S$;
- 3) s_0 is an initial state;
- 4) S_{end} is a set of end states;
- 5) S_{cache} is a set of states which have two or more out-transitions;
- 6) S_{normal} is a set of states which is not an initial state, S_{cache} and S_{end} ;
- 7) $\rightarrow \subseteq S \times S$ is the transition relation;
- 8) AP is the set of atomic propositions;
- 9) $L : S \rightarrow 2^{AP}$ is a power of the label function.

도면4



도면5



도면6

Algorithm 1 Pseudocode of the Abstracting Phase With Cache

Input: System model as finite state machine(FSM)

Output: Abstracted finite state machine(CA-FSM)

- 1: $CA - FSM = null$;
- 2: **for** FSM.nextTransition **do**
- 3: $cachedDB.papameter(transition)$;
- 4: **end for**
- 5: **for** FSM.hasEndState **do**
- 6: $CA - FSM + = searchNext(FSM, endStateNow)$;
- 7: **end for**
- 8: **return** $CA - FSM$

도면7

Algorithm 2 Pseudocode of Path Extraction

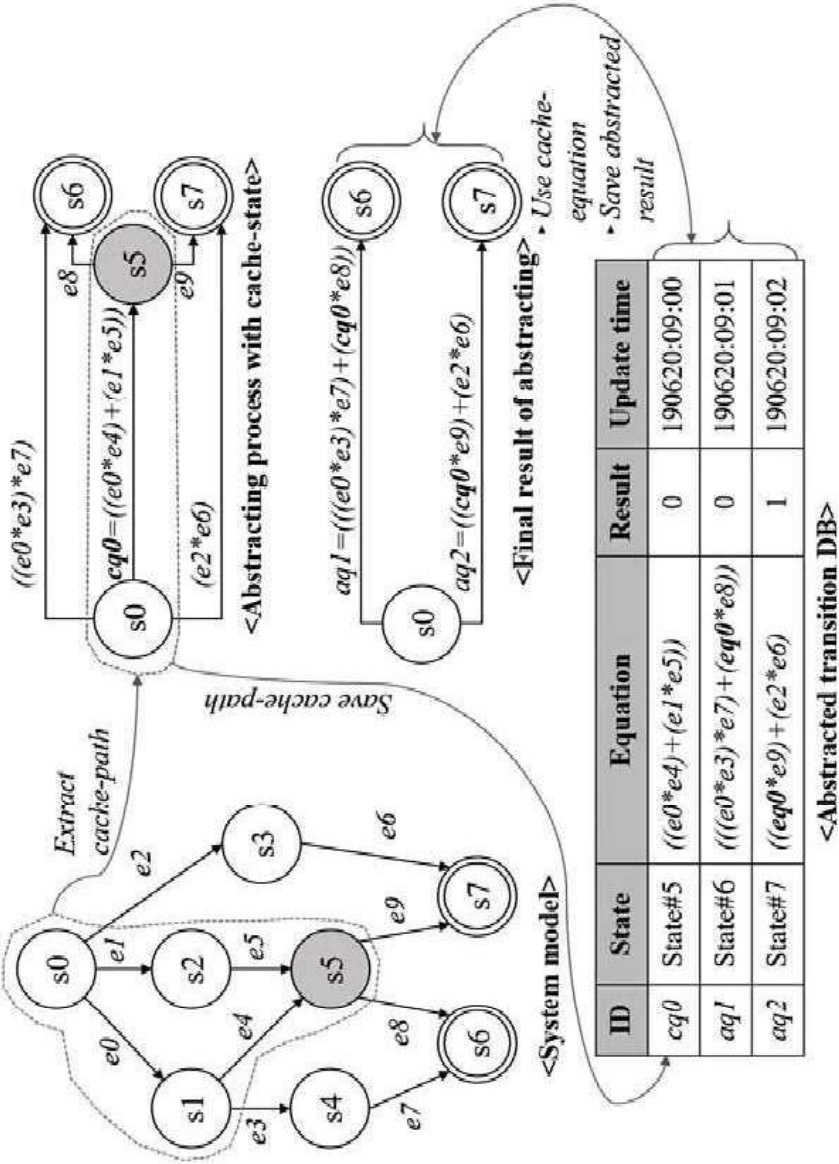
Input: System model as finite state machine (FSM), current state id (stateNow)**Output:** Path consisted with transitions

```

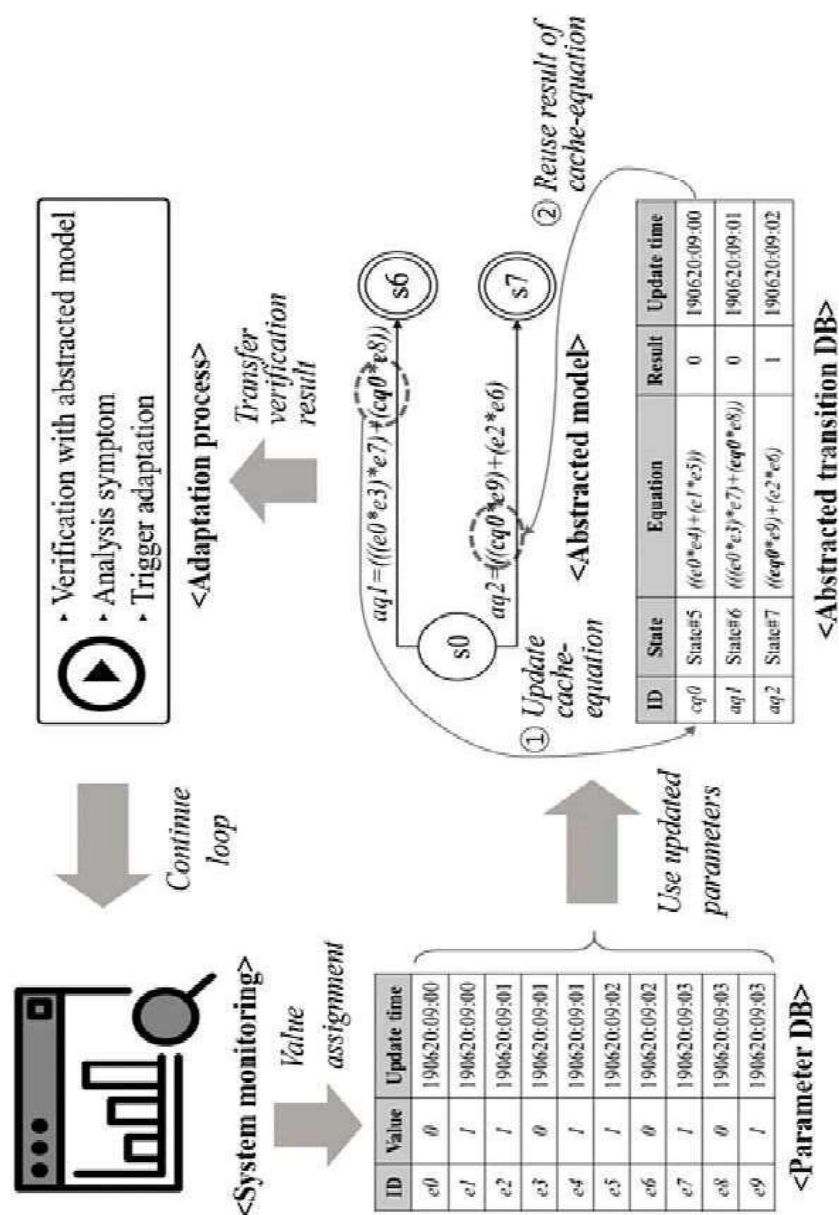
1: returnPath = null;
2: for stateNow.hasInTranstion do
3:   if stateNext = initialState then
4:     returnPath+ = stateNow.outTranstion
5:   else
6:     if nextState = normalState then
7:       if cacheDB.has(nextState) then
8:         returnPath+ = cached.path(nextState);
9:       else
10:        returnPath = stateNow.outTranstion +
          searchNext(nextState, FSM);
11:      end if
12:    end if
13:  end if
14: end for
15: if stateNow = cacheState then
16:   cached.add(stateNow, path);
17:   returnPath = cached.path(stateNow);
18: end if
19: return returnPath;

