

X-Plane 및 MATLAB/Simulink 기반의 복수무인기 모의실험 시스템 개발

Implementation of a X-Plane and MATLAB/Simulink based Simulation System for Multiple UAVs

문 상 우, 오 은 미, 유 동 일, 심 현 철*

(Sangwoo Moon¹, Eun-Mi Oh², Dong-II You³, and David Hyunchul Shim³)

¹Department of Aerospace and Mechanical Engineering, Korea Air Force Academy

²CNS/ATM & Satellite Navigation Research Center, Korea Aerospace Research Institute

³Department of Aerospace Engineering, KAIST

Abstract: In this paper, a simulation system based on X-Plane and MATLAB/Simulink for multiple UAVs is presented. For the conceptual design of this proposed system, a hierarchical system architecture for multiple UAVs is presented. This architecture has object-oriented data structure which consists of three objects (UAV status, mission and task, and environment) and a hierarchy consisting of four layers (decision making layer, task assignment layer, path and motion planning layer, and collision avoidance layer) is also proposed. In addition, this paper shows a implementation of simulation system based on the proposed system architecture using X-Plane and MATLAB/Simulink. The result of simulation from the developed system in this paper validate capability of application for multiple UAVs in real environment.

Keywords: multiple UAVs, system architecture, X-Plane, simulation

I. 서론

무인기(UAV: Unmanned Aerial Vehicle)는 최근 십수년간 비약적으로 발달하여 적진 상공에서의 장시간 체공을 통한 정찰과 같이 유인기가 기존에 수행하던 단순하고 반복적이며 위험한 임무를 점진적으로 대체해 나가고 있다. 무인기를 비롯한 무인항공시스템(UAS: Unmanned Aerial System)의 성능이 개선되고 통신 및 운용기법의 진보가 이루어짐에 따라 무인기를 활용한 미래의 전투체계는 다수의 무인 시스템이 체계적, 계층적, 유기적으로 통합되어 정확하고 다면적인 정보를 수집하고 최소의 부수적 손실로 최대한의 성과를 거두는 것으로 바뀌게 될 것이다[1].

이를 위해 다수의 무인기를 동시에 투입하여 종합적인 정보를 동시에 수집하거나 한 대가 하기에는 너무 복잡한 임무를 효율적으로 수행하는 방향으로 연구가 진행되고 있다[2-5]. 더 나아가 무인기 관련 기술이 진보됨에 따라 이와 같은 개념을 무인기의 작전개념에도 적용, 아주 많은 수를 동시에 투입하여 주어진 임무를 수행하는 군집(swarming)의 개념도 제안되고 있다[6,7].

이러한 복수 무인기(multiple UAVs)의 운용은 단수 무인기가 하지 못했던 복합적이고 복잡한 임무에 대해서 효율적으로 임무를 수행할 수 있는 방안이며, 고가의 단수 무인기 보다 저가의 복수 무인기를 통하여 임무를 완수하는 방식을 채택함으로써 임무 수행에 따른 위험성을 줄이는 한편으로 임무 수행의 가능성을 향상시킬 수 있을 것으로 기대된다[1,2].

이처럼 복수무인기의 운용의 중요성이 부각됨에 따라, 현재 무인기를 기반으로 한 시스템의 연구가 활발하며, 이를 실제적으로 검증할 수 있을 모의실험 시스템 개발의 시도 역시 이루어지고 있다. 이미 복수의 지상로봇에 대한 모의실험 시스템의 개발은 이루어진 바 있으며, 실제 로봇의 센서와 통신환경 등을 고려하였다[8]. 복수무인기를 대상으로 하는 실시간적인 모의실험 시스템 역시 개발된 바 있으나, 실제 환경 하에서의 모의실험은 미비한 실정이다[9,10].

실제로 복수무인기를 운용하는 데 있어서 하드웨어와 소프트웨어를 지상에서 다양한 비행 시나리오에 대한 충분한 시뮬레이션을 수행하여 알고리즘의 타당성, 적합성 및 각종 장비들의 작동 이상 유무를 확인해야 한다. 각각의 시스템이 정상적으로 작동하더라도 통합된 시스템은 예기치 못했던 간섭 등으로 인하여 정상적으로 작동하지 않는 상황이 빈번하게 발생되기 때문에 매우 신중하게 이루어져야 한다. 이러한 확인 작업으로 먼저 순수 소프트웨어 시뮬레이션을 통하여 구현한 알고리즘을 확인해야 하며, 이를 평가하고 보완하는 것이다.

