

SCADE를 활용한 항공기 공중충돌회피시스템-II의 통합 시뮬레이터 설계

Design of an Integrated Simulator for TCAS-II using SCADE

전병규*, 임상석*

Byung-Kyu Jun*, Sang-Seok Lim*

요 약

본 논문에서는 SCADE를 사용하여 항공기 안전필수 시스템인 공중충돌회피시스템(TCAS-II)을 위한 통합 시뮬레이터를 설계한다. 이 TCAS-II시뮬레이터는 기존의 충돌회피 알고리즘의 문제점을 분석하고 실제 공중 충돌 상황을 더욱 현실성 있게 시스템 평가에 반영할 수 있도록 편리한 통합시뮬레이션 환경을 반영하여 구현한다. 또한 안전 필수요건인 소프트웨어 국제표준 신뢰성 요건을 만족하면서 실제로 실험할 수 없는 상황에 대한 TCAS-II의 성능분석 및 새로운 충돌회피 로직 연구를 위한 시나리오 편집기를 제시하고 이를 사용하여 전형적인 공중충돌 시나리오에 대한 시뮬레이션을 통해 그 성능을 검증한다.

Abstract

In this paper, we propose a new SCADE-based integrated simulator for aircraft mid-air collision avoidance. The new simulator is designed to provide a convenient simulation environment for analyzing the limitations of the existing collision avoidance system(CAS) and for realizing more realistic collision environment into the CAS performance evaluation. Furthermore, we have developed a scenario editor for TCAS-II performance analysis to simulate actual flight environment which is difficult to experiment. The performance of the proposed simulator for typical collision scenarios is demonstrated by simulation using the proposed scenario editor.

Key words : TCAS-II, ACAS, DO-178B, SCADE

I. 서 론

항공산업은 전 세계적인 항공 노선망의 형성과 항공수송수요의 급증 등으로 급속한 발전을 거듭하였다. 그럼에도 불구하고 아직도 이착륙 지연, 항공사고 등의 문제점이 끊임없이 발생하고 있으며, 이에 전 세계적으로 이러한 문제들을 해결하기 위해 ICAO, FAA, RTCA, EASA, EUROCAE 등의 국제기구를 설립하여 주기적으로 모임을 갖고 현행 시스템

의 보완 및 새로운 기술도입에 필요한 기준 및 국제표준을 정비해 나가고 있다[1][2]. 특히 항공기간 공중충돌은 사고발생 시 더 많은 인명피해와 재산피해가 발생하므로 RTCA의 DO-185B 등의 문서에서 중점적으로 다루고 있다. 또한 각 연구기관에서도 현 시스템을 보완하기위해 노력하고 있다[3].

하지만 이러한 노력과 관련 항공안전 관련 항공전자 기술의 꾸준한 발전에도 불구하고 항공사고는 아직도 빈번히 발생하고 있으며 해마다 평균 30건 이상

* 한국항공대학교 항공전자 및 정보통신공학부(School of Avionics, Telecommunication and Computer Eng., Korea Aerospace University)

· 제1저자 (First Author) : 전병규

· 투고일자 : 2009년 10월 8일

· 심사(수정)일자 : 2009년 10월 9일 (수정일자 : 2009년 10월 23일)

· 게재일자 : 2009년 10월 30일

의 사고와 2000여명의 사망자가 나오고 있다[4]. 또한 준사고(Incident) 역시 꾸준히 발생하고 있으며, 이는 현 시스템이 근본적인 문제의 해결책을 제시해 주지 못하고 있다는 것을 의미한다[5][6].

본 논문에서는 기존의 공중충돌회피장치 TCAS (Traffic alert and Collision Avoidance System)-II의 문제점을 분석하기 위한 시뮬레이터를 구성하고 이 시스템의 동작을 분석한다. 또한 새로운 충돌회피알고리즘을 시험하기 위한 환경을 구축하고 이를 통해 보다 쉽고 편리하게 새로운 알고리즘을 검증할 수 있는 통합시뮬레이션 환경을 제시하고자 한다. 이 통합시뮬레이션 환경을 사용하면 실제 비행시험이 매우 어려운 항공기 공중충돌 상황에 대한 분석을 용이하게 수행할 수 있으며 더욱이 새로운 알고리즘과 계산로직을 적용하였을 때 그 영향을 쉽게 분석할 수 있다. 따라서 여기서 제안하는 새로운 시뮬레이션 환경을 사용하면 기존의 필드테스트가 가지는 한계에서 벗어나 수시로 다양한 조건과 상황에 대한 검증과 영향을 확인할 수 있다.

본 논문의 구성은 다음과 같다. 제 2장은 TCAS-II의 기본개념과 회피 알고리즘에 대해 기술한다. 제 3장에서는 본 연구를 위해 사용한 TCAS-II 로직 개발 환경인 SCADE 소프트웨어 특성을 소개하고 안전필수적인 시스템의 소프트웨어 인증기준인 DO-178B와의 연관성을 기술한다. 제 4장에는 SCADE를 기반으로 설계한 TCAS-II 로직과 현시화면에 대해 설명하고, 제 5장에서는 시뮬레이션을 통해 제안한 통합 시뮬레이터 환경의 동작을 검증한다.

