

UAV를 위한 이산사건 및 연속시간 시스템간의 연동 모델링에 대한 연구

강광천^{1†} · 유용준¹ · 지승도¹

A study on the co-operative modeling between discrete-event system and continuous-time system for UAV system

KwangChun Kang · YongJun You · SungDo Chi

ABSTRACT

The major objective of this paper is to propose a hybrid simulation environment for autonomous UAV system by integrating the continuous-time model with discrete-event model. Proposed system is able to support high autonomous behavior by combining the planner, recognizer, and controller model to deal with the HL20 AIRPLANE model. Thus, the high level decision may be efficiently issued even upon the unexpected circumstance. The proposed system model has been successfully verified by several simulation test performed on the DEVS simulation S/W environment.

Key words : UAV, DEVS, autonomous, hybrid

요 약

UAV(Unmanned Air Vehicle) 시스템은 새로운 알고리즘과 소프트웨어 디자인에 바탕을 두고 빠르게 발전하고 있다. 빠르게 발전하는 현대의 UAV 시스템은 상황에 따른 효과적이고 지능적인 제어를 요구한다. 이에 본 논문에서는 UAV 시스템의 효과적이고 지능적인 제어를 위하여 이산사건 시스템인 조종사 모델과의 연동 모델링을 제안한다. 비행기 모델은 연속시간 시스템으로 표현되며, 자세한 표현력을 바탕으로 정량적이고 정확한 비행기 모델을 표현할 수 있다. 또한, 조종사 모델은 이산사건 시스템으로 표현되며, 각 사건과 시스템의 상태에 따른 정성적인 행동제어를 가능하게 한다. 본 연구는 한국항공대학교에서 개발한 이산사건 시뮬레이터인 DEJAVA(DEVS Java)와 연속시간시뮬레이터인 MATLAB 시뮬레이터 환경을 바탕으로 NASA에서 개발된 HL20 비행시뮬레이터와 지능제어시스템을 이용한 조종사 모델을 사용하여 구현되었다.

주요어 : 무인항공기, DEVS, 하이브리드, 지능제어, 항공기

1. 서 론

UAV(Unmanned Air Vehicle:무인항공기) 시스템은 조종사가 타지 않고도 스스로 주어진 명령에 따라 비행하며, 일을 하는 하늘의 로봇으로서 조종사 양성에 필요한 전문화적인 비용에 대한 보완책으로 많은 연구가 이루어지고 있다. 특히, 현대의 UAV 시스템은 자율적인 제어, 공중 충돌 감지 및 회피 기술과 같은 임기응변 등의 효율적이고 지능적인 비행제어를 요구하고 있다. 그러나, 이러

한 지능적인 비행제어를 위해서는 높은 자율성과 상황에 따른 정성적인 제어가 반드시 필요하다.^[1,2] 이에 본 논문에서는 효과적인 UAV의 제어를 위하여 연속시간 모델인 비행기 모델과 이산사건시스템인 조종사 모델의 연동 모델링을 제안한다.

연속시간시스템은 시간 변수가 연속이고 상태 변수가 연속인 아날로그 회로와 같은 시스템을 의미하며, 항공기 모델은 기체의 움직임과 환경모델, 물리적인 힘(force)과 모멘트(moment)와 같은 연속적인 시스템의 조합으로 이루어진다. 이산 사건시스템은 시간변수가 연속이고 상태 변수는 이산적인 영역을 지칭하며, 시스템의 상태에 따른 지능적이고 정성적인 모델링이 가능하다. 이산사건시스템은 DEVS(Discrete Event Systems Specification)형식론

2005년 12월 30일 접수, 2006년 3월 7일 채택

¹⁾ 한국항공대학교 컴퓨터공학과

주 저 자 : 강광천

교신저자: 유용준

E-mail; ilog21c@hau.ac.kr

을 통해 컴퓨터 네트워크, 공장자동화, 지능 로봇 시스템, 교통망 모델링 등 여러 분야의 지능제어 시스템으로 성공적으로 적용된 바 있다.^[3-5] 또한, 이산사건 및 연속시간 시스템의 연동 모델링은 각 시스템이 가지는 요소를 유지하면서, 전체 시스템을 모델링하기 위한 방법으로서, 시스템의 효과적인 제어를 위한 방법으로 로봇, 자동차 제어 등에 성공적으로 적용된 바 있다.^[6,7]

