STATEMATE를 이용한 AWES(Auto Warning/Ejection System) 명세

장성호° 최진영
고려대학교 컴퓨터학과
{shjang, choi}@formal.korea.ac.kr

Specification of AWES(Auto Warning/Ejection System) with STATEMATE

Sung-Ho Jang° Jin-Young Choi
Dept. of Computer Science, Korea University

요 약

공군에서 전투기 사고는 적은 비율이지만 계속적으로 발생한다. 이 시스템의 명세 목적은 사람이 결심하여 전투기에서 비상탈출 하는 것에 컴퓨터의 역할를 추가하여 조종사의 생존 가능성를 높이는데 있다. 사람은 모든 감각기관를 통하여 미래의 상황를 예견할 수 있는 능력이 있는 반면, 착각 등를 통하여 실수를 포함하는 행위를 할 수도 있고, 비상탈출을 결심하였다고 하더라도 전투기 기동이 매우 급변하여 반드시 비상탈출을 성공한다고 볼 수 없다. 그래서, 기존에 장치되어 있는 측정장치들을 이용하여, 비 정상적인 위치에 전투기가 위치했을 때는 Warning를 하고, 명백하게 비상탈출를 하여야할 경우가 발생했을 때는 자동으로 비상탈출를 실행하는 System를 명세하였다. AWES(Auto Warning/Ejection System)은 Safety-Critical System의 일종이라 할 수 있다. 그래서, 개발 초기 단계부터 정형기법(formal methods)에 기반하여 개발되어야 한다. 본 논문에서는 Reactive system의 행위적인 면를 명세하는데 장점를 가지는 Statecharts를 이용하였으며, STATEMATE라는 도구로 AWES(Auto Warning/Ejection System)를 명세하였다.

1. 서 론

시스템 개발 절차에서 요구조건 분석, 명세, 디자인, 시뮬레이션, 구현, 테스팅, 유지보수등에 대하여 주요 관심을 가지며 발전해왔다. 개발 절차의 후반부에서 오류 또는 잘못된 생각등을 교정하는 것은 개발 전반부에서 교정하는 것 보다 훨씬 많은 시간적 재화적 낭비를 발생시킨다. 또한, 최근 software는 빠른 속도로 거대화되고, 병렬적이고 분산된 시스템으로 구성됨에 따라 시스템이 처리해야할 정보의 복잡도와 규모는 훨씬 높아지고 있다. 그래서, 시스템은 쉽게 오류를 발생하게되고 예상치 못했던 행위를 하는 경우가 발생된다. 결국, 시스템 개발 절차의 초기 단계부터 정형기법(formal methods)에 기반하여 시스템을 개발하는 방법이 절실히 요구되고 있다.

정형기법은 크게 정형 명세(formal specification)와 정형 검증(formal verification)으로 나눌 수 있다. 정형 명세는 수학적인 기반으로 요구사항을 기술하는 것이며, 정형 검증은 명세에 대하여 모든 경우에 대하여 요구 조건에 대한 결과가 도출되는지 검증하는 것이다. [1]

군 전력에서 전투기가 차지하는 중요성은 따로 말하지 않아도 충분할 것이다. 그러나, 전투기를 운용함에 있어서 비 전시 상황에서 치명적인 사고는 적은 비율이지만 꾸준히 발생하고 있다.

사고로 인하여 고가의 전투기 손실 뿐 아니라, 몇 년을 걸쳐 양성된 인력의 손실은 더욱 크다 할 수 있다. 또한, 조종사의 죽음으로 인하여 사고의 원인 규명이 불분명해져 추가적인 사고의 잠재요소를 내포하고 있다고 볼 수 있다.

AWES(Auto Warning/Ejection System)은 비상탈출상황이 발생하였을 때 반드시 작동하고, 그 외의 경우에는 반드시 작동하지 않아야 하며, 조종사의 생명과 직접적으로 연관이 되는 Safety-Critical System[2] 의 일종으로 볼 수 있다. 그래서, 개발 단계부터 반드시 정형기법에 기반하여 개발하여야 한다.

여기에서는 이 시스템에 대하여 Statecharts를 이용하여 명세하였다. David Harel은 Statecharts는 State-diagrams + Depth + Orthogonality + Broadcast-communication로 간략하게 설명하였다. 또한, 시각적으로 명세하기 때문에 이해가 쉽고, Reactive system의 행위적인 부분을 명세하는데 뛰어난 장점을 가진다.[3][4]

여기에서 사용한 명세도구는 STATEMATE 이다. 이 도구를 사용하여 모델에 대하여 명세할 수 있으며, Simulation을 통하여 요구조건을 만족하는지 확인할 수 있다.

