열차제어시스템 소프트웨어의 신뢰성 향상에 대한 고찰

김종기, 최규형, 이종우
한국철도기술연구원, 전기연구본부

A Study on the Software Reliability Improvement in Railway Control System

Jongki Kim, Kyoohyung Choi, Jongwoo Lee
Korea Railroad Research Institute, Electric Research & Development Dept.

Abstract - 열차제어시스템의 소프트웨어는 높은 신뢰성과 안전성이 요구된다. 이를 위해 철도에서는 고장회피(Fault Avoidance) 기법과 고장허용(Fault Tolerance)기법을 사용하고 있다. 그러나 최근 신호설비의 소프트웨어 신뢰성을 좀더 향상시키기 위해 일본, 프랑스, 스웨덴 등에서는 안전에 치명적인 시스템에 사용되고 있는 정형기법(Formal Method)을 포함하여 많은 연구를 수행중이다. 본고에서는 국외 관련분야를 분석하고, 열차제어시스템 소프트웨어의 신뢰성 향상에 관하여 기술하고자 한다.

1. 서 론

철도신호기술은 1825년 철도발명 당시 기마수에 의한 적색신호기를 사용한 이래, 산업기술혁신 성과를 적극 활용하여 자동신호화, 계전화를 시작으로 CTC, ATS, ATC, 건널목보안장치 등 질적, 양적으로 진전을 거듭했다. 우리나라의 신호기술도 1899년 9월 18일 노량진-인천간에 최초로 철도가 부설되고 완목신호기 설치를 시작으로 비약적인 발전을 거듭하여, 고속철도의 도입과 함께 선진외국의 신호제어기술 이전을 통해 자동폐색장치, 계전연동장치, 전자연동장치, CTC장치, ATS장치 등의 신호보안시스템을 국산화하는 등 산학연이 합심하여 계속 연구개발 중에 있다.

열차의 원활한 운행 및 승객의 안전과 밀접한 관련을 가지고 있는 신호보안시스템은 종래부터 신뢰성 향상에 적극적이었다. 철도 초기에는 신호보안시스템의 신뢰성 향상을 위해 2가지 방향으로 시스템의 신뢰성을 높이고자 했다. 첫째는 고신뢰도가 증명된 구성품을 사용하는 것이고 둘째는 여러 형태의 이중화 구성을 통하여 시스템의 신뢰도를 높였다.

그후 컴퓨터 기술이 철도에 도입되면서 정보통신분야의 온라인 및 실시간 시스템이 철도에 적용되기 시작했다. 이러한 컴퓨터화된 제어시스템의 신뢰성 문제는 새로운 이슈로 떠오르기 시작했다. 우리나라의 열차제어시스템도 점차 자동화, 첨단화됨에 따라 필연적으로 하드웨어, 소프트웨어의 안전성 보장이 요구되고, 제품의 설계단계부터 폐기단계까지 전 과정에 걸쳐 신뢰성 분석이 요구되고 있다.

현재 컴퓨터화된 제어시스템의 근간을 이루고 있는 소프트웨어 신뢰성 기술은 기존의 하드웨어 신뢰성 기술만큼 정립되어 있지 않으며, 철도선진국에서 이 분야에 대한 연구가 기존의 축적된 데이터와 연구성과를 바탕으로 진척되고 있으며 연구결과도 조금씩 나오고 있다.

따라서 지금도 가속되고 있는 정보화시대에는 첨단산업의 집적화, 지능화, 무인화 등 요소기술과 발맞추어 Smart-Rail, Cyber Rail, 무인운전 및 무선을 이용한 열차제어장치 등 새로운 신기술의 상용화에 따른 열차제어시스템의 발전방향을 예견할 때 소프트웨어 신뢰성 향상을 통한 철도시스템의 신뢰성 향상은 요긴하다.

2. 본 론

2.1 열차제어시스템 소프트웨어 신뢰성 기술

컴퓨터로 제어되는 열차제어시스템의 소프트웨어는 고안전과 고기능이 요구되는데 이를 위해 주로 고장회피와 고장허용 기법 두 가지 기법을 사용해 왔다.

소프트웨어의 고장회피기법은 소프트웨어의 품질을 개선하기 위해 버그를 제거하는 것으로 프로그램 버그를 제거하는 전통적인 여러 기법들이 있다.