2. UAV 비행 시뮬레이션 시스템 구성

본 논문에서 제안된 UAV 비행시뮬레이션 시스템은 그림 1과 같이 표현된다. 시스템은 크게 EF(Experimental Frame) 모델, PILOT 모델, AIRPLANE 모델, SPACE 모델로 구성된다.^[8]

UAV 비행 시뮬레이션 시스템 모델의 AIRPLANE 모델은 비행기의 움직임을 표현하기 위한 연속시간모델이며, PILOT 모델은 의사결정을 위한 이산사건모델이다. 서로 다른 두 모델을 연동하기 위해서는 시간의 동기화와 데이터의 교환이 반드시 필요하다.^[6,7] 그림 2는 서로 다른 두 모델을 연동하기 위한 시간의 동기화와 데이터의 이동을 보여주는 순차 다이어그램이다. 시스템의 데이터 흐름은 다음과 같다.

- (1) EF 모델은 클럭을 발생시키고, 이를 SPACE 모델에 전달한다.
- (2) SPACE 모델은 이전 클럭의 제어 변수를 Cockpit

모델에 전달한다.

- (3) Cockpit 모델은 연속시간모델인 HL20 모델을 동작시킨다.
- (4) Cockpit 모델은 HL20 모델의 출력 변수(위치정보, 속도정보, 모멘트)를 받는다.
- (5) Cockpit 모델은 HL20 모델의 출력 변수를 SPACE 모델로 전달한다.
- (6) SPACE 모델은 HL20 출력 변수를 지형 정보와 함께 Recognizer 모델에 전달한다.
- (7) Recognizer 모델은 현재 HL20 출력 변수를 전달받아서 이를 기초적 상태 정보(high level information)로 변경한다. 만약, 상태정보의 변화가 있다면, Recognizer 모델은 상태정보를 Planner 모델에 전달한다.
- (8) Planner 모델은 규칙기반시스템을 바탕으로 현재 상태정보에 따른 의사결정(high level decision)을 내리고, 이를 Controller 모델에 전달한다.
- (9) Controller 모델은 의사결정을 바탕으로 다음 시간 축의 비행기 모델 제어를 위한 제어변수(받음각, 옆 미끄럼각)를 결정하여 SPACE 모델에 전달한다.
- (10) SPACE 모델은 시뮬레이션의 결과를 EF 모델에 전달한다.

본 연구에서는, 비행기를 위한 연속시간시뮬레이터로 MATLAB을 이산 사건 모델의 시뮬레이터로 한국항공대학교에서 개발한 DEJAVA(DEVS Java)를 사용하였다. MATLAB은 공학 분야에서 가장 많이 쓰이는 시뮬레이터 중 하나로서 다양한 응용분야(예: 전자공학, 기계공학 등)에 걸쳐서 연속모델 라이브러리를 제공한다.^[6] DEJAVA는 DEVS 모델 개발과 이의 시뮬레이션을 지원하는 추상 시뮬레이션 알고리즘을 JAVA 언어로 구현한 것으로서, 이산 사건시스템을 객체지향적인 관점에서 계층적으로 모듈화하여 모델링할 수 있는 환경을 제공한다. 또한, MATLAB과 DEJAVA를 연동하기 위해 JAVA 언어 인터페이스를 통해 외부 프로그램에서 MATLAB 시뮬레이션을 구동하는 JMatLink 모듈을 사용하였다.^[9]

UAV 비행 시뮬레이션 시스템

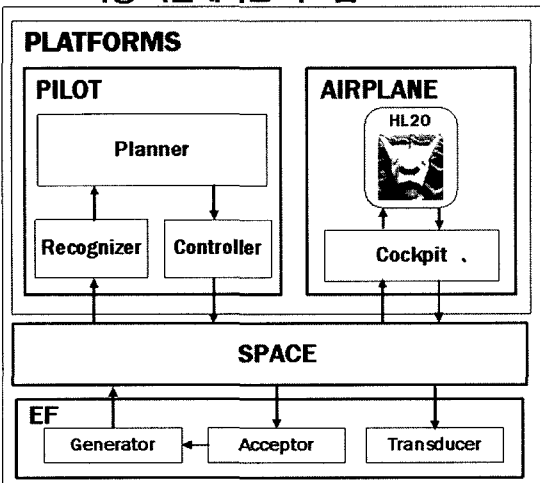


그림 1. UAV 비행 시뮬레이션 시스템 모델 구성도

3. UAV 비행 시뮬레이션 모델 구성

3.1 AIRPLANE 모델

AIRPLANE 모델은 비행기의 움직임을 담당하는 연속 시간 모델이다. AIRPLANE 모델은 HL20 모델과 Cockpit 모델로 구성된다.