2. 본 론

전투기가 Stall(항공기 실속)이 발생하였을 경우 전투기 종류에 따라 다르기는 하지만 지상에서 일정고도 이하에서는 비상탈출을 실행하라고 되어있다. 또한, 비행장에서 이륙을 할 때에는 항상 정해진 높이까지는 계속적인 상승을 하게 되어 있다.

위의 2가지 상황을 기준하여 Stall(항공기 실속)상황이 발생했을 때 사람이 항상 비상탈출을 성공한다고 말하기 어렵다. 먼저 위의 두가지 경우 Stall(항공기 실속) 상태이기 때문에 전투기의 추락이 매우 가속화되는 경우이고, 전투기의 상태 또한 매우 급변하는 상황이 된다. 조종사는 분명 정상적인 상태로 비행기를 회복하기 위해 회복절차를 실행하면서 비행기를 정상적인 상태로

만들려고 노력할 것이다. 급변하는 전투기 속에서 짧은 시간 사이에 고도 및 자세를 확인하면서 비상탈출에 성공하는 것은 높은 판단력과 민첩성을 요구할 것이다.

다른 경우로 바다와 같은 곳에서 야간비행을 실시할 경우 조종사는 비행착각으로 인하여 하늘과 땅을 반대로 생각하는 경우가 발생한다.

그래서, 반드시 비상탈출을 하여야 할 상황에 대하여 사람이 비상탈출을 실행 못하였을 경우 자동으로 비상탈출을 실행하고, 비상탈출 상황은 아니지만 비정상적인 위치 및 고도에 있을 경우 Warning을 하는 장치를 AWES(Auto Warning/Ejection System) 이라고 명칭하고 명세하였다.

2.1 AWES 의 자연어 요구사항

이 시스템은 비상탈출과 Warning의 두 가지 역할을 수행하여 조종사의 생존능력을 높인다. 외부에서 들어오는 3가지 Indicator(고도계, Stall 감지 장치, INS(관성항법장치))를 이용하여 비상탈출 또는 Warning 상태를 파악하여 상황에 맞게 처리한다.

2.1.1 비상탈출

비상탈출이 결정되면, 조종사의 생명을 보호하기 위하여 자동으로 안전벨트에 모터를 작동시켜서 조종사의 몸을 좌석으로 밀착시킨 후 비상탈출이 실행되고, 현재의 위치정보와 비상탈출 되었다는 정보를 관제센터에 보고하게 된다.

· Take_off(이륙) mode

비행기는 운항의 특성상 일정 이륙 안전고도 도달까지는 계속적으로 상승 비행을 한다. 그러나, 정해진 최소 이륙 안전고도에 도달하기 전에 Stall이 발생하고, 300ft 이상 연속적으로 하강을 하게 되면 자동 비상탈출을 실행한다.

· Cruise(순항) mode

입력된 최소 이륙 안전고도 이상으로 상승하게 되면 자동으로 Cruise mode로 전환되며, 비행장 영역과 저공비행 구역으로 정해지지 않은 곳에서 Stall(항공기 실속)인 상태이고, 입력된 최소 순항 안전고도 이하로 떨어지면 자동으로 비상탈출을 실행한다.

2.1.2 Warning

조종사에게 발생하는 사고 중에서 비행착각으로 인하여 발생하는 사고를 방지한다. 비상탈출과 같은 조건에서 Stall(항공기 실속)이 아닌 경우로, 경고음과 lamp를 작동시켜서 조종사에게 주의를 준다. 경고음과 lamp는 작전 수행을 위하여 저고도로 비행을 실행 할 수 있으므로 시스템 작동 도중 경고음과 lamp를 ON/OFF 시킬 수 있어야 한다.

2.2 STATEMATE로 구현된 AWES

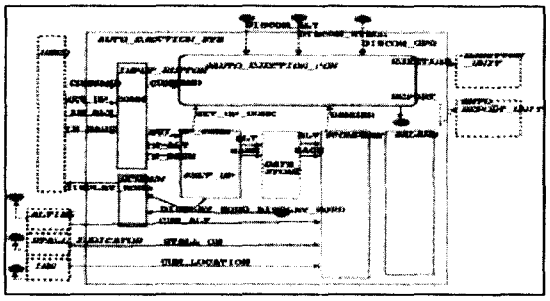
위의 자연어 요구조건을 충족하도록 명세된 AWES은 총 16 개의 Charts로(Activity-charts : 4, Statecharts : 12) 구성되었다.