고장허용기법은 주로 복수의 프로그램을 사용하여 에러를 처리하는 방법을 사용한다. 에러 수정 방법으로 Majority Logic(Masking Technique)과 전환기법(Switching Technique)이 있다. 또 고장허용기법으로 N-버전 프로그래밍과 Recovery Block이 있다.

철도신호시스템의 소프트웨어는 다음의 기법들이 사용된다.

(a) 확인된 전통적 제어 알고리즘의 사용
(b) 안전이 요구기능의 명확한 분리
(c) 이론적 다중과업처리 금지 및 단일처리기법 채택
(d) 인터럽트 핸들링의 제한
(e) 확인된 프로그래밍 언어의 채택
(f) 프로그램상에서 안전측 정보와 위험측 정보 구별

고장허용기법의 기본적인 원리는 소프트웨어 버그나 하드웨어 고장이 발생할 때 출력값을 안전 차원에서 고정하도록 기능을 제한하는 것이다. 이것은 열차운행정지가 가장 안전하다는 '철도철학'과도 부합된다.

입력에러나 부적절한 제어에서도 안전하지 않은 출력을 피하려면, 에러의 감지는 매우 중요하며 철도신호시스템에서는 다음의 진단기능을 사용하고 있다.

(a) 여러 CPU간의 작동비교
(b) 전용 하드웨어를 갖는 CPU의 모니터링
(c) 하드웨어를 감시하는 과정에서 발생하는 에러를 모니터링(Comparator와 인터럽트 회로의 작동을 확인하는 것을 포함한다)
(d) 잠재적인 하드웨어 에러의 현실화(입력회로의 체크와 RAM의 Read/Write의 대조)

이런 진단기능들이 하드웨어 결함과 에러를 감안한 안전 테스트를 통해 확인된다.

안전에 치명적인 철도시스템에서 신뢰성을 언급하게 되는 이유 중 하나로 사고를 예방하고 승객의 안전을 보장하는 안전성과 밀접한 관련이 있기 때문이다. 신뢰성 기술과 안전성 기술이라는 이 유사 기술은 다양한 입장이 오래 전부터 존재해 왔다. 항공분야에서는 사소한 고장이 곧 사고와 직결되므로 신뢰성과 안전성을 동일하게 취급했으나 철도신호분야에서는 신뢰성과 안전성에 대한 공학적인 접근이 진전됨에 따라 이 두 개념을 구별하는 사고방식이 일반화되어가고 있다. 그러나 양쪽의 접근철학은 상이하나 기술은 서로 중복되는 바가 많고 평가방법, 분석기법도 유사하다.

소프트웨어 신뢰성에 대한 여러 가지 정의가 있으나

'어떤 소프트웨어가 주어진 환경하에서 일정한 기간동안 주어진 명세서 대로 정상 가동될 확률'로 정의할 수 있다. 소프트웨어의 실행결과가 요구사항과 다를 때는 오류 또는 고장이 발생한 것으로 간주한다. 소프트웨어의 정량적인 신뢰도 측정에 관한 것으로 통계적인 관점에서 접근했던 Jelinski-Moranda 모델, Non-Homogeneous Poisson Process 모델 등을 비롯하여 많은 신뢰도 모델이 있으나 CENELEC, 일본의 열차보안제어시스템의 안전성기술지침 등을 참고할 때 철도분야에서 이들 모델이나 이와 유사한 형태로 소프트웨어의 신뢰성 측정 및 향상을 도모하는 것 같지 않아 보인다.

그래서 소프트웨어 신뢰성은 정량적인 수치에 근거하기보다는 신뢰성을 높이는 기술을 시스템에 구현하는 방식으로 진행되어 왔다. 소프트웨어에 의해 구동되는 하드웨어와 함께 신뢰성을 고려하여 시스템을 다중계로 구성하거나 전송회로에 우회로를 부가하는 등의 방식으로 신뢰성 향상책이 취해져 왔다.

그러나 최근에는 철도시스템에 소프트웨어 사용이 일반화되면서 소프트웨어의 신뢰성도 수명주기를 통하여 CENELEC 규격을 비롯한 유럽 각국의 규격에서 신뢰성과 안전성을 보증하는 체계를 확립해가고 있다.

