2001 년 한국정보처리학회 춘계 학술발표논문집 제 8 권 제 1 호

STATECHART 로 설계한 Digital Plant Protection System 의 정형 검증+

°김일곤[1], 김진현[1], 남원홍[1], 이나영[2], 곽희환[3], 최진영[1]
고려대학교 컴퓨터학과[1], 서울대학교 원자핵공학과[2], Synopsys,Inc.[3]
{igkim,jhkim,wnam,choi}formal.korea.ac.kr[1],grasia2@snu.ac.kr[2], hkwak@synopsys.com[3]

Formal Verification of Digital Power Plant System Designed by STATECHART

°Il-Gon Kim[1], Jin-Hyun Kim[1], Won-Hong Nam[1], Na-Young Lee[2], Hee Hwan Kwak[3], Jin-Young Choi[1]
Department of Computer Science & Engineering Korea University[1],
Department of Nuclear Engineering Seoul National University[2], Synopsys,Inc.[3]

요 약

원자력 발전소 내장형 시스템과 같이 시스템 오작동으로 인하여 엄청난 재난을 불러 올 수 있는 시스템은 시스템을 구축하기 이전에 완전한 설계 및 검증이 절대적으로 필요하다. 이에 따라 원자력 발전소의 비상 차단 시스템과 같이 고도의 안정성을 요하는 부분에 대해 정형 명세 언어인 STATECHART 를 이용하여 명세하고 테스팅하는 연구가 있어 왔다. 하지만 테스팅 기법만으로는 시스템에서 생길 수 있는 예기치 못한 오류들을 정확히 검출해 낼 수 없다. 그래서 본 논문에서는 시스템의 보다 높은 안전성과 신뢰성을 제공하기 위해 원자력 발전소 비상 차단 시스템인 DPPS(Digital Plant Protection System)를 분석하여 이를 시각적 기반의 설계 명세 언어인 STATECAHRT 를 이용하여 명세함으로써 설계자와 구현자간의 의사 소통을 원활하기 전달함은 물론 모델 체킹 검증 도구인 SMV 로 검증함으로써 실제 원자력 발전소 비상 차단 시스템의 신뢰성과 안전성을 높이고자 한다.

1. 서론

원자력 발전소와 같이 시스템 오작동으로 인하여 환경적인 큰 재난을 불러 올 수 있는 시스템에 대한 완전한 설계 및 검증은 절대적으로 필요한 과제로 국내. 국외에서 원자력 발전소 소프트웨어 기반의 비상 차단 시스템(Emergency shutdown system)에 소프트웨어 공학의 일종인 정형기법[1]을 이용하여 예기치 못한 오류를 검증하고 시스템의 효율성을 증가시키고자 하는 노력이 있어 왔다. 하지만 정형기법을 도입한 설계는 수학적 기호나 전문용어로 기술된 사용자의 요구명세를 검증한 경우로, 전문가의 설명이나 지식이 없이는 이해하기 쉽지 않아 그 효용도가 낮은 편이었다. 이에 대해 설계자와 구현자가 쉽게 이해할 수 있어 원활한 의사소통을 가능하게 해주는 그래픽기반의 정형 명세 언어인 STATECHART 를 이용하여 명세하고 테스팅하는 연구가 진행되고 있지만 이 또한 그 한계점을 가지고 잇다. 단순한 테스팅 기법만으로는 시스템에서 생길 수 있는 예기치 못한 오류들을 정확히 검출해 낼 수 없다. 전문용어로 기술된 사용자의 요구명세를 검증한 경우로, 전문가의 설명이나 지식이 없이는 이해하기 쉽지 않아 그 효용도가 낮은 편이었다. 본 논문에서는 정형기법을 이용하여 원자력