II. TCAS-II

TCAS는 항공기 공중충돌회피시스템으로 유럽에서는 ACAS(Airborne Collision Avoidance System)라고도 불린다. 이 시스템은 지상과 독립적으로 운용되는 항행안전시스템의 하나로, 그 성능에 따라 TCAS I, II, 또는 III로 구분되며 현재 TCAS-II Version-7이 널리 사용되고 있다. TCAS-II는 자신과 상대항공기의 위치를 항공기 조종사에게 알려주고 충돌위험이 발생하면 수직방향에 대한 회피경로를 제공해 준다.

TCASII - Version 7은 RTCA의 표준성능지침인

DO-185B에 시스템의 요구사항과 테스트를 위한 단계, 요구되는 성능기준 및 표준성능 등이 기술되어 있다[7].

2-1 TCAS-II의 동작원리

TCAS-II는 접근하는 항공기에 대한 충돌위험을 판단하기 위해 거리테스트와 고도테스트를 수행한다.

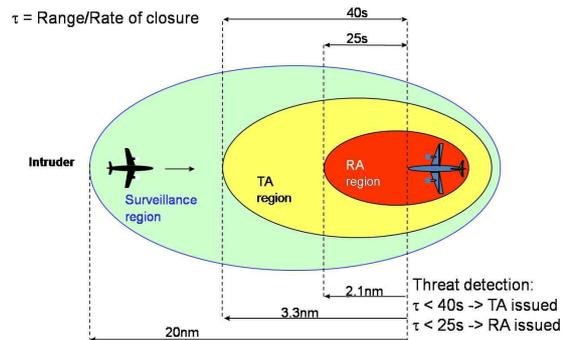


그림 1. τ -공간 및 거리테스트

Fig. 1. The range test in τ -space.

거리테스트는 τ -공간(τ -space)라는 시간개념을 이용하여 상대항공기와의 조우예상지점 또는 최단접근지점 CPA (Closest Point of Approach)에 대한 도달시간을 초 단위로 계산한다. 그리고 이 값을 미리 지정되어 있는 위험정도인 TA, RA의 문턱값과 비교하여 충돌 위험여부와 그 정도를 판단한다. TA (Traffic Advisory)는 상대 항공기와 자신의 항공기가 현재 경로와 속도를 유지하였을 경우 위험해질 가능성이 크다는 의미이며 이를 지정된 심볼로 디스플레이 장치에 현시하고 음성경고를 통해 조종사에게 알려준다. RA(Resolution Advisory)는 충돌위험에 직면한 상황이므로 조종사는 즉각적으로 TCAS-II에서 제공하여 주는 회피경로로 항공기를 기동해야 한다는 의미이다. TA와 마찬가지로 디스플레이 장치와 음성경고로 조종사에게 위험여부 및 회피경로를 지시한다. TA와 RA가 발생하는 영역을 도시하면 그림 1과 같다. 그림에서, 상대 항공기가 탐지영역에서 발견되면 침입기(Intruder)로 간주되고 계속 접근하여 노란색으로 표시된 TA영역에 도달하면 고도테스트의 결과에 따라 이 항공기는 TA(위험경보)를 발생시킨다. 마찬가지로 탐지항공기가 빨간색으로 표시된 RA영역에 들

어오면 고도테스트를 거쳐 해당 항공기를 위협기 (Threat)로 판단하고 RA(회피경보)를 발생시키고 이 항공기에 대한 회피방향을 지시한다.

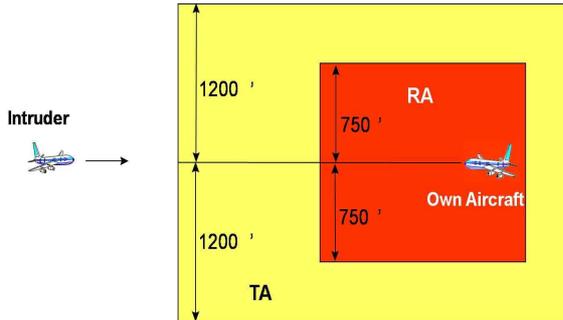
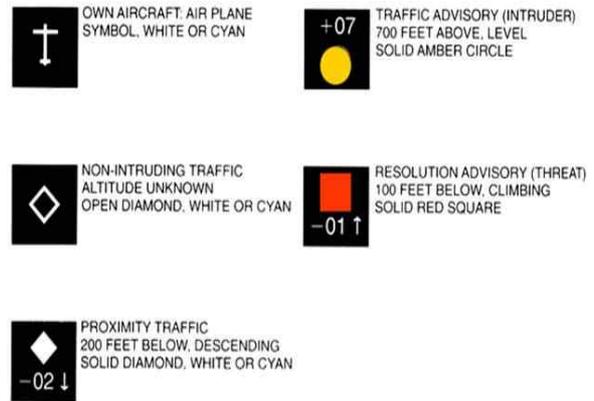


그림 2. 고도테스트에서의 수직분리
Fig. 2. Vertical Separation in altitude test.