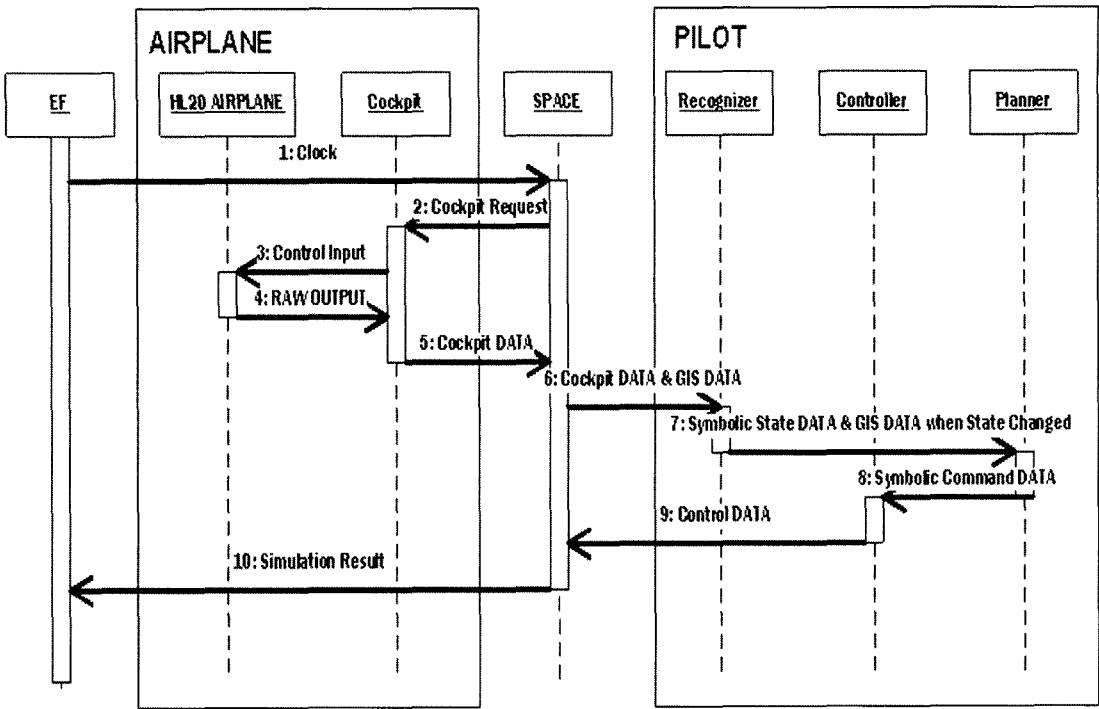


그림 2. UAV 비행 시뮬레이션 시스템 순차 다이어그램

HL20: HL20은 1989년 이후 NASA에서 진행된 프로젝트로서, 10명 내외의 인원과 물품을 운반하기 위한 PLS (Personal Launch System) 컨셉트 항공기이다. HL20 비행 시뮬레이터는 MATLAB 환경에서 구동되며, 받음각 (Angle of Attack)과 옆 미끄럼각(Angle of Sideslip)을 입력값으로, 위치정보(Location), 속도정보(Velocity), 모멘트(Roll, Pitch, Yaw)를 출력값으로 가진다. 그림 3은 HL20 비행시뮬레이터의 구성을 보여준다. HL20 비행시뮬레이터는 받음각, 옆미끄럼각, 관성, 위치정보, 속도 등을 Control System에 전달하게 된다. Control System은 Actuators를 통해 HL20 Airframe 모델에 비행기의 제어 정보를 보내게 되고, HL20 Airframe은 제어정보를 바탕으로 비행기의 움직임을 표현한다.^{[10],[11]} 표 1은 HL20 비행시뮬레이터의 구성요소들에 대한 설명이다.