본 논문에서는 복수무인기의 임무수행을 효율적으로 관리하기 위한 개념설계로써 객체지향적인 데이터 구조와 이

* 책임저자(Corresponding Author)

Manuscript received February 20, 2013 / revised March 10, 2013 / accepted March 15, 2013

문상우: 공군사관학교 항공우주기계공학과(dear.sangwoo@gmail.com)

오은미: 한국항공우주연구원 교통신호기술연구소(emoh@kari.re.kr)

유동일, 심현철: KAIST 항공우주공학과

(ryu1213@kaist.ac.kr/hcshim@kaist.ac.kr)

※ 본 논문은 국방과학연구소의 국방개발기초연구사업(과제번호 09-01-03-04)에서 지원하여 연구하였음.

에 필요한 변수를 설정하고 계층적인 복수무인기 시스템 구조를 상정하였다. 이렇게 설정한 시스템 구조는 개념적이고 추상적인데, X-Plane과 MATLAB/Simulink를 이용한 모의 환경을 개발함으로써 이를 구체화하고 현실적으로 가능하게 하였다.

개발한 모의실험환경은 각각의 무인기가 직접 계산하고 결정하는 분산식 방식과 중앙관리센터가 존재하여 임무계획에 관여하는 방식으로 구분할 수 있으며, 모의실험의 검증을 위한 임무할당 알고리즘으로 협상 기반의 임무할당법을 적용하였다.

II. 객체 기반의 복수 무인시스템 데이터 구조

객체 기반의 복수 무인시스템 구조는 데이터 필드(data fields)와 함수와 같이 연산의 속성을 지니는 것들(methods)을 하나의 데이터 구조(data structure)로 구성하는 객체 기반 프로그래밍의 원리를 이용한 것이다[11]. 비록 임무에 투입되는 복수의 무인 시스템의 구조는 매우 복잡하여 이를 위한 임무 계획(mission planning)이 어려울 것으로 보이지만, 크게 네 가지의 속성을 지닌 데이터 구조로 나눌 수 있으며, 이렇게 나뉜 데이터 구조들 내부에는 시스템의 임무 수행에 필요한 모든 정보를 포함시킬 수 있다. 먼저, 무인기의 상태와 속성, 그리고 현재 자신이 맡은 임무 등을 하나의 객체로 표현이 가능하다. 무인기가 임무에 투입되는 임무 환경에 대해서 또 다른 객체로 표현할 수 있으며, 현재 각 임무(mission)에 할당된 작업(tasks)들의 상태 역시 하나의 객체로 표현할 수 있다. 마지막으로, 각각 중복될 수 있는 정보를 공유하거나 각 객체간의 통신을 담당하는 객체가 존재한다.

그림 1은 이렇게 제시한 객체 기반의 시스템 구조를 모식적으로 나타낸 것이다. 각 객체는 제시한 시스템 구조가 독립적으로 구성되어 있으며, 임무에 투입된 무인기들 중에서 일부가 임무수행에서 이탈하더라도 전체 군집시스템이 최종적으로는 임무를 수행할 수 있는 강인함을 강점으로 갖는다. 뿐만 아니라, 독립적으로 변화하는 조건들에 대해

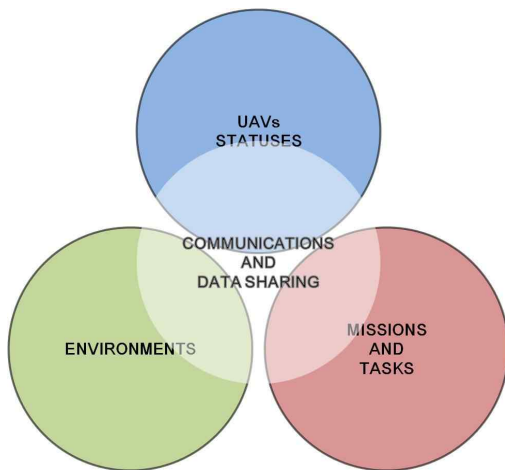


그림 1. 복수무인기 시스템 구조의 모식도.
Fig. 1. Mimetic Diagram of System Architecture for Multiple UAVs.