+ 본 연구는 한국 전력 공사의 지원에 의하여 기초전력공학공동연구소의 주관으로 수행되었음

발전소의 실제 비상 차단 시스템인 DPPS(Digital Plant Protection System)를 분석, 시각적 언어인 STATECHART[2]언어로 명세함으로써 STATECHART 가 갖는 효율성을 증대 시킴은 물론 테스팅이 갖는 한계점을 극복하기 위해 , 정형 검증 기법인 모델 체킹[3] 기법을 사용하여 명세된 시스템을 검증함으로써 실제 원자력 발전소 비상 차단 시스템가 반드시 만족시켜야하는 안정성과 신뢰성을 높이고자 한다. 본 논문에서는 i-Logix 에서 만든 STATEMATE[3]을 이용해 원자력 발전소의 DPPS STATECHART 로 명세하고 시뮬레이션 하며, STATECHART 명세를 모델 체커의 일종인 SMV[5]로 바꾸어서 검증한다. 본 논문은 2 장에서는 정형기법에 대해 간략히 소개하며 3 장에서는 원자력 발전소 DPPS(Digital Plant Protection System)을 자연어 명세, STATECHART 를 이용한 명세로 나누어 살펴보고 4 장에서는 정형기법의 일종인 모델체킹의 자동화 도구인 SMV 를 통한 검증에 대해 기술한다. 5 장에서는 결론 및 향후 연구 방향을 제시한다.

2. 정형기법과 모델체킹

정형기법 (Formal Methods)은 컴퓨터 시스템을 개발하기 위해 수학적으로 분석하고 설계하는 기술을 의미한다. 이 정의에서 말하듯이 정형기법은 수학과 논리학에 기반을 둔 방법으로 하드웨어나 소프트웨어 시스템을 명세하고 시스템이 만족해야 할 특성 역시 수학적인 논리로 표현하여 시스템이 특성을 만족하는지를 증명한다. 이러한 정형기법은 수학적 모델을 설명하기 위한 문법, 그리고 문법의 의미를 설명하는 의미론 및 의미 관계를 포함한다. 따라서 정형기법은 자연어가 내포할 수 있는 애매모호함이나 불확실성을 최소한으로 줄일 수 있다. 또한 설계된 시스템이 처음에 의도된 요구사항과 동일한지를 수학적 성질을 이용하여 증명할 수 있기 때문에 개발초기에 큰 실수를 발견할 수 있다. 그러나 정형기법도 다른 많은 기술들과 마찬가지로 적용에 한계가 있다. 이 한계는 현재의 기술 수준에서 발생하기도 하고 이론적인 한계에서도 야기되기도 한다. 현재 사용되는 정형기법에서 실제로 문제가 되는 것은 어떻게 실시간적인 특성을 정확히 표현하고 증명을 할 수 있는가 하는 문제이다. 또한 정형기법의 이론적인 한계는 다음의 두 가지 관점에서 보일 수 있다. 첫째, 실세계와 수학적인 세계의 경계는 무엇인가? 둘째, 수학의 내부적인 한계는 무엇인가? 이러한 질문은 정형기법이 개발되고 발전하면서 끊임없이 제시되고 있다. 수학적인 기반을 갖고 개발되고 있기 때문에 그 한계는 halting 문제와 같은 수학에서의 한계와 거의 비슷하다. 이러한 한계를 잘 이해하는 것은 정형기법을 안전에 민감한 시스템에 적용하는데 매우 중요하다. 정형기법은 크게 정형 명세(Formal Specification)와 정형 검증(Formal Verification)의 두 가지로 구분할 수 있다. 정형 명세는 시스템이 만족해야 할 요구사항과 그 요구사항을 만족할 수 있는 설계를 기술하는 것을 말한다.

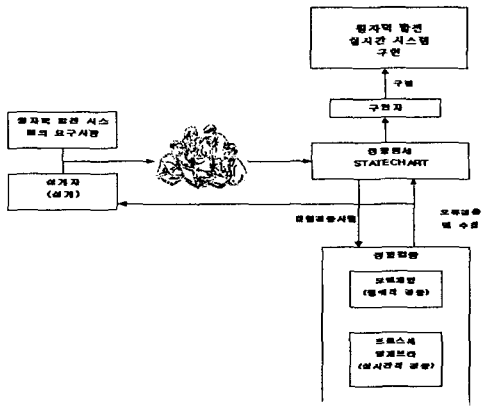

그림 1. 정형기법을 이용한 설계 및 구현

정형 검증은 명세가 정확한지, 즉 설계가 요구사항을 만족하는지를 검사하는 것이다. <그림 1>은 정형기법을 이용하여 시스템을 설계하고 구현하는 과정을 보여주고 있다.