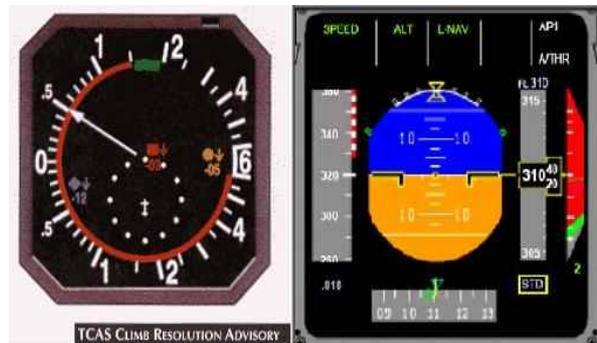
고도테스트(그림 2)는 거리테스트와는 조금 다르게 시간개념이 아닌 고도차를 기준으로 위협여부를 판단한다. 이 고도차는 수직분리(VERTICAL SEPARATION) 또는 고도 문턱값(Altitude Threshold)라고 불린다. 거리테스트에서 시간이 미리 지정되어 있듯이 고도 또한 그 문턱값이 정의되어 있으며 고도테스트의 예를 그림 2에 나타내었다. 거리테스트의 τ 값과 고도테스트의 수직분리값은 Sensitivity Level(SL)에 따라 더욱 세분화되며 이 그룹은 자신의 항공기의 고도를 기준으로 분류된다[7]. 각 SL는 TA와 RA의 τ 값과 수직분리값, 그리고 매우 느린 접근에 대비한 DMOD의 문턱값으로 구성되어 있다.

2-2 TCAS-II 경보

TCAS-II에서 조종사에게 위협을 알리는 방법은 2가지가 있다. 첫 번째는 디스플레이 장치에 위협 항공기를 특정한 모양과 색을 통해 현시하는 방법이고, 두 번째는 음성으로 경고하는 방법이다. TA와 RA 상황에서 이 두 가지가 모두 사용되며 조종사가 상황을 인지하고 회피기동을 할 수 있도록 도와준다. 현시 장치는 독립적인 TCAS-II 디스플레이가 탑재되는 경우가 있고 PFD(Primary Flight Display)에 통합되는 경우도 있다. 또한 ND(Navigation Display)와 기상레이더 디스플레이에 통합되는 경우도 흔히 있다.



(a) TCAS 디스플레이에 사용되는 심볼



(b) TCAS 수직속도지시기 및 TCAS 통합 PFD

그림 3. TCAS 심볼 및 수직속도지시기
Fig. 3. The symbols for TCAS display and VSI.

그림 3(a)는 TCAS-II 디스플레이에 사용되는 심볼을 나타내며 그림 3(b)는 TCAS-II 디스플레이와 PFD에 통합된 TCAS-II VSI(VERTICAL SPEED INDICATOR)를 보여준다. PFD에 통합된 이 VSI를 통해 조종사는 TCAS-II 디스플레이나 ND를 확인하지 않고 계속 PFD를 주시하면서 회피기동을 수행할 수 있다.

III. SCADE와 DO-178B

3-1 SCADE

SCADE (Safety-Critical Application Development Environment)는 Esterel Technologies사에서 제작한 항공·우주, 방위, 원자력, 철도, 자동차 및 신호체계관리 등 안전필수적 또는 임무결정적(Mission-Critical)인 시스템의 소프트웨어를 개발할 때 사용할 수 있는 통합개발환경이다. 특히 DO-178B, IEC 61508, 또는

EN 50128 등 각 분야에서 소프트웨어 개발 시 사실상 필수적으로 따라야 하는 인증절차에 맞는 개발과정을 수행할 수 있게 최적화 되어있다. 개발과정이 텍스트로 이루어진 코드기반이 아닌 그래픽으로 소프트웨어를 구성하는 디자인기반, 또는 모델기반이라 불리는 정형기법 개발방법을 택하고 있어 기존의 개발방법보다 나은 가독성(Readability)을 제공한다. 또한, 직관적인 심볼의 사용으로 시스템 엔지니어와 소프트웨어 개발자 간의 개발협력을 보다 용이하게 만든다. 사용자가 디자인기반으로 소프트웨어를 설계하면 검증된 코드생성기에 의해 규격화된 C 코드를 생성하여 준다. 이렇게 생성된 코드는 일체의 검증과정과 수정 없이 약간의 글루코드(Glue code)만으로 바로 사용할 수 있도록 검증이 자체적으로 수행된다. 또한 소프트웨어 요구사항에서부터 최종 C 소스 코드 생성까지 완전한 추적성(Traceability)을 제공하여 줌으로써 해당 분야의 소프트웨어 개발에 최적화된 환경을 제공한다[8][9].