서 크게 종속적이지 않는 확장성(scalability)을 지니는 장점도 있다.

1. 무인기 상태(UAVs status)

임무에 투입되는 각각의 무인기는 현재의 상태(states)와 속성(statuses)을 가지고 있으며, 이 중 무인기의 상태는 임무를 수행함에 따라서 변화한다. 뿐만 아니라, 무인기가 맡고 있는 임무의 번호나 수행 상태 등의 정보 역시 이 객체에 속한다.

표 1은 무인기 상태 객체에 속해 있는 변수들이며, 무인기 상태 객체는 총 20개의 변수로 구성되어 있다. 이들 변수들 역시 크게 무인기 상태, 속성, 임무상황의 세 가지 범주로 나눌 수 있다.

2. 임무 환경(environments)

임무 환경에 대한 모든 정보는 이 객체 내부에 존재하며, 이런 정보는 임무를 수행하기 전에 주어질 수도 있으나 임무를 수행하면서 탐색기 등을 이용하여 획득할 수 있다. 따라서 임무환경 객체에 포함되는 변수들은 각 무인기가 지니기도 하지만 전체적인 정보들을 총괄하고 관리할 필요가 있다. 전체 정보는 임무 관리 센터(mission monitoring center)에 저장되며, 수집된 정보는 다시 임무 관리 센터에서 각 무인기 개체들에게 전달된다.

표 1. 무인기 상태 객체의 구성 변수.

Table 1. Variables of UAV Status Class.

U_x	무인기의 x축 위치
U_y	무인기의 y축 위치
U_z	무인기의 z축 위치
U_ϕ	무인기의 롤(roll) 각도
U_θ	무인기의 피치(pitch) 각도
U_ψ	무인기의 요(yaw) 각도
U_u	무인기의 x축 속도
U_v	무인기의 y축 속도
U_w	무인기의 z축 속도
U_p	무인기의 롤(roll)방향 각속도
U_q	무인기의 피치(pitch)방향 각속도
U_r	무인기의 요(yaw)방향 각속도
$U_{I_{xx}}$	무인기의 x축 방향 모멘트
$U_{I_{yy}}$	무인기의 y축 방향 모멘트
$U_{I_{zz}}$	무인기의 z축 방향 모멘트
U_m	무인기 중량
U_t	무인기의 종류(고정익, 회전익)
U_r	무인기의 속성(정찰, 공격, 수송)
U_M	무인기의 임무 수행 상태 (임무없음, 임무예약, 임무수행, 임무완료, 긴급상황)
U_{Tn}	무인기가 맡은 임무의 번호

표 2. 임무환경 객체의 구성 변수.

Table 2. Variables of Environment Class.

E_t	속성 (장애물, 위협요소, 무인기)
E_m	유동성 여부 (정적, 동적)
E_x	x축 위치
E_y	y축 위치
E_z	z축 위치
E_u	x축 속도
E_v	y축 속도
E_w	z축 속도
E_r	위협 반경 (range)
E_g	여유 공간 (buffer margin)

표 2는 임무 환경 객체에 속해 있는 변수들이며, 총 10개의 변수로 구성되어 있다. 이들 변수들은 크게 환경 정보 (information of environment)의 위치, 상태, 속성의 세 가지 범주로 나눌 수 있다.

3. 임무 및 작업(missions and tasks)

다수의 무인기가 투입되어 달성해야 할 최종 목적까지의 과정 전체는 하나의 임무(mission)로 정의되며, 각 과정에 필요한 세부적인 목표들이 작업(task)으로 정의된다. 임무에 대한 모든 정보는 이 객체 내부에 존재하며, 이런 정보는 임무를 수행하기 전에 주어질 수도 있으나 임무를 수행하는 도중에 사용자(user)의 요구에 의하여 변경될 수 있다. 임무환경 객체와 마찬가지로, 변수들은 각 무인기가 지니기도 하지만 전체를 총괄하고 관리할 필요가 있다. 전체 정보는 임무 관리 센터(mission management center)에 저장되며, 임무가 변경될 경우 새로운 임무에 대한 정보는 다시 임무 관리 센터에서 각 무인기 개체들에게 전달된다.