```

도면8



도면9



도면10

Algorithm 3 Pseudocode of the Abstracting Phase With Cache

Input: Abstracted finite state machine (CA-FSM), cached parameter list (parameters), cached equations (cacheEqs), abstracted transitions (absTrans)

Output: None

```

1: for true do
2:   for parameters.hasNext do
3:     updateDB(monitor(parameterNow));
4:   end for
5:   for cacheEqs.hasNext do
6:     resultEqu = calculate(equNow, parameters);
7:     updateDB(equNow, resultEqu);
8:   end for
9:   for absTrans.hasNext do
10:    resultTran = calcualte(absTranNow,
      parameters, cacheEqs);
11:    updatdDB(absTrannow, resultTran);
12:   end for
13:   triggers = analysis(cacheDB.getAbsTrans);
14:   execute(triggers);
15: end for

```

도면11

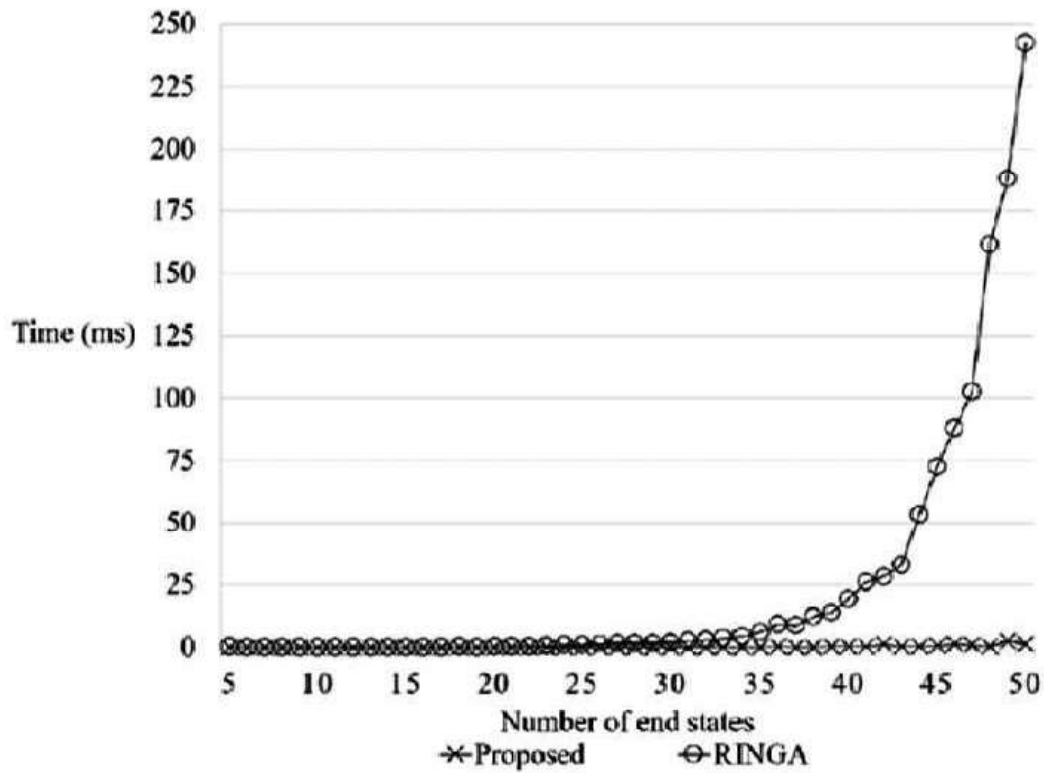


Fig. 6. Results of model abstraction with increasing states (with two transitions and three end states).