3. DPPS(Digital Plant Protection System)

3.1 DPPS 자연어 명세

DPSS 시스템은 크게 A,B,C,D 4 개의 각각의 독립적인 equipment room 들로 구성되어 있으며, 이 각각의 equipment room 들은 크게 bistable processor, coincidence processor , RT, ESF 로 구성되어 있다. 이 각각의 모듈의 자세한 동작 사항은 다음과 같다.

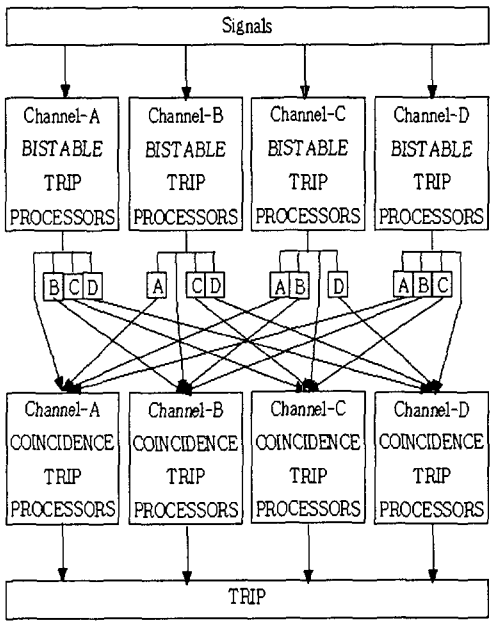

그림 2. DPSS 동작 절차

1. bistable processor

각각의 4 개의 채널에서 모니터링 하는데 필요한 프로세스의 개별적인 측정값을 받아들인다. bistable function 은 측정한 프로세스를 미리 정해진 제한값과 비교하여 trip 상태로 전이할 것인지를 결정한다. 원자로가 start-up 되고 shutdown 될 동안 불필요한 trip 을 발생시킬 수 있는 제한값을 갖는 bistable function 은 불필요한 trip 을 막는 operating bypass 에 의해 처리되게 된다. Operating bypass 는 프로세스가 정해진 값의 범주내에 있으면, 작동자에 의해 수동적으로 동작하지만 프로세스가 정해진 값의 범주를 벗어나면, 자동적으로 제거되게 된다.

2. coincidence processor

각각의 채널에서 coincidence processor 는 bistable function 마다 local coincidence logic(LCL) 알고리즘을 적용한다. LCL 알고리즘은 각각 4 개의 A, B, C, D 채널에서 생기는 출력을 입력받아 두 개 이상이 error 상태에 있는지를 4 번 voting 해서 그래도 error 상태에 있으면 위험한 상태를 알리기 위해 trip 하게 된다. 즉 다시말해서, local coincidence logic 은 다음과 같은 입력을 받는 coincidence signal 을 생성하게 된다. {AB, AC, AD, BC, CD, ABC, ACD, BCD, ABCD}

3. RT, ESF

4 개의 독립된 채널중 2 개 이상의 신호가 4 회 이상 지속될 경우, 제어봉으로 연결된 전원을 차단하여 중력에 의한 제어봉의 자유 낙하에 의해 핵반응을 제어할 수 있도록 하는 RT(Reactor Trip), ESF(Engineered Safety Features)