표 3은 임무 환경 객체에 속해 있는 변수들이며, 총 8개의 변수로 구성되어 있다. 이들 변수들은 크게 무인기가 수행해야 할 작업의 위치, 상태, 속성의 세 가지 범주로 나눌 수 있다. 그러므로 기본적인 범주는 임무환경 객체의 경우와 비슷하다.

표 3. 임무 및 작업 객체의 구성 변수.

Table 3. Variables of Mission and Task Class.

M_s	임무의 상태 (미예약, 예약, 수행, 완료)
M_m	속해 있는 임무 번호
M_{Ts}	작업의 상태 (미예약, 예약, 수행, 완료)
M_{Tt}	작업 종류 (정찰, 공격, 경로점, 이륙, 착륙)
M_U	작업 예약을 받은 무인기 번호
M_x	작업의 x축 위치
M_y	작업의 y축 위치
M_z	작업의 z축 위치

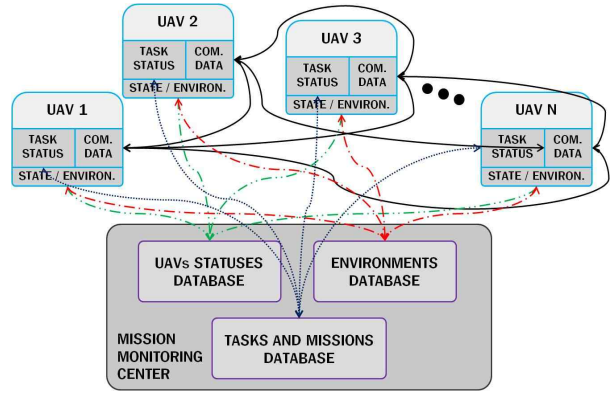


그림 2. 시스템의 통신 및 정보공유 모식도.

Fig. 2. Mimetic Diagram of Communication Architecture.

4. 통신 및 정보 공유(communications and data sharing)

비록 주어진 세 개체에 속해 있는 변수들이 서로 독립적으로 수행된다고 해도, 일부 변수들에 대해서는 중복되거나 종속적인 관계를 지니고 있다. 예를 들어, 임무 환경은 장애물과 위협요소뿐 아니라 임무를 수행 중에 있는 다른 무인기까지도 포함하는데, 이는 무인기 상태 객체에서의 변수와 상당 부분이 중복된다. 따라서, 중복되는 정보에 대한 관리가 필요하며, 이를 위해서는 각 무인기 개체간의 정보를 공유를 위한 통신이 절실히 필요하다. 더 나아가서, 이런 통신과 정보를 공유하는 일련의 과정을 관장하는 객체가 별도로 필요하다.

그림 2는 복수무인기시스템의 통신 구조로, 각 무인기는 각자 맡을 임무와 무인기 자신의 상태, 그리고 주변의 임무 환경을 앞서 제시한 세 가지 객체의 형태로 지니고 있다. 여기서 무인기의 상태 객체는 임무 관리 센터에 전달되고, 임무 환경 객체는 임무 관리 센터와 쌍방향으로 정보전달이 이루어져 무인기가 전체 임무 환경을 알 수 있게 한다. 실제로 무인기 간의 직접적인 통신을 통해서 주고받는 정보는 임무 및 작업 객체이다. 이 객체에 속해있는 정보는 실시간 연산을 위하여 최소한의 패킷(package)으로 구성되어 통신이 이루어지며, 작업 할당(task assignment)을 포함한 임무 계획(mission planning)에 관여한다. 마지막으로, 임무를 관리하고 임무에 필요한 정보들을 공유하기 위하여 임무 관리 센터가 위치한다.

III. 계층화된 복수무인기 시스템 개념 구현

그림 3은 앞 절에서 제안한 데이터 구조를 활용하여 계층화된 복수무인기 시스템의 개념을 보여주고 있다. 각각의 무인기에는 서로 계층적으로 작동하는 의사결정(decision making), 임무할당(task assignment), 경로계획(path and motion planning), 충돌회피(collision avoidance)의 네 가지 단계로 이루어진 시스템을 가지고 있다. 따라서 상위 단계에서 우선적인 결정권을 가지고 있으며, 상위 단계에서 행해지는 결정에 따라서 하위 단계가 작동하게 된다[12].