도면12

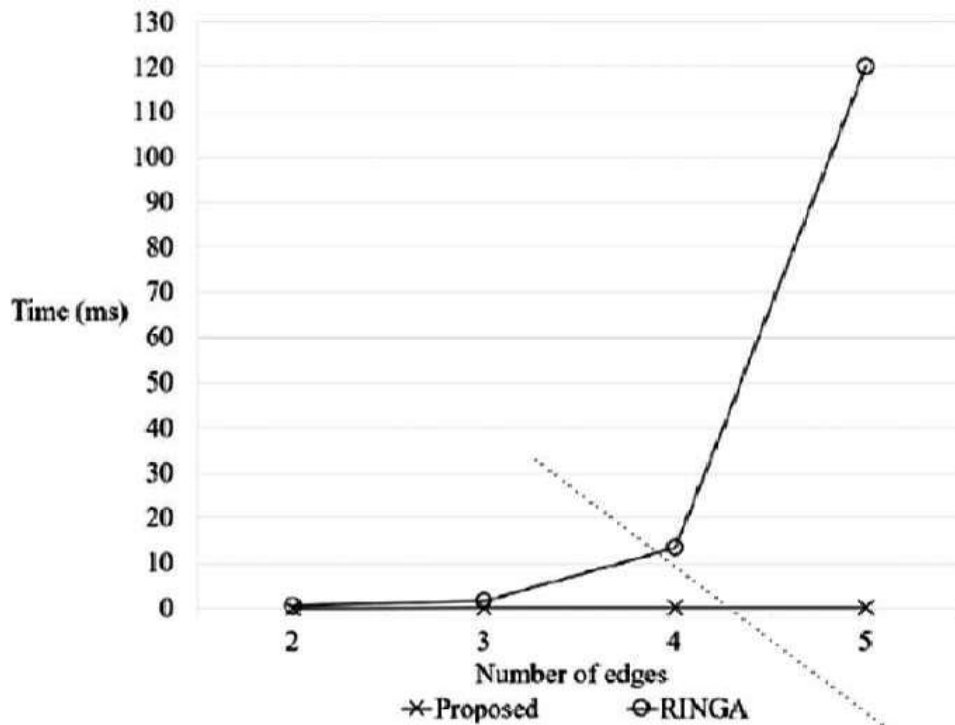


Fig. 7. Result of model abstraction with increasing transitions (with 15 states and three end states).