비록 SCADE를 이용하여 소프트웨어의 개발을 용이하게 진행할 수 있지만 기본적으로 알고리즘의 설계와 모델의 구현은 전적으로 개발자의 몫이다. 다만 SCADE가 정형기법을 이용하므로 SCADE에 대한 사용법을 익힌 후에는 보다 쉽게 안전필수적 소프트웨어를 설계하고 구현할 수 있다. 또한 기존의 검증 방법에 비해 소요되는 시간과 비용, 인력을 격감시킬 수 있으므로 소규모 팀에서도 개발을 진행 할 수 있다.

Mathematical	
Logical	
Structure/Array	
Higher Order	
Comparison	
Time	
Choice	

그림 4. SCADE에서 사용하는 기초연산모듈
Fig. 4. The Elementary modules in SCADE.

SCADE에서는 그림 4에서 사용된 것과 같은 기본

적인 연산모듈과 리미터, 각종 필터 및 삼각함수 등의 연산을 처리하기 위한 라이브러리를 제공하여 주기 때문에 기초모듈을 조립하여 설계하고자 하는 소프트웨어의 구조와 알고리즘을 구현한다. 이들 각 기초모듈은 일반적인 도식 형태로 표현되어 있으므로 사용상 편의성을 제공한다. SCADE에는 데이터흐름이나 제어흐름에 따른 알고리즘을 설계할 수 있는 SCADE Suite와 디스플레이 장치의 소프트웨어를 설계할 수 있는 SCADE Display가 있다.

3-2 RTCA DO-178B

항공기 탑재 시스템의 소프트웨어는 분야의 특수성으로 인해 다른 소프트웨어와 다르게 매우 엄격한 기준으로 설계되고 제작되어야 한다. 이에 RTCA는 DO-178B라는 소프트웨어의 개발 계획에서부터 최종 결과물까지의 전 생명주기(Life-Cycle)를 아우르는 인증과정을 제시하고 있으며 사실상 민간 항공기에 탑재되는 안전 관련시스템의 소프트웨어는 모두 이 인증과정에 따라 검증되어야 한다[10]. DO-178B에는 특히 소프트웨어 요구사항과 최종 결과물인 목적코드(Object Code)까지의 완전한 추적성을 요구하고 있으며 특히 소프트웨어의 검증에 매우 큰 비중을 두고 있다. 이 검증절차는 소프트웨어가 올바르게 동작하는지를 단순히 테스트하는 것뿐만 아니라 테스트가 소프트웨어의 구조를 모두 포괄할 수 있도록 진행되는지, 즉 검증에 대한 검증까지 포함하고 있다.

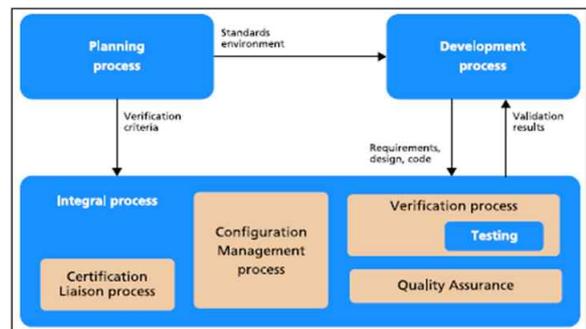


그림 5. DO-178B의 소프트웨어 생명주기
Fig. 5. The software life cycle in DO-178B.

DO-178B는 시스템의 안전필수적 요소에 따라 A부터 E까지 총 5단계로 세분화되며 레벨A는 공중충돌회피 시스템, 지형지물충돌회피 시스템, 항공기 제

어시스템 등 주요 항공전자시스템의 소프트웨어가 충족해야하는 안전단계이다. 레벨이 낮아질수록 항공기 안전과의 관련성이 떨어진다는 의미이며 최하위 레벨인 E에는 각종 오락시스템(Entertainment system)의 소프트웨어에 관한 기준 절차가 제시되어 있다.

그림 5는 DO-178B에서 제시하는 소프트웨어 개발 생명주기를 나타내며 각 단계는 세부단계를 포함하고 있다. 계획과정(Planning Process)에서는 요구사항을 반영하여 소프트웨어 개발을 계획한다. SCADE에서는 RM Gateway라는 기능을 통해 DOORS 등의 전문 요구사항관리 툴과의 연결을 지원하며 자체적으로도 마이크로소프트 오피스 파일 등으로 제작된 요구사항과 SCADE의 각 모듈, 또는 구성요소간의 추적성을 유지시켜 준다. 개발과정(Development Process)에서는 계획과정에서 결정된 개발표준에 근거한 실제 소프트웨어의 개발에 대한 내용을 기술하는데 SCADE를 사용하는 경우 기본 모듈개발에 대한 검증이 자체적으로 이루어진다. 여기에 인증 받은 자동코드생성기를 사용하면 기존방식에서 개발한 소스코드를 각 라인별로 하나씩 안전기준에 적합한지 검증해야하는 작업을 제거해 주며 개발 기간과 비용을 획기적으로 줄일 수 있다. 통합과정(Integral Process)은 개발과정과 병렬적으로 수행된다. 이 과정은 소프트웨어의 검증절차와 형상관리, 인증획득절차, 품질보증절차 등을 포함하고 있다. SCADE는 자체적으로 정형검증기법과 시뮬레이션, 그리고 소프트웨어 모델의 구조에 대한 검증을 위한 모델 테스트 망라도 분석(Model Test Coverage Analysis)을 제공한다. 따라서 개발한 소프트웨어의 검증 뿐 아니라 검증에 대한 검증까지 자체적으로 수행할 수 있다.