최상위 단계라 볼 수 있는 의사결정 단계에서는 무인기의 동작 상태를 감지하는 단계이며, 모든 데이터 구조의 접근이 가능하다. 의사결정 단계에서 해당 무인기가 정상적인

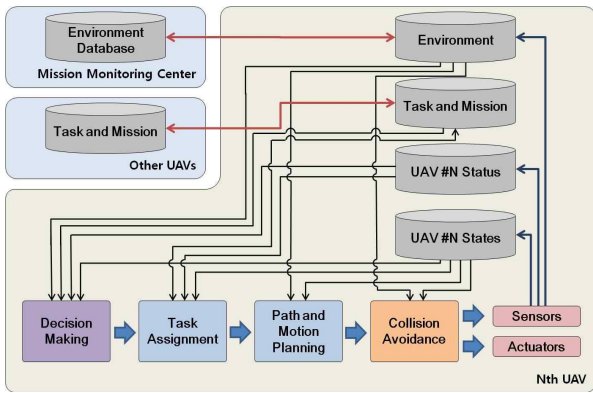


그림 3. 계층화된 복수무인기 시스템 모식도.
Fig. 3. Mimetic Diagram of Hierarchical Architecture for Multiple UAVs.

로 동작이 가능하다고 결정하면 임무를 수행하기 위한 임무할당 단계가 작동한다. 임무할당 단계에서는 해당 무인기의 상태 객체에 접근하여 최적의 임무를 설정하는데, 이 때 다른 무인기 간의 정보교환이 이루어져야 하므로 임무 및 작업 객체에도 접근하여 최종적으로 임무를 설정하게 된다. 따라서 임무 및 작업 객체를 관장하는 데이터베이스는 임무 및 작업을 다른 무인기들과 공유한다. 최적의 임무할당을 산출하기 위하여 다음 단계에 해당하는 경로계획의 결과를 참고하게 된다. 최하위 단계에 해당하는 충돌회피는 실시간적으로 예측하지 못한 장애물을 파악하여 충돌을 피하는 궤적을 생성하게 해 주며, 이러한 정보는 앞 절에서 제안한 데이터 구조 중 임무환경 객체와 무인기 상태 객체에 축적된다. 마지막으로, 환경 객체 역시 임무를 관리 센터에 축적된 데이터와 공유를 한다. 이러한 과정은 주어진 임무를 모두 수행할 때까지 계속 이루어진다.

IV. X-Plane을 이용한 복수무인기 모의실험 환경 구성

앞 절에서 제안한 복수무인기 시스템을 실제적으로 구현하기 위하여 실제와 유사한 환경을 제공하는 소프트웨어를 채택하였고 계층적인 복수무인기 시스템을 적용한 모의실험환경을 개발하였다[13].

대부분의 복수 무인기 운용 모의실험은 비행체의 동역학이 포함되지 않고 단지 운동학모델만을 사용한다. 따라서 비행체의 동역학이 반영되어 있지 않으므로 모의실험 결과에서 높은 신뢰도를 기대하기 어렵다. 따라서 효율적인 모의실험을 위해서는, 시뮬레이션 환경 구성에 있으며, 알고리즘들을 실제 무인기에 적용하기에 앞서 안정성을 확인하기 위해 현실적인 가상 환경에서의 모의실험은 필수적이다. 항공기 시뮬레이션은 현실 세계의 물리적 법칙들을 최대한 반영해야 하는데, 이러한 높은 수준의 현실성 구현에는 매우 복잡한 모델들과 방대한 계산 량이 필요로 하므로 실제로 구현하기에는 높은 비용과 시간이 든다. 이를 해결하기 위해, 이미 존재하는 시뮬레이터를 변경하여 무인기 가상 실험을 수행할 필요가 있다.

비행 시뮬레이션을 위한 소프트웨어 중 널리 사용되는 프로그램은 Flight-Gear, Microsoft Flight Simulator, X-Plane

등 여러 프로그램 들이 있다. 이 중에서 MATLAB/Simulink 와 연동 가능한 프로그램은 Flight-Gear와 X-Plane 프로그램 이며, 본 연구에서는 항공기 상태 정보를 UDP를 통해 주고 받을 수 있고 기존의 공항 및 기상 상태 구현을 그대로 사용 가능한 X-Plane을 모의실험환경 구축을 위한 소프트웨어 로 채택하였다.