Cockpit: Cockpit 모델은 HL20 연속시간 모델과 이산사건 모델을 연결시켜주는 역할을 담당한다. 서로 다른 두 시뮬레이터를 연동하기 위하여 Cockpit 모델은 JMatlink 모듈을 사용한다.^[9]

3.2 PILOT 모델

PILOT 모델은 HL20 AIRPLANE 모델에 명령을 내

리기 위하여 비행기의 현재 상황을 판단하고, 그에 적합한 제어명령을 내리는 모델이다. PILOT 모델은 Recognizer 모델, Planner 모델, Controller 모델로 구성된다.

Recognizer: Recognizer 모델은 비행기의 위치, 모멘트, 속도 등의 아날로그 데이터를 받아서, 이를 이산사건 시스템에서 사용되는 기울기와 방향에 대한 기호적 상태 정보(high level information)로 변환한다. Recognizer 모델은 상태 정보의 변화가 발생했을 때, 변경 상태를 Planner 모델에 전달한다.

Planner: Planner 모델은 Recognizer 모델로부터 받은 비행기의 상태 정보를 바탕으로 비행기 제어를 위한 의사결정(high level decision)을 내린다. 의사결정은 규칙기반시스템을 기반으로 이루어진다. 결정된 의사결정은 Controller 모델에 전달된다.

Controller: Controller 모델은 Planner 모델로부터 전달받은 이산사건 시스템의 의사결정정보를 바탕으로 아날로그 시스템에서 사용될 비행기의 제어값(받음각, 옆미끄럼각)을 설정하여 SPACE 모델에 전달한다.

표 1. HL20 비행 시뮬레이터 구성요소¹⁰⁾

구성요소	설 명
HL20 Airframe	HL20 비행기의 추상화 모델로서 기체의 움직임과 환경 모델에 대한 시스템 힘(force)과 모멘트(moment)에 의한 영향력을 계산하여 비행기의 움직임을 표현한다.
IMU,GPS	비행기의 관성정보(IMU)와 위치정보(GPS)를 Control System에 보내준다.
Airdata System	비행기 제어에 필요한 현재 위치(position)와 속도(velocity)를 Control System에 보내준다.
Guidance Model	Discrete Event 모델로부터 결정된 받음각(AoA:Angle of Attack)과 옆 미끄럼각(AoS:Angle of Sideslip)을 Control System에 보내준다.
Control System	입력 정보를 바탕으로 보조날개(aileron), 상승과 하강을 조종하는 승강기(elevator)와 방향을 조종하는 방향기(rudder), 플랩(flap)의 값을 계산하여 이를 Actuator에 보낸다.
Actuators	구동기로서, Control System으로부터 받은 보조날개, 승강기, 방향기 정보를 바탕으로 편향편차를 계산하여 실제 모델의 움직임을 계산한다.
Visualization	HL20 기체의 움직임을 3D 그래픽으로 표현한다.

3.3 SPACE 모델

SPACE 모델은 PILOT 모델과 AIRPLANE 모델간의 통신을 제어하는 역할을 담당한다.

3.4 EF 모델

EF(Experimental Frame) 모델은 시뮬레이션 평가를 위한 기반 환경을 제공하는 모델이다. EF는 Generator 모델, Transducer 모델, Acceptor 모델로 구성된다.¹²⁾

Generator: Generator 모델은 이산사건과 연속시간 모델의 동기화를 위하여 SPACE 모델에 클럭을 발생시키고, 시뮬레이션 시나리오의 초기화를 담당하는 모델이다.

Transducer: Transducer 모델은 SPACE 모델로부터 전달받은 시뮬레이션 결과를 분석하는 모델이다.

Acceptor: Acceptor 모델은 시뮬레이션의 시작과 종료 를 제어하는 모델이다.

4. 사례 연구

본 장에서는 사례연구를 통해 제안된 UAV 비행 시뮬레이션을 검증한다. 그림 3은 시뮬레이션의 이동 시나리오를 나타낸다. HL20 비행 시뮬레이터는 초기조건인 좌표 A[-12000, 0]에서 B[-5000, 2000], C[0, 4000], D[7000, 4000], E[13000, 1000]를 통해 목표좌표인 F[-20000, -2000]로 이동하게 되며, 그림 4(a)와 (b)는 각 시간에 따른 의사결정에 의한 받음각(Angle of Attack)과 옆 미끄럼각(Angle of Sideslip)의 제어과정을 보여준다.