IV. SCADE기반의 TCAS-II 시뮬레이터 설계

여기서는 SCADE를 사용하여 가상적인 항공기를 추적하여 충돌회피 상황에 대한 정보를 제공할 수 있는 TCAS-II 시뮬레이터를 설계한다. 또한 전반적인 항공기의 접근 상황과 주요 데이터를 산출하는 시뮬레이터를 구성하고 시뮬레이션 결과를 나타내기 위해 TCAS-II 디스플레이를 구현한다. 이 시뮬레이터

와 TCAS-II 디스플레이를 통합하여 각 시나리오에 따라 실제 TCAS-II 시스템에서 발생할 수 있는 공중 충돌 상황을 시뮬레이션 할 수 있도록 조종석에서 일어나는 환경을 그대로 반영하도록 구성한다. SCADE Suite로 설계한 TCAS-II Logic Computer의 최상위 레벨 흐름도를 나타내면 그림 6과 같다.

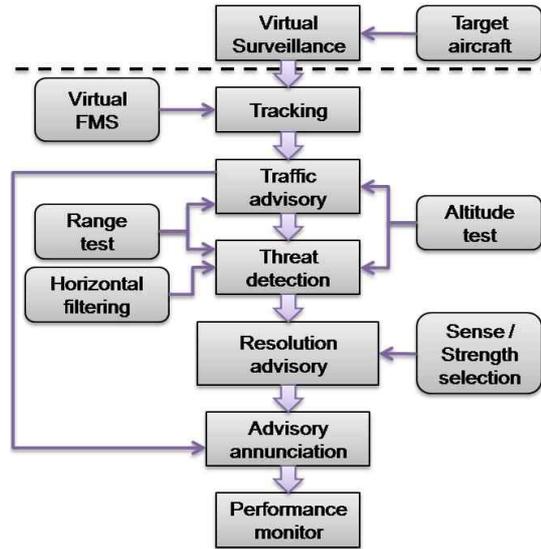


그림 6. TCAS-II 컴퓨터의 최상위 레벨 흐름도
Fig. 6. Top-level flow chart of TCAS-II Computer.

이 흐름도에서 각 모듈은 SCADE 모델에 상응되며 SCADE 환경 내에서 전 과정을 DO-178B에서 제시하는 기준에 맞추어 설계하였다. 이 흐름도는 DO-185B에서 제시하는 TCAS-II 로직을 사용하여 설계하였다. 각 모듈과 요구사항의 추적성은 SCADE 자체적인 요구사항관리 기능을 사용하여 유지하였다. 실제시스템의 환경을 그대로 반영하여 통합 시뮬레이터로 동작시키기 위해서 감시단계 및 비행정보는 각각 가상감시단계 및 가상비행관리 시스템으로 대체하였다. 이와 같이 통합 시스템을 구성함으로써 실제상황과 동일한 환경에서 테스트하듯이 시스템 성능을 검증할 수 있도록 구성하였다(그림 6). 그림 6의 가상감시시스템과 가상비행관리시스템은 두 항공기에 대한 비행정보와 필요한 환경 데이터를 생성하여 추적단계(Tracking)로 보내준다. 이 추적단계에서 TCAS-II 컴퓨터는 추적 대상 항공기에 대한 비행궤적을 추적하고 저장한다. 이 자료를 이용하여 TA 단계와 RA 단계에서는 2.1절의 기준에 따라 각각

항공기의 위협 정도와 충돌 위험을 판정한다. 이 판정 결과에 따라 RA 경보가 발생하면 회피 경로를 조종사에게 제공한다.

그림 7은 SCADe로 구현한 TCAS-II 시뮬레이터의 최상위 레벨을 나타낸다. 이 시뮬레이터의 데이터 흐름은 그림 6의 흐름도와 동일하게 구성하였으며 흐름도의 각 단계는 하나의 블록으로 표현되어 있다. 이 최상위 디자인은 여러 단계의 하위 디자인을 포함하며 계층적인 특성을 가지고 있다. 즉, 흐름도의 각 단계는 각 모듈로 구성되고 이 모듈 또한 하위 세부 모듈로 구성되어 있다. 이 계층구조의 최하위 모듈은 그림 8과 같이 SCADe의 기초연산모듈로만 구성된다.