2.2 열차제어시스템 소프트웨어 신뢰성 향상을 위한 활동

2.2.1 IEC와 CENELEC

IEC는 일반 산업분야의 전자기기에 대한 안전성 규격인 IEC61508을 제정했다. 이 규격에는 안전 수명주기와 안전 무결성 수준(Safety Integrity Level: SIL)이 포함되어 있다. 유럽의 CENELEC와 일본의 안전성기술지침은 IEC61508에 기초하고 있다.

CENELEC(European Committee for Electrotechnical Standardization: 전자기술 표준화를 위한 유럽위원회)은 유럽에서 철도신호시스템의 과학기술적 요건을 반영하고 법적 구속력을 가지며 유럽내 철도의 상호운용성을 높이는 규격을 제정했다. 이중 EN50128은 철도신호보안시스템의 소프트웨어에 대한 신뢰성과 안전성에 대한 기준으로 SIL을 정의하여 소프트웨어의 안전성을 분류하고 있으며 SIL의 구체화, 관련요인의 책임, 수명주기, 문서화, 요구사항, 구조, 설계 및 구현, 증명 및 시험, 소프트웨어의 결합, 검증, 평가, 품질보증, 유지보수 등의 절차를 제시하고 있다.

2.2.2 일본

일본은 철도보안제어시스템에 대한 안전성기술위원회를 조직하여 2년 동안 토의한 끝에 1996년에 열차보안제어시스템의 안전성기술지침을 발표하였다. 이 지침은 금세기에 축적된 경험의 토대 위에 철도신호의 안전기술과 경영을 위한 요구사항들을 통합했고 CENELEC과는 달리 법적 구속력은 없다. 철도제어시스템의 수명주기를 통해 안전 경영과 기술적 활동에 대한 요구사항을 규정하고 있다.

여기서도 SIL의 개념을 사용하고 있으며 소프트웨어에 대해서는 수명주기의 초기단계에서 신뢰성과 안전성은 개발 조직과 부여된 SIL에 따라 달라진다.

소프트웨어에 대한 전통적인 방법은 최종단계의 테스트를 중시하는 경향이 있으나 이 지침에서는 시스템의 수명주기 내내 위험요소를 분석하고 관리한다.

예를 들어 이 지침에 따르면 작동중인 시스템을 교체하기 위해서는 다음의 절차를 거쳐야 한다.

(a) 소프트웨어 변경에 따른 효과 분석
(b) 프로그램에서 변경된 부분의 검증
(c) 변경으로 영향을 받는 부분의 재검증
(d) 전체 소프트웨어의 검사

다음 표는 일본 철도분야의 소프트웨어 신뢰성 기술을 정리한 것이다.

〈표 1〉 소프트웨어 신뢰성, 안전성 기술

1. OS (a) Interrupt - Interrupt 금지 및 초기치 설정
2. 공통서브루틴, 펌웨어 (a) 공통서브루틴 - 위상차 동기식 공통처리(검증된 모듈)
3. 응용분야 (a) 시스템 - 현 시스템의 알고리즘 활용 - 현장기기 제어불능시 제어취소 방법 - 전송 두절 시 안전성 처리 - 다중 컴퓨터의 Fail-Soft화 (b) 프로그램 - 프로그램 구조의 단순화 - goto 문 사용 금지 - 안전성 기능의 분리 - 안전/불안전 상태정보 할당 통일
4. 입출력 인터페이스 (a) 입출력 - 입력조건의 조합 사용 - 과도적인 불안정 감소를 위한 수회의 일치 - 입력데이터의 검사(변화, 평균치, 범위) - 정정이 곤란한 위험측 출력의 제어방법 - 입력회로의 하드웨어 고장진단 - 제어출력의 피드백 검사 (b) 전송 - 시리얼 데이터 전송의 안전성 대책 (c) CPU - CPU간 전송검사 (d) Man-machine - CRT 디스플레이 표시정보의 정확성 보정 - 오조작의 거부 - Man-machine 기기의 오조작 방지기능 - 운전대 조건 입력의 정합성 검사 - 오조작 방지를 위한 취급지침
5. 진단프로그램 (a) 프로그램 처리 - 다수결 처리 - 프로그램 폭주 검지 - 프로그램의 비대칭성 구조 - 비대칭 다중 승인 시험 - 데이터의 더블 링크 - 메모리 보호
6. 진단프로그램 (a) 리커버리 - 2중계 절체시의 처리, 연속성 확보 - 이상 검출시의 처리, 복귀의 지표 - LAN에서 수신데이터 오류시의 처리 - Point 부절환시의 재제어 - 현장기기 제어불능시 제어취소 방법 - 시스템 고장의 고장진단 - 설계와 검사의 독립 - 계층화 시스템의 공통기능 검사