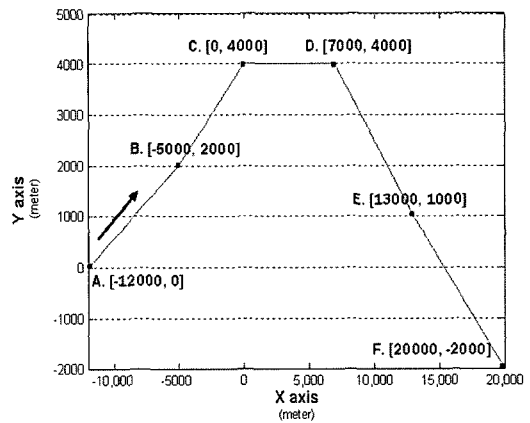
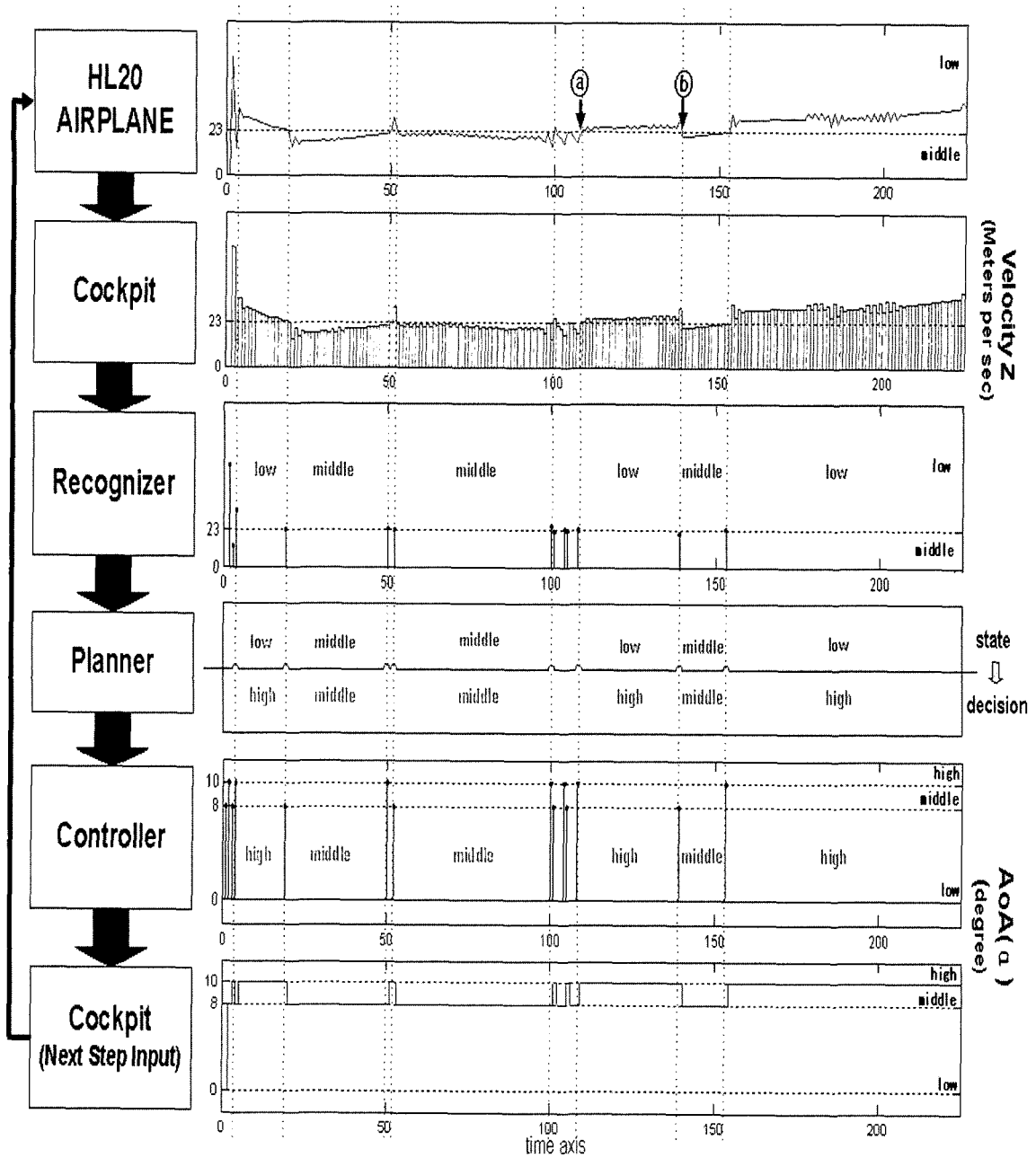