3.2 DPPS 정형명세

본 논문에서는 이 DPPS 에 대한 STATECHART 의 명세는 3 개의 부분으로 나누어져 있다. 첫째, 입력 시그널로 들어오는 아날로그 변수와 디지털 변수에 대한 이상여부를 확인하는 bistabple trip processors 부분. 둘째, 다른 채널과 brocasting 하면서 실제적으로 시스템에 이상이 있는지 확인해주는 coincidence processors 부분, 마지막으로 RT, ESF 부분은 실제로 어떤 역할을 수행하느것이 아니라 단지 coincidence processor 로 발생한 trip signal 을 받는지 까지만 명세에 포함시켰으며 bypass 또한 시스템의 초기단계에서부터 영향을 미치는 부분으로 DPPS 에 포함시키지 않았다. 또한 이 논문에서 세운 가정은 A, B, C, D 4 개의 채널이 입력시그널을 받아들일 때의 각각의 채널에 대한 지연시간을 50ms 까지만 허용하는 것으로 설정하였다. STATECHART 에서 25ms 는 한 step 으로 처리하였다.

<그림-2>은 DPPS 의 전반적인 동작 절차를 나타내고 있으며 <그림-3>은 이 DPPS 를 STATECHART 를 이용하여 명세한 부분이다.

4. 검증

본 논문에서는 검증코드를 만들기 위해 원래의 STATEMATE 의 STATECHART 의 문법에 얼마의 제약을 가하여 이를 SMV 로 바꾼 후 시스템이 요구하는 사항을 CTL 로 바꾸어 검증하였다.

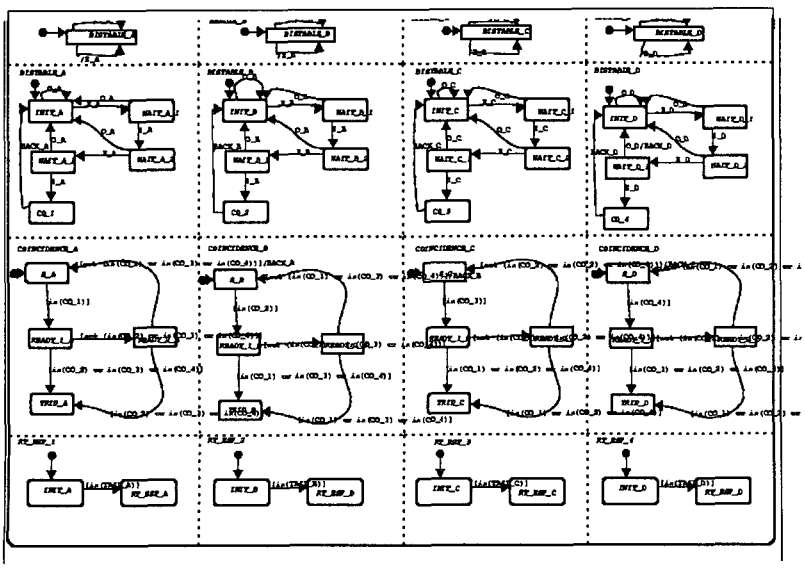

그림 3. STATEMATE 를 이용한 DPPS 명세

본 논문에서 DPPS 가 만족해야 하는 중요한 성질은 다음과 같다.

Property : Safety_01

Always (((*ERROR i* ∧*ERROR i*) → (*ERROR i* ∧*ERROR i*) → (*ERROR i* ∧*ERROR i*) → (*ERROR i* ∧*ERROR i*) → *TRIP*) ($i \neq j$, $0 \leq i,j \leq 3$))

Property Fairness_01:

Always (*NOT_ERROR j* ∧ *NOT_ERROR j* ∧ *NOT_ERROR j*) → *SYSTEM_RESET* ($i \neq j \neq k$, $0 \leq i,j,k \leq 3$)