그 중 상위의 2가지 Charts에 대하여 간략히 알아보겠다.

[그림 1]은 가장 상위의 Activity-charts 이고, Auto_ejection_con Statecharts가 주위의 Activity-charts들을 제어하고 있음을 나타내고 있고, 그 중 중요한 2가지 Activity의 내용에 대하여 알아본다.

Set_up Activity는 Indicator의 입력 수치 값과 비교할 최소 이륙 안전고도, 최소 순항 안전고도, 저공비행 가능 장소(기지, 저공비행훈련구역)등의 비교 값을 입력받아 저장장치에 저장을 한다.

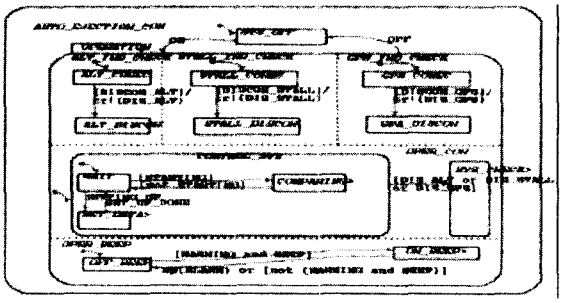
Compare Activity는 Indicator의 입력 값과 저장장치에 입력된 데이터 값을 비교하여 비상탈출 또는Warning 상황인지를 판단한다.



[그림 1] Auto_ejection_sys(Activity-charts)

[그림 2]는 Auto_ejection_sys Charts를 제어하는 부분으로 외부의 Indicator들이 연결이 되어 있지 않으면 자동으로 동작을 멈추고 Error 메시지를 출력하게 하였다. 또한, Warning을 하는 경고음 및 lamp를 제어하는 Oper_beep State는 작동 중 개별적으로 제어할 수 있도록 명세하였다.

주위의 Activity들의 제어는 State에 Static Reaction으로 표현하였고 State name 옆에 '>' 모양이 나타난다.



[그림 2] Auto_ejection_con(Statecharts)

2.3 Prototype Simulation 과 C Code 생성 및 실행

Trailblazer는 STATEMATE에서 제공하는 Simulation 도구이다. Simulation을 통하여 명세가 원하는 결과가 나오는지 확인할 수 있다. Sharpshooter는 STATEMATE에서 제공하는 Code 생성도구이다. 이 도구는 Statecharts를 곧바로 기계에 적용하도록 되어있다.

생성된 코드의 실행과 Simulation의 차이는 Simulated time을 사용하지 않고 Real time을 사용하는 것에 있고, 명세된 시스템에 대한 분석보다는 실행(Performance)에 중점을 두는 것에 있다.

외부의 입력은 여러 가지 방법이 있는데 여기에서는 User가 입력하는 것에 대하여서 Panel Graph를 이용하였고, Indicator 3가지는 Testbench로 구성하였다.

2.3.1 Testbench

Testbench는 Statecharts로 작성되며, 사용자 입력등을 모델화하여 만들거나, 외부입력을 모델화 하여 작성하며, 명세한 시스템의 Driver/Observer등의 용도로 사용한다.

3가지 경우에 대하여 Testbench를 Statecharts로 모델화하였으며 내용은 다음과 같다.

· Case 1 : 정상적인 비행

사용자가 Panel Graph에서 나타난 입출력 Interface를 통하여 Set_up Activity에 연관된 행위를 할 수 있고, 이륙을 시작하면 정상적으로 비행 후 귀환한다.

· Case 2 : 이륙시 비상탈출

최소 이륙 안전고도등 기준이 되는 데이터 값들을 자동으로 할당 시키고, 이륙도중 Stall(항공기 실속)이 발생하면서 추락한다.

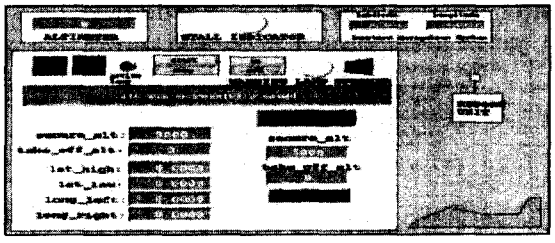
· Case 3 : 순항시 Warning

순항도중 Stall(항공기 실속)은 아니면서 최소 순항 안전고도보다 낮아져서 Warning 상태가 되었다가, 안전한 고도로 복귀한 후 정상적인 고도로 비행장으로 귀환한다.