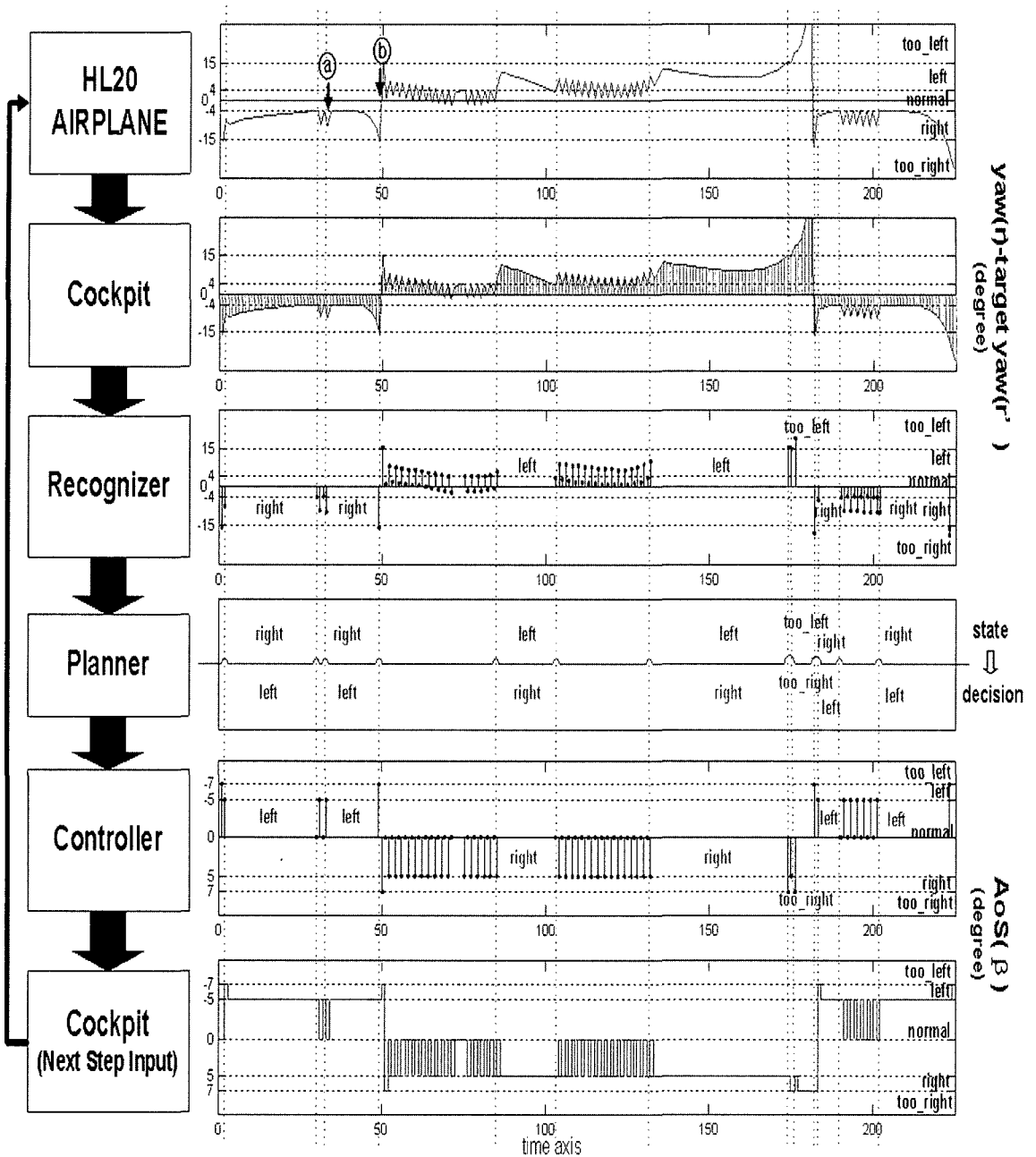
그림 3. UAV 비행 시뮬레이터 이동 시나리오

받음각 제어 의사결정과정의 예를 들면, 그림 4(a)의 ㉠ 속도 Z(Velocity Z)값은 17.49(107s)에서 23.71(108s)로 변하게 된다. Recognizer는 비행기의 기울기 상태를 인지하여, 비행기의 기울기 상태정보를 middle에서 low로 변경하고, Planner는 이에 따른 high 라는 의사명령을 내리게 된다. Controller는 high 라는 의사명령에 따라 비행기의 받음각 값을 +10도로 결정한다. ㉡ 에서는 Velocity Z값이 30.3(138s)에서 21.15(139s) 값으로 변하게 될 때, Recognizer는 비행기의 기울기 상태에서 low에서 middle로 변경하게 된다. Planner는 이에 따라 현재 상태를 유지하기 위한 middle 명령을 내리게 되며, Controller는 middle이라는 의사명령에 따라 비행기의 받음각 값을 +8도로 결정한다. 비행기의 방향제어를 위한 옆 미끄럼각의 의사결정은 이와 동일한 방법으로 이루어진다. 그림 4(b)에서 비행기는 목표 yaw(r)를 계산하여 인지된 현재 yaw(r)와 비교하여 비행기의 방향에 대한 상태정보를 인지한다(r-r'). 예를 들면 ㉢ 에서 r-r' 값은 -3.75(32s)에서 -9.7(33s)로 변하게 된다. Recognizer는



(a) 받음각(Angle of Attack) 의사결정과정

그림 4. 시뮬레이션 결과 : 받음각과 옆 미끄럼각



(b) 옆 받음각(Angle of Sideslip) 의사결정과정
 그림 4. 시뮬레이션 결과 : 받음각과 옆 미끄럼각