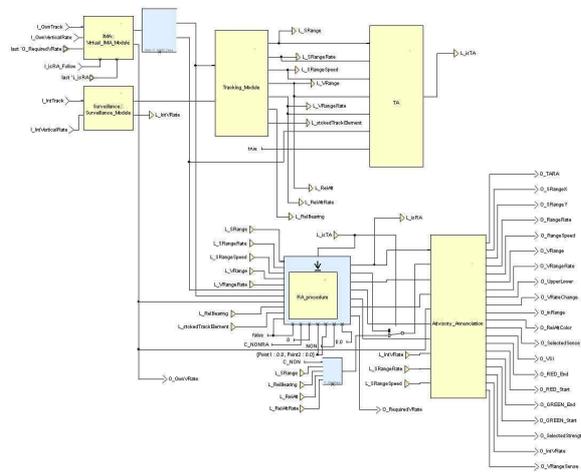


그림 7. 제안된 TCAS-II 시뮬레이터의 최상위 레벨
Fig. 7. Top-level of the proposed TCAS-II simulator.

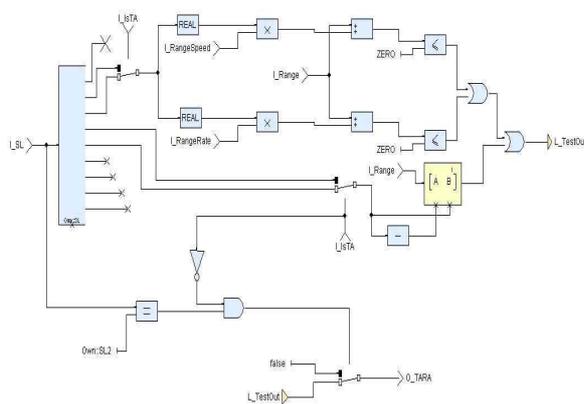


그림 8. SCADe 기초연산모듈로 구성된 최하위 레벨
Fig. 8. The Lowest-level of the simulator consisted of SCADe elementary operators.

그림 8은 거리테스트 단계를 구현한 최하위 모듈의 한 예이다. 마찬가지로 그림 6의 각 단계도 동일한 원리로 구현하고 그 모듈을 모두 조합하여 통합 TCAS-II 시뮬레이터를 설계하였다.

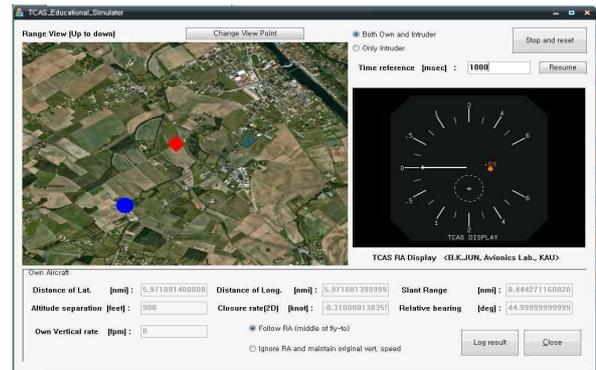


그림 9. TCAS-II 통합 시뮬레이션 환경
Fig. 9. TCAS-II integrated simulation environment.

V. 시뮬레이션

5-1 TCAS-II 통합 시뮬레이션 환경

위에서 설계한 TCAS-II 시뮬레이터의 성능을 평가하기 위해 SCADe로 구현한 TCAS-II 모델로부터 생성된 C Code를 사용하여 M/S Visual C++ MFC 환경에서 시뮬레이터를 구성하였다. 즉, 전반적인 인터페이스 및 자신의 항공기와 상대 항공기의 전반적인 비행 상황, 즉 거리, 방위, 고도 차 등을 확인할 수 있는 상황 디스플레이를 M/S MFC로 구성하였으며, 계산 로직은 SCADe로 생성한 C Code를 사용하였다. 이 C Code는 SCADe에서 검사 완료된 모델로부터 자동 생성함으로써 코드에 대한 검증과정은 생략하였다. 또한, 실제상황에서 TCAS-II를 시험하는 것과 마찬가지로 상황에서 성능을 평가할 수 있도록 그림 9와 같이 통합 시뮬레이션 환경을 구현하였다.

이 통합 시뮬레이터는 크게 좌측 상단의 상황디스플레이화면과 우측의 TCAS-II 디스플레이화면, 그리고 하단의 비행 데이터를 보여주는 부분으로 구성하였다. 상황디스플레이의 원형 심볼과 다이아몬드형 심볼은 각각 자신의 항공기와 상대 항공기를 나타내며, 'Change View Point' 버튼을 이용하여 거리상황을 파악할 수 있는 상대거리시점과 고도차를 파악할 수