X-Plane은 미국 Laminar Research 사에서 개발한 비행 시뮬레이션 소프트웨어로서, 미연방항공국(FAA)으로부터 41 개 기종에 대해서 FTD (Flight Training Device Level 2) 인증을 받은 정교하고 사실성 높은 시뮬레이션 프로그램이다. X-Plane은 기본적으로 깃요소 해석(blade element analysis)을 사용해 여러 비행체에 대한 공기역학적인 시뮬레이션을 수행이 가능하며, 비행지역의 실제 날씨를 실시간으로 구현이 가능하다. 또한 공항의 활주로, 유도로(taxiway), ILS, VOR 등의 항공운항 제반시설도 세부적으로 묘사되어 있으며, Plane-Maker라는 내장된 프로그램을 이용해 여러 형상의 항공기를 직접 제작해보고 시뮬레이션 해볼 수 있어 신개념 무인기 항공기 시뮬레이션에 적합한 여러 가지 기능들을 제공한다.

X-Plane은 UDP 통신을 통해 항공기 상태 정보를 실시간으로 전송/수신 받을 수 있어 운용 확장성이 높으며 이를 이용해 MATLAB Simulink 프로그램을 이용한 시뮬레이션 환경 구축이 가능하다. 그림 4는 X-Plane과 MATLAB Simulink를 연동하여 비행제어 프로그램을 구성한 것으로써 Simulink 상에서 대상 항공기에 대한 오토파일럿 시스템을 설계하고 X-Plane 상에서 자동 비행 시험을 할 수 있도록 구성된 제어 및 시뮬레이션 시스템을 보이고 있다.

이와 함께 X-Plane은 여러 대의 항공기가 동시에 비행할 수 있는 멀티플레이 기능을 제공함으로써 내부적으로 제공되는 공항 데이터 및 풍경(scenery)을 이용하면 자동 이착륙, 항공기 충돌 회피, 비행 궤적 추종, 다수 항공기의 편대 비행 등의 여러 가지 알고리즘들을 구현하고 검증해 볼 수 있어 이를 이용해 다양한 가상의 환경에서의 복수 무인 항공기의

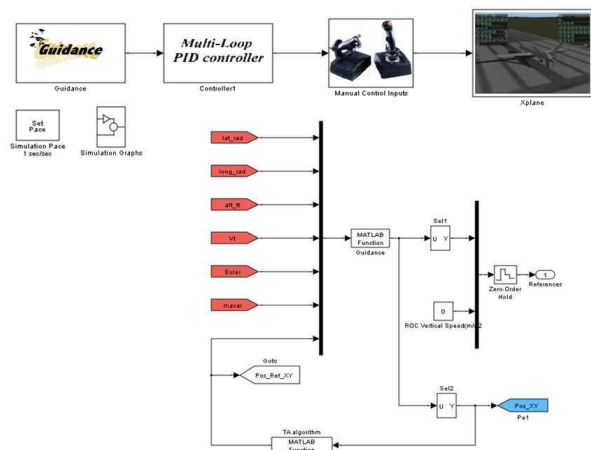


그림 4. X-Plane 및 MATLAB/Simulink 기반 오토파일럿 시스템 일부.
Fig. 4. Part of X-Plane and MATLAB/Simulink based Autopilot System.

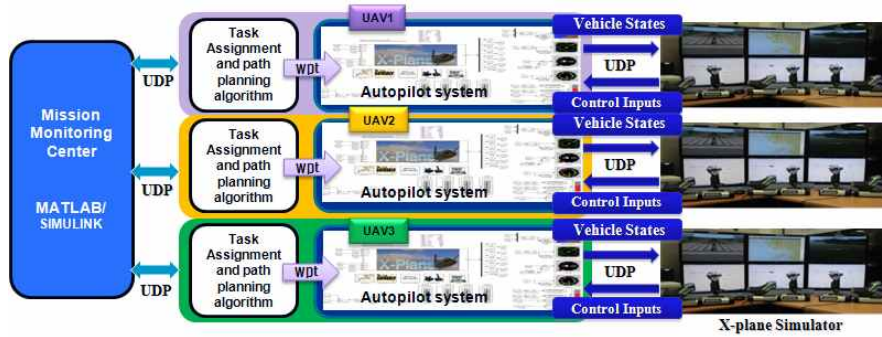


그림 5. X-Plane 및 MATLAB/Simulink 기반 분산식 복수 무인기 모의실험 시스템 구성.