방향 상태를 인지하여 현재 이동 방향 상태를 normal에서 right로 변경하게 되며, Planner는 이에 따른 left 명령을 내리게 된다. Controller는 left라는 의사명령에 따라 비행기의 옆 미끄럼각을 -5도로 결정한다. ⑤에서 r-r'값은 -10.84(48s)에서 -15.96(49s)으로 변화하며, Recognizer는 방향에 따른 상태정보를 right에서 too_right로 변경한다, Planner는 이에 따라 too_left 명령을 내리게 되고, Controller는 too_left라는 의사명령에 따라 비행기의 옆 미끄럼각을 -7도로 설정한다.

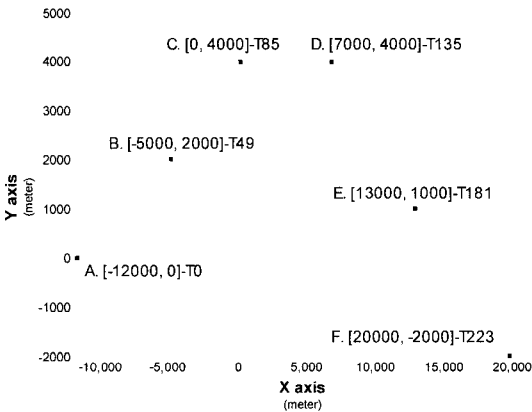


그림 5. 시물레이션 결과: 좌표

그림 5는 제안된 UAV 비행 시물레이터의 시간에 따른 좌표변화를 나타낸 것이다, T는 각 위치에 따른 시간을 나타낸다. 위 실험을 통해, UAV 비행시물레이터가 PILOT 이산사건 모델 및 PILOT 연속시간 모델간의 연동에 의해 성공적으로 동작하는 것을 볼 수 있다.

5. 결론 및 향후연구

본 논문에서는 UAV의 효과적인 제어를 위하여, 연속 시간 시스템으로 표현되는 비행기 모델과 이산사건시스템으로 표현되는 조종사 모델간의 연동 시스템을 제안하였다. 상태 정보에 의한 의사결정과 그에 따른 비행제어 시스템은 환경의 변화에 따른 지능적인 비행 제어를 가능하게 한다^{13,131}. 또한, 제안된 시스템은 DEVS 시물레이터 환경을 통해 실험 및 검증을 수행하였다. 향후 연구로, UAV의 효과적이고 능동적인 제어를 위한 의사결정기능에 대한 연구와 함께 UAV의 상태변화에 따른 이상상태 진단, 복구에 대한 연구가 필요하다.