2.2.3 프랑스의 활동사례

프랑스 지하철의 자동 열차운영 시스템에 정형기법(Formal Method)을 사용하여 소프트웨어 신뢰성을 향상시켜 안전성을 확보한 'Météor 프로젝트'가 있다. 파리 지하철의 남동선과 동남선은 러시아워 때 시간당, 노선 한 방향별로 평균 4만명의 승객을 수송한다. 이 노선에 MTI(Matra Transport International)에서 개발한 자동 열차운영시스템을 채택하여 자동 무인운전열차와 수동 운전열차를 병행하여 사용하고 있다. 여기는 높은 안전성과 고품질의 서비스를 비롯하여 고도의 신뢰

성을 가진 소프트웨어가 요구된다.

MTI는 시스템과 장비의 각 단계별로 수행되는 안전성 분석에 의거하여 자동 열차운영시스템의 소프트웨어를 안전에 치명적인 소프트웨어와 안전에 치명적이지 않은 소프트웨어 두 가지로 구분하고 소프트웨어 결함을 다시 다음 두 가지로 분류했다.

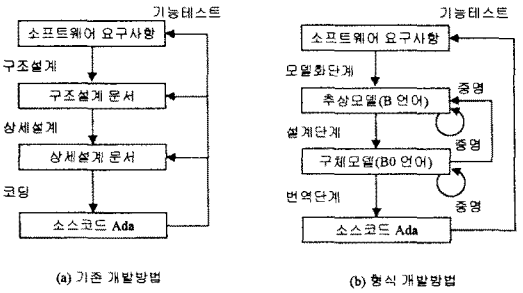
- 설계 또는 코딩에러: 소스코드가 소프트웨어 요구사항을 만족하지 않는 경우
- 코드생산과정(컴파일, 링크)이나 하드웨어 결함으로 인한 에러

MTI는 안전에 치명적인 소프트웨어를 개발할 때 다음 두 가지의 방법을 채택했다.

- 단일 소프트웨어 설계와 개발단계에서 B 정형기법을 사용하여 요구사항에 따라 버그가 없는 소프트웨어의 정확성을 수학적으로 증명한다.
- VCP(Vital Coded Processor Technique)를 사용하여 코드생산과정과 하드웨어결함에 따른 에러를 검출한다.

VCP는 1980년대 MTI가 개발한 방법으로 중복으로 코드화된 데이터 요소를 사용하여 각 안전성 주기를 실행한 후에 코드의 정확성을 체크해가며 소프트웨어 실행의 안전성을 확인하는 확률적인 접근방법이며 기대 총안전수준을 보증한다.

소프트웨어의 명세화부터 설계, 개발에 이르기까지 B 정형기법을 사용했다. 또 각기 다른 컨트롤 단위와 하드웨어 결함을 방지하고자 VCP를 사용했다. B 정형기법과 VCP는 서로 보완관계에 있으며 이를 적절하게 사용하여 높은 신뢰성을 보증하게 되었으며 이 방법이 기존의 신뢰성 활동을 자연스럽게 대신하게 되었다. 여기서 사용된 B 정형기법은 J. R. Abrial이 연구한 것으로 다음 그림에 기존의 개발 방법과 정형개발방법의 절차가 나와있다.



<그림 1> 기존 개발방법과 정형 개발방법

MTI의 정형개발방법은 다음 3단계로 구성되어 있다.

- 모델화 설계: 소프트웨어 요구사항을 B 언어로 명백하게 정형화한다. 이렇게 해서 구축된 추상모델은 필요한 모든 속성과 기능을 표현해야 한다.
- 설계단계: 구축된 추상모델을 구체화한다. 추상적인 표현을 구체적인 구현으로 변환한다. 알고리즘과 데이터 구조의 선정, 추상적인 표현의 삭제 등. B 언어의 일부인 B0로 구체모델을 완성할 때까지 변환을 계속한다. 이 과정의 마지막 단계는 '구현(Implementation)'인데 전통적인 프로그래밍 언어로 쉽게 변환될 정도로 컴퓨터 프로그램에 아주 가까운 형태로 모델을 표현한다.
- 번역단계: Ada 소스코드로 변환한다.