도면13

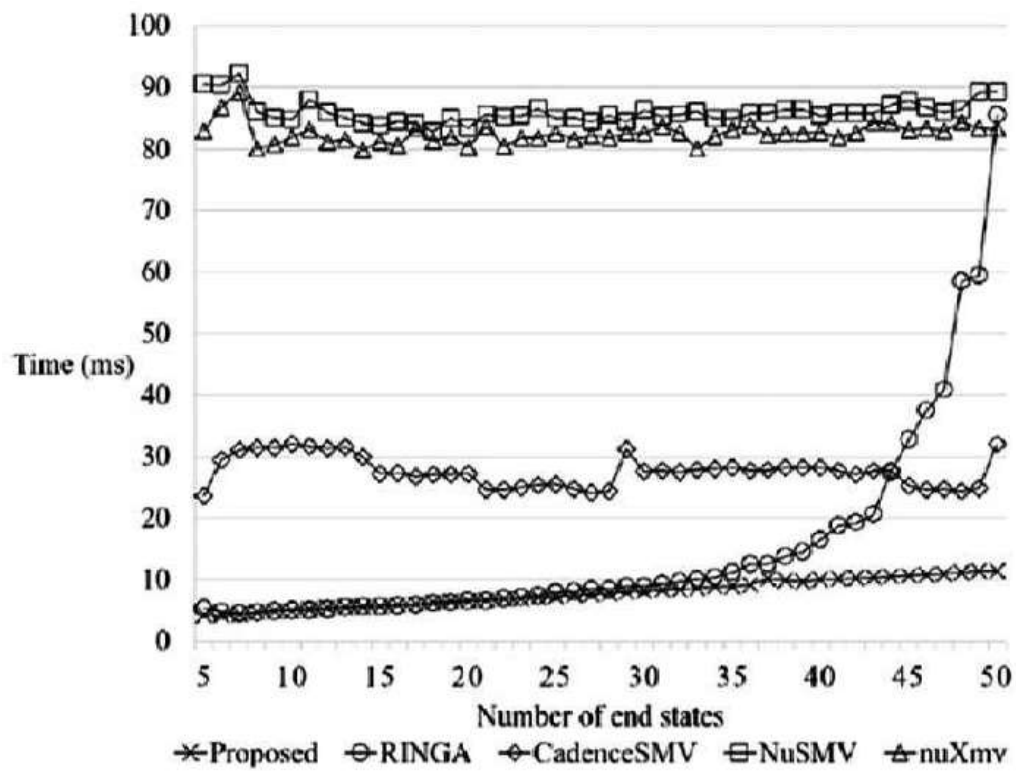


Fig. 8. Results of runtime performance with increasing states (with two transitions and three end states).

도면14

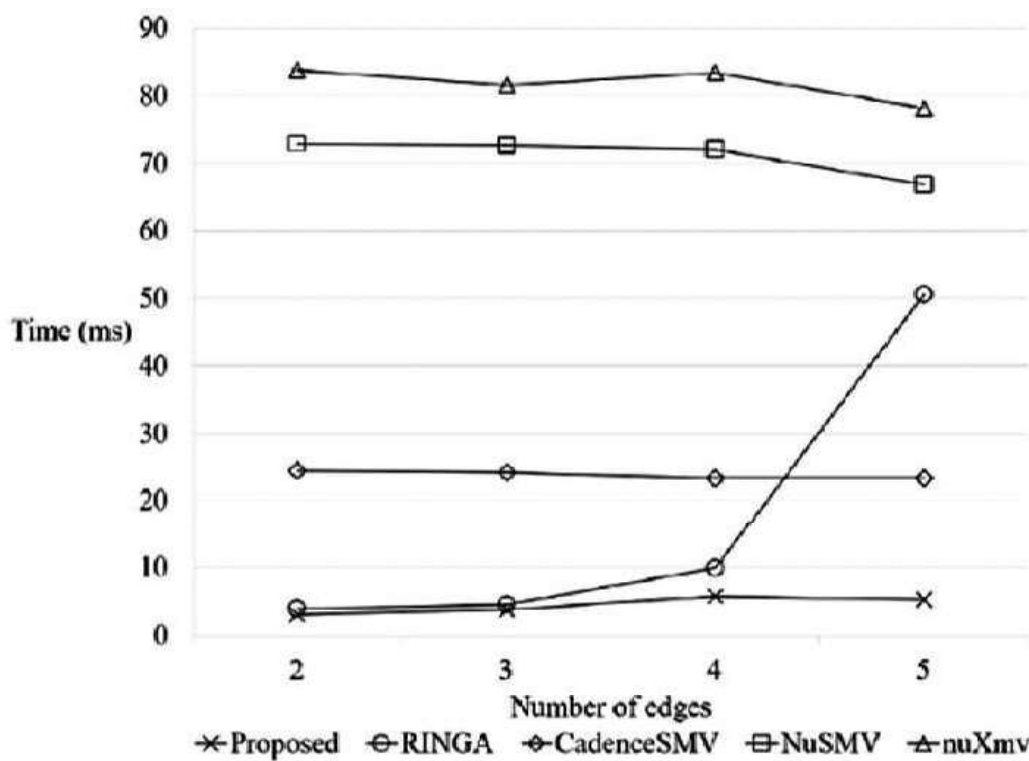


Fig. 9. Result of runtime performance with increasing transitions (with 15 states and three end states).

도면15