6. 참고문헌

1. R A Hyde, "FOSTERING INNOVATION IN DESIGN AND REDUCING IMPLEMENTATION COSTS BY USING GRAPHICAL TOOLS FOR FUNCTIONAL SPECIFICATION", AIAA Modeling and Simulation Technologies Conference and Exhibit 5-8, August 2002.
2. Tasic, Predrag, Gul Agha. "Understanding and Modeling Agent Autonomy in Dynamic Multi-Agent, Multi-Task Environments," Proc. First European Workshop on Multi-Agent Systems (EUMAS '03) (Day 2, Panel 2 section), Oxford, England (UK), Dec. 18-19, 2003.
3. Chi, S.D. J.K. Lee; J.S. Lee. 1998. "Model-based Design for Autonomous Defense Systems", In Proceedings of the Military, Government and Aerospace Simulation part of the 1998 Advanced Simulation Technologies Conference(Boston, MA. April 5-9), Vol 30. pp. 61-65.
4. Yong-Jun You, Jang-Se Lee, Sung-Do Chi, SIMVA : A tool for the Network Vulnerability Analysis, PROCEEDING of International Conference on Internet Information Retrieval 2003, Korea, Oct. 2003.
5. 지승도, 박종서, 이장세, 김환국, 정기찬, 정정래, "SES/MB 프레임워크를 이용한 네트워크 보안 모델링 및 시물레이션", 한국통신정보보호학회 2001 제11회 제2호 p.15-16.
6. 임성용, 김탁곤, "하이브리드 시스템 모델링 및 시물레이션 - 제 1 부: 모델링 및 시물레이션 방법론", 한국시물레이션학회 논문지, 제 10 권 3 호, 2001년 9 월, p.1-14.
7. 임성용, 김탁곤, "하이브리드 시스템 모델링 및 시물레이션 - 제 2 부: 시물레이터 연동 환경", 한국시물레이션학회 논문지, 제 10 권 3 호, 2001년 9 월, p.15-29.
8. Steven B.Hall, P. Zeigler, "Joint MEASURE : Distributed Simulation Issues In a Mission Effectiveness Analytic Simulator", 1999 Fall Simulation Interoperability Workshop. September 12-17, 1999.
9. <http://www.held-mueller.de/JMatLink/>
10. "Aerospace Blockset". The MathWorks, Inc., 3 Apple Hill Drive, Natick, MA 01760-2098, April 2002.
11. Jackson E. B., Cruz C. L., "Preliminary Subsonic Aerodynamic Model for Simulation Studies of the HL-20 Lifting Body", NASA TM4302, August 1992.
12. Zeigler, B.P. Multifaceted Modeling and Discrete Event Simulation, Academic Press, 1984.
13. S.D. Chi, Modeling and Simulation for High Autonomy Systems, Ph.D. Dissertation, Dept. of Electrical and Computer Engineering, Univ. of Arizona, 1991.



강 광 천 (kkckc@korea.com)

2004 한국항공대학교 컴퓨터공학과 학사
2006 한국항공대학교 컴퓨터공학과 석사
2006~현재 네이블 커뮤니케이션즈 연구원으로 근무

관심분야 : 지능시스템, 시뮬레이션, 패턴인식



유 용 준 (ilog21c@hau.ac.kr)

2003 한국항공대학교 컴퓨터공학과 학사
2005 한국항공대학교 컴퓨터공학과 석사
2005~ 현재 한국항공대학교 컴퓨터공학과 박사 과정

관심분야 : 모델링 및 시뮬레이션, 네트워크 보안



지 승 도 (sdchi@hau.ac.kr)

1982 연세대학교 전기공학과 학사
1984 연세대학교 전기공학과 석사
1985~1986 두산 컴퓨터 (현 한국 디지털) 근무
1991 미국 아리조나대학교 전기전산공학과 박사
1991~1992 미국 SIMEX Systems and S/W 회사 S/W 담당자로 근무
1992~현재 한국항공대학교 컴퓨터공학과 교수

관심분야 : 이산사건 시스템 모델링 및 시뮬레이션, 컴퓨터 보안, 지능시스템 디자인 방법론, 시뮬레이션 기반 인공생명, 교통 모델링