있는 상대고도시점을 선택할 수 있게 구성하였다. 이 화면은 M/S사의 MFC로 구현하였다. SCADE Display 소프트웨어로 직접 제작한 TCAS-II 디스플레이 화면은 DO-185B 기준에 부합하도록 만들었다. 이 화면은 탐지된 항공기와 그 항공기의 위협여부, 상대 고도 및 상대고도변화율을 현시하며 수직속도지시계 VSI가 통합되어있다. RA 발생 시 이 지시계가 가리키는 도넛형 눈금에 녹색과 적색으로 회피해야 할 방향과 속도를 지시해 준다. 지시되는 비행 데이터는 항공기 상대거리, 고도 차, 접근속도 및 상대방위각 등이며 회피지시경로로 자동 조종되는 옵션과 이 회피지시를 따르지 않는 옵션을 선택할 수 있도록 설계하여 TCAS-II의 회피 특성과 그 문제점을 분석할 수 있게 하였다. 하단 우측의 'Log result' 버튼을 통해 RA 회피지시 준수 여부에 따른 항공기 궤적을 텍스트와 그림파일로 출력할 수 있도록 구성하였다. 또한, TTS(Text-To-Speech) 프로그램을 사용하여 음성 경고를 제작하여 시뮬레이터에 통합하였다. 따라서 TCAS-II 디스플레이 화면과 함께 조종사가 직면하는 실황을 현시화면에 동일하게 연출할 수 있도록 통합 시뮬레이션 환경을 구성하였다.

5-2 통합시뮬레이터에 의한 TCAS-II 특성분석

시뮬레이션을 위해 몇 가지 전형적인 시나리오를 작성하여 테스트하였으며, 지면 제한으로 인하여 그 중 한 가지 시나리오에 대한 결과를 그림 10에 나타내었다. 그림 10(a)는 항공기가 탐지되었으나 아직은 위협적인 항공기로 판정되지 않은 경우를 나타낸다. 그림 10(b)는 해당 항공기가 충돌위험 가능성이 있음을 보여주는데 당초의 흰색-다이아몬드에서 황색-원형으로 심볼의 색과 모양이 변화한 것을 통해서 TA 경고 상황을 확인할 수 있다. 그림 10(c)는 상대 항공기가 충돌위험항공기로 판정된 상황을 보여주는데 그에 따라 이 위험항공기 심볼은 황색-원형에서 적색-사각형으로 모양과 색이 변화됨과 동시에 회피(RA) 경보를 발생시킨다. 이와 동시에 수직속도지시기에 초록색과 적색의 구간을 표시하여 줌으로써 항공기의 안전한 회피를 위해서 초록색 구간으로 회피할 수 있도록 지시하여 주는 상황을 보여준다.

이 시뮬레이션을 통해 확인한 심볼의 모양과 색의

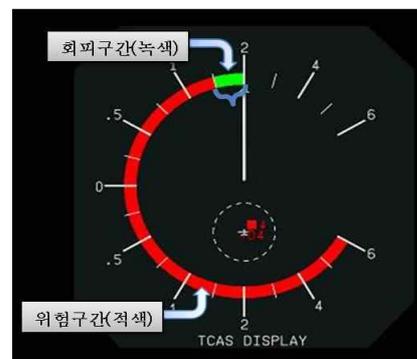
변화는 실제 조종석에서 확인할 수 있는 결과와 동일하며 DO-185B에 규정된 TCAS-II의 최소운용성능표준 MOPS (Minimum Operational Performance Standards) 의 요구사항을 만족하는 것을 확인하였다.



(a) 탐지영역에 있는 항공기



(b) TA 발생



(c) RA 발생

그림 10. TCAS Display 현시화면.
Fig. 10. TCAS Display

5-3 알고리즘의 변경에 따른 회피성능의 개선

일반적으로 무인기(UAVs)처럼 여객기나 군용기에

비해 최대 고도상승률이 낮은 항공기의 경우 TCAS-II 알고리즘에서 지시하는 회피 상승률을 따르지 못할 수도 있다. 예를 들어, 최대상승률(MVS) 600fpm의 무인기는 기존 TCAS-II 알고리즘의 큰 고도상승지시를 수행하기 불가능하기 때문에 더 낮은 고도상승률이 지시되어야 한다. 이런 상황에서 TCAS-II 알고리즘의 RA 기준을 사용할 경우, 상대 항공기의 기동이나 두 항공기의 회피궤적에 따라 충돌 또는 NMAC(Near Mid-Air Collision)와 같은 위험이 존재한다. 따라서 RA의 τ 기준값과 회피강도는 적절히 조정할 필요가 있다. 이러한 충돌 위험 상황에 대한 특성을 분석하기 위해 표준 시나리오를 사용하고 시뮬레이션 결과를 통해 제안된 시뮬레이터의 유용성을 예증한다. 표준 시나리오는 DO-185B에 성능 테스트용으로 제시된 'EN02TS02.DAT' 파일을 사용하였고, 회피지시에 대한 반응시간은 0s, RoC(Rate of Climb)의 변화율은 5ft/s를 사용하고 회피강도는 600fpm이라 가정하였다. 시뮬레이션 결과는 그림 11에 나타내었다.