2.3.2 Panel Graph

Panel Graph는 AWES에 사용되는 Event, Condition, State 등의 정보와 연결되어 작동을 하며, 사용자와 명세한 시스템과의 Interface를 제공하고, 명세한 시스템의 Prototype을 나타낸다..

[그림 3]은 Panel Graph를 추가하여 정상적인 비행부분을 Simulation하는 것으로 User가 데이터를 Set_up하는 과정이다.



[그림 3] Panel Graph : 최소 순항 안전고도가 저장됨.

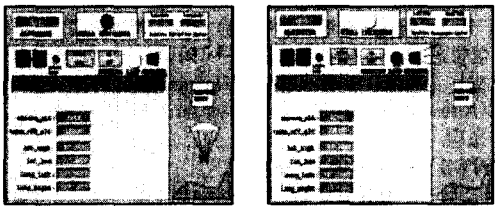
2.3.3 Prototype Simulation 및 실행

Trailblazer는 명세하는 동안 자신의 명세가 올바르게 나오는지 명세도중에 사용할 수 있을 뿐 아니라, 연관된 차트들을 같이 통합하여 Simulation 할 수 있다. 또한, 외부에서 들어오는 행위들은 Testbench로 모델화하여 명세한 시스템과 같이 Simulation을 할 수 있으며, Panel Graph를 사용하여 외부에서 발생할 수 있는 Event등을 같이 Simulation 할 수 있다.

AWES에 Panel Graph를 추가하고, 위에서 제시된 3가지 Case 별로 Simulation을 하였으며 결과는 요구조건을 만족하였다.

C Code를 위와 같은 조건으로 생성하여 실행한 결과 AWES은 Simulation 한 것과 같이 정상적으로 작동하는 결과가 나왔다. 또한, 시간에 관련된 행위(lamp 및 경고음 발생)는 실제 시간으로 동작하였으며, Testbench에 대한 처리가 Simulation 보다 빠르게 작동하여서 Testbench에서 고도변화의 입력단위를 100ft에서 15ft로 적은 단위로 입력하여 실행하였다.

[그림 4]의 좌측은 비상탈출 상황이 발생했을 때 Panel Graph에 나타난 Simulation 및 C Code 실행 결과이고, [그림 4]의 우측은 Warning 상황이 발생했을 때 Panel Graph에 나타난 Simulation 및 C Code 실행 결과이다.



[그림 4] Simulation 및 C Code 실행

3. 결론 및 연구 계획

전투기 조종사의 생존성을 높이기 위해서 AWES(Auto Warning/Ejection System)을 명세하였다. 이 시스템은 STATEMATE를 이용하여 명세 하였는데, 특히 시스템의 행위적인 측면에 대하여 잘 표현할 수 있었다. 또한, 데이터 처리방법 등에 대하여 자세히 명세할 수 있어서 구현자가 시스템을 구현하는데 근접한 명세를 할 수 있었고, Simulation을 통하여 명세된 모델에 대하여 요구조건들을 충족하는지 확인할 수 있었으며다. Code 생성 및 실행을 통하여는 Real time에서 제대로 동작을 하는지 알 수 있었다.

하지만, 명세의 Simulation만으로 이 모델이 정말로 정확한 명세라고 말하기에는 부족하다. 특히 이 시스템은 생명을 대상으로 하는 Safty-Critical System의 일종으로 반드시 발생할 상황에 대하여 동작하고 그 외의 상황에서는 발생하지 않아야 한다. 그래서, 명세의 신뢰도를 높이기 위해서는 반드시 검증의 단계(Verification)를 거쳐야 한다.

여러 가지 모델에 대한 검증 방법들이 있는데, SPIN/SMV을 이용한 검증을 통하여 모델이 요구하는 조건을 정말로 만족하는지 검증(Verification)할 계획이다.[5][6]

참고 문헌

[1] Andre M. van Tilborg, "Foundations of Real-Time Computing-Formal Specifications and Methods", Kluwer Academic Publishers, 1991

[2] Debra S. Herrmann, *Software Safety and Reliability : techniques, Approaches, and Standards of Key Industrial Sectors*, 1998

[3] Harel, D. and M. Politi, *Modeling Reactive Systems with Statechsrts : the STATEMATE Aproach*, 1998

[4] David Harel and Amnon Naamad, "The STATEMATE Semantics of Statecharts", *ACM Trans. soft. eng. Method 5:4*, 1995.

[5] Gerard J. Holzmann, *Design and Validation of Computer Protocols*, Prentice Hall, 1991

[6] Kenneth L. McMillan, *Symbolic Model Checking*, *Kluwer* Academic Publisher, 1993