Property Safety_01 은 만약 2 개 이상의 채널에서 에러가 검출된다면 그 즉시 trip 신호가 발생해야 함을 요구조건으로 둔 것이며 두 번째 Property Fairness_01 은 만약 어떤 채널의 coincidence processor 에서 trip 을 판가름하기 위해 기다리고 있다 하더라고 다른 모든 채널에서 에러가 검출되지 않는다면 bistable coincidence 와 coincidence processor RESET 시키게 하는 Fairness 를 검증한 것이다. 단 이미 각각의 채널이 처음 입력신호를 받을 때 가질 수 있는 시간 지연은 명세상에서 두 step 까지 허용하는 것으로 하였다. 예를 들어 bistable 에서 입력신호를 받을 때, C, D 채널에서는 모두 정상 신호가 들어오고 A 에서는 B 채널에서 보다 에러 신호를 2 step 먼저 입력을 받게 되면 A 채널에 있는 coincidence processor 는 trip 을 발생시키지 않게 된다. <그림-4> 에서 나타난 바와 같이 SMV 로 검증한 결과 위의 요구사항을 만족시킴을 알 수 있다.

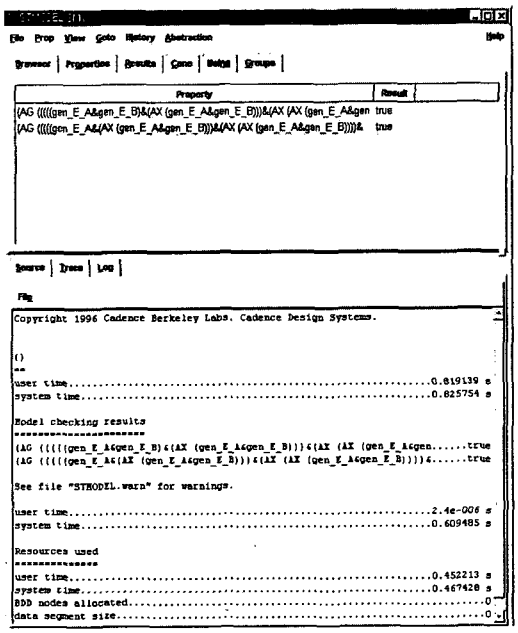

그림 4. SMV 를 이용한 검증 결과

5. 결론 및 향후 연구 방향

본 논문에서는 기존의 원자력 발전소 시스템에 대한 명세에 그치지 않고 정형기법을 이용하여 실제 원자력 발전소의 비상 차단 시스템인 DPPS 를 도식적인 STATECHART 를 이용하여 명세함으로써 설계자와 구현자 모두 보다 이해하기 쉬워 상호간의 의사소통을 원활하게 할 뿐만 아니라 테스팅 기법으로 찾을 수 없는 오류를 모델체커인 SMV 로 DPPS 가 2 개 이상의 채널에서 에러상태를 확인한다면 TRIP 상태로 전이하는지를 검증하여 시스템의 안정성과 신뢰성을 도모하였다. 이 사항은 원자력 발전소 시스템이 그 안전성을 위해 반드시 갖추어야할 필수 사항이다. 하지만 DPPS 를 명세함에 있어 각각의 채널이 입력 신호를 받는 시간 지연을 어느 정도 까지 허용해 줄 것인지에 대해서는 보다 정확한 요구사항을 바탕으로한 명세가 이루어져야 할 것이며, DPPS 를 STATECHART 를 이용하여 보다 상세한 측면, 즉 원자력 발전소를 처음 on 시켰을 때 에러가 아닌 상태를 처리하기 위한 bypass 나 실제로 에러가 발생하여 trip 한 상태에서 보호 역할을 수행하는 RT(Reactor Trip), ESF(Engineered Safety Feacture)의 역할을 분석하여 명세하고 검증하고자 한다.

6. 참고문헌

[1] Edmund M.Clarke. jr, Orna Grumberg, Doron A. Peled. Model Checking, 1999, The MIT press

[2] David Harel, STATECHART: A VISUAL FORMALISM FOR COMPLEX SYSTEMS, Science of Computer Programming 8 (1987) pp231-274

[3] David Harel and Ammon Naamad, The STATEMATE Semantics of STATECHARTs, ACM Trans. Soft. Eng. Method. Oct. 1996

[4] K. L. McMillan. Symbolic model checking - an approach to the state explosion problem. PhD thesis, SCS, Carnegie Mellon University, 1992