위의 단계에서 수학적 증명을 수행한다. 증명은 수작업으로 하거나 증명 생성기(Proof Generator)와 정리 증명기(Lemma Demonstrator) 등의 툴을 사용한다.

그리고나서 소프트웨어의 검증활동을 검증팀에서 수행한다. 이것은 개발팀에서 수행하는 검증과는 별도로 수행된다. 검증절차에는 안전에 치명적인 기능을 충분히 검증할 수 있도록 모든 활동이 포함된다. 여기서 수행되는 절차는 구체 B 모델의 검정, B 인터페이스의 검증, 추가 증명법칙의 검증, Ada 소스변환 검증, 검증에 근거한 기능테스트 등이 있다.

호스트 컴퓨터의 기능 검증 결과 버그가 발견되지 않았고 타겟 컴퓨터에 대한 통합 검증, 현지 테스트, 현장 작동시에도 모두 버그가 발견되지 않았다. 이것은 기존의 방법에 비해 매우 향상된 결과였다.

3. 결 론

철도시스템에 정형기법을 적용하는 것은 현재 연구 중에 있다. 프랑스는 ATC에 적용했고 스웨덴은 전자연동장치의 데이터를 검증하는데 정형기법을 사용하여 성공을 거두고 있다.

철도신호 소프트웨어의 개발에 이 방법을 적용하는 목적은 소프트웨어 사양(Specification)의 타당성을 보증함으로써 안전 소프트웨어의 현실화와 검증 및 테스트에 드는 인력(Manpower)을 줄이고자 하는 것이다. 정형기법은 사양의 모호성을 제거시켜 고신뢰성이 필수적인 다른 시스템에도 적용될 수 있다. 이 방법은 소프트웨어의 개발에 드는 총 인력을 효과적으로 감소시키는 것으로 보고되었다.

정형기법들 중에서 VDM(Vienna Development Method)는 필요한 사양에서 소스 프로그램까지 다루고 있으며 소프트웨어 개발까지 다음 3단계를 거친다.

(1) 정형사양 언어인 VDM-SL의 정형사양의 준비
본 단계에서는 Object 시스템을 모델링하고, 상태변수와 작동함수를 갖는 시스템을 표현하고, 제약조건을 정의하고, 항상 불변조건으로 만족되어야만 하는 사전조건과 사후조건을 결정하여 작동함수가 불변조건을 만족시키도록 한다.

(2) 준비된 사양에 대한 검증
본 단계에서는 불변조건이 항상 만족하는 지를 증명한다. 사전조건과 사후조건이 나타날 때, 상태변수의 초기상태가 불변조건을 만족하고 작동함수가 불변조건을 만족하는지를 검증한다. 여기서는 사양에 에러가 없는지를 검증한다.

(3) 준비된 사양을 실제 모델로 변경하고 최종 프로그램으로 발표
본 단계에서는 무결성 수준을 유지하면서 검증된 사양을 최종 코드로 변경한다.

본 논문에서는 국내 열차제어시스템의 발전과정과 열차제어용 소프트웨어와의 관계를 기술한 후 철도선진외국에서 추진중인 열차제어시스템 소프트웨어 신뢰성 기술 활용 동향을 조사분석한 후 그 발전 동향을 서술하였다. 우리나라에서도 철도선진국의 신뢰성 기술의 활용사례와 정형기법의 소개를 통해 열차제어분야에서 많은 향상이 이루어질 수 있도록 산학연관의 지속적인 연구가 추진될 것으로 기대된다.

[참 고 문 헌]

[1] 우치수 외 2명, "소프트웨어 신뢰도 측정에 관한 연구," 한국정보과학회논문지, Vol. 15, No. 3, pp. 182-193, 1988.

[2] Faivre, Alain and Benoit, Paul, "Safety Critical Software of Météor Developed with the B Formal Method and the Vital Coded Processor," The 4th WCRR Conference, Tokyo, 1999.

[3] Hirao, Yuji and Fukuda, Mitsuyoshi, "Software Safety Technologies for Railway Signalling," Japanese Railway Engineering No. 141, pp. 12-14 , 1998.