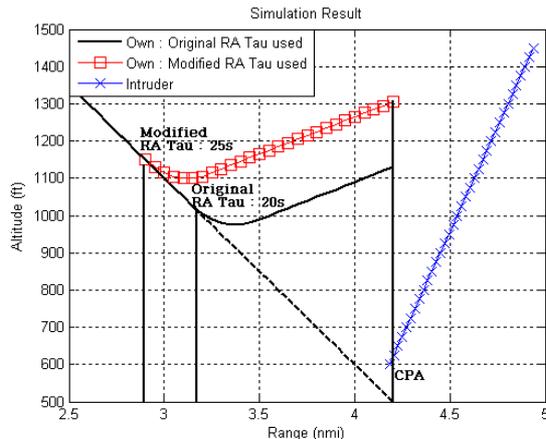


그림 11. τ 값 변경에 따른 충돌회피 결과

Fig. 11. Avoidance result by modifying τ -value.

이 결과를 보면 τ 가 현행 TCAS-II 시스템의 RA 판별 문턱값 τ 가 20초이면 CPA에서 수직분리간격이 530ft로 ALIM(Altitude Limit)에 너무 근접하여 매우 불안하지만, τ 를 25초로 조정하면 수직분리간격은 705ft로 증가하여 안전하게 회피할 수 있음을 확인할 수 있다. 이 결과를 보면 제안된 시뮬레이터를 사용한다면 공중충돌회피가 어려운 상황에서도 안전

한 회피를 하기 위해 필요한 τ 값 산출을 원활하게 수행할 수 있음을 알 수 있다.

VI. 결 론

본 논문에서는 항공기 안전필수 시스템 중 하나인 TCAS의 충돌회피 성능에 대한 분석 및 결점을 보완하는데 활용할 수 있는 통합 시뮬레이터를 SCADE 기반으로 설계하였다. 이 시뮬레이터를 사용하여 기존 TCAS 충돌회피 알고리즘의 문제점을 분석하고 새로운 충돌회피 알고리즘을 개발할 수 있는 환경을 제시하였다. 또한, 실제 RTCA DO-185B에 포함되어 있는 성능평가용 시나리오에 대해 새로운 τ 값 적용에 의한 효과적인 충돌회피 결과를 확인하였다. 따라서 이 시뮬레이터를 사용하면 새로운 충돌회피 알고리즘 개발에 대한 연구 뿐 아니라 실제 상황에서 실험할 수 없는 위험한 여러 가지 충돌상황들을 연출하여 직접 그 영향을 평가할 수 있다. 나아가서 TCAS 시스템에 대한 조종사 교육은 물론 항공전자 관련 시스템 운용평가 시험용으로 활용할 수 있을 것으로 기대된다.

참 고 문 헌

- [1] 오경륜, 김종철, 남기욱, “차세대 위성항행시스템(CNS/ATM) 기술개발 동향”, *항공우주산업기술동향*, 제2권 제1호, pp. 51-63, 7. 2004.
- [2] 임상석, “차세대 항행 시스템 CNS/ATM”, *전자공학회지*, 제26권 제4호, pp. 15-26, 4. 1999.
- [3] A. D. Zeitlin, T. Arino and J. Kuchar, “Improving the Resolution Advisory Reversal Logic of the Traffic Alert and Collision Avoidance System,” *Proc. of 25th Digital Avionics Systems Conference*, IEEE/AIAA, Oct. 2006
- [4] H. Ranter, *Airliner Accidents Statistics 2006*, Aviation Safety Network, Jan. 2007.
- [5] <http://www.ntsb.gov/Aviation>
- [6] 하재구, “항공교통 공중충돌 방지대책”, *항공진흥*, 제31호, pp. 9-21, 8. 2003.

- [7] "Minimum Operational Performance Standards for Traffic Alert and Collision Avoidance System II (TCAS II)", *RTCA, Inc. - RTCA DO-185B*, Vol. 1 and 2, Mar. 2007
- [8] "Software Considerations in Airborne Systems and Equipment Certification", *RTCA, Inc. - RTCA DO-178B*, Dec. 1992
- [9] "Software Considerations in Airborne Systems and Equipment Certification", *RTCA, Inc. - RTCA DO-248B*, Oct. 2001
- [10] http://www.faa.gov/regulations_policies/

전 병 규 (田秉圭)



2009년 2월 : 한국항공대학교 항공
전자공학과(공학사)
2009년 10월 : 한국항공대학교 항
공전자공학과 석사과정
관심분야 : 항공전자, CNS/ATM,
SVS, TCAS, TAWS 등

임 상 석 (林尙石)



1976년 2월 : 한국항공대학교 항공
전자공학과(공학사)
1984년 9월 : Ottawa Univ. 전자 및
컴퓨터 공학과(공학석사)
1990년 3월 : Ottawa Univ. 전자 및
컴퓨터 공학과(공학박사)
1976년 ~ 1982년 : 국방과학연구소

연구원

1990년 ~ 1992년 : Canada Royal Military College, R.A.
1992년 ~ 1995년 : DREO Canada, Research Scientist
1995년 ~ 현재 : 한국항공대학교 항공전자 및 정보통신
공학부 교수
관심분야: 항공전자, CNS/ATM, SVS, TCAS, 필터설계