Fig. 5. Distributed X-Plane and MATLAB/Simulink based Simulation System for Multiple UAVs.

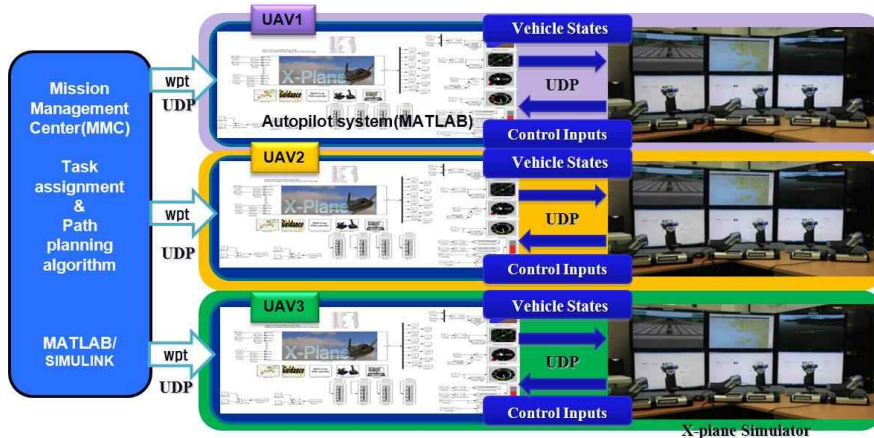


그림 6. X-Plane 및 MATLAB/Simulink 기반과 임무관리센터 처리 기반의 복수 무인기 모의실험 시스템 구성.

Fig. 6. X-Plane and MATLAB/Simulink based and Central Management Center based Simulation System for Multiple UAVs.

운용 알고리즘을 검증함으로써 연구 수행의 효율성을 높일 수 있다. 본 연구에서는 총 3대의 시뮬레이션 컴퓨터를 연결한 복수 무인 항공기 모의실험 시스템을 구성하였다.

이러한 시스템은 시뮬레이션 컴퓨터를 추가함으로써 더 많은 수의 복수 무인 항공기 시뮬레이션이 가능하다. 임무 할당 및 경로계획 알고리즘에서 계획된 비행 기동 및 비행 계획은 UDP 통신을 통하여 각각의 시뮬레이션 컴퓨터의 오토파일럿의 명령 값으로 입력된다. 이는 하나의 상위 계획단계에서 효율적으로 복수 무인기 운용을 관리가 가능하게 해 주며, 시뮬레이션 컴퓨터를 추가로 연결해 줌에 따라 쉽게 비행체를 추가가 가능하므로 확장성이 용이하다. 본 논문에서는 복수의 무인기를 운용할 수 있는 시스템을 두 가지 방법으로 구축하였다.

1. 분산식 복수 무인기 모의실험 시스템

첫 번째 구성은 그림 5와 같이 경로생성법을 포함하는 임무할당 알고리즘을 각각의 무인기가 계산하는 방법으로 기체에 탑재가 가능한 알고리즘의 경우 적용이 가능하다. 이 구성은 X-Plane 상에서 각 무인기의 위치 정보를 MATLAB/Simulink 상의 UDP 통신을 통해 임무모니터링센터로 보내게 된다. 임무모니터링센터에서는 전달받은 무인기의 정보들을 저장해 두고 모니터링(monitors)한다. 한편, 각각의 무인기는 임무할당과 경로계획을 독립적으로 계산

하고 이런 결과를 임무 관리 센터에 UDP 통신을 통해 전달된다.

이러한 UDP 통신은 임무를 수행하고 있는 무인기와 임무모니터링센터에 연결하는 통신이기도 하지만 임무모니터링센터에 매개로 하여 임무를 수행하고 있는 무인기들 간에 정보의 공유를 가능하게 하였다. 이를 바탕으로 각각의 무인기는 인가할 임무를 선정하게 되며, 선정된 임무로 가기 위해 적용된 경로생성법을 통해 가야할 지점에 대한 경로점 waypoint을 찾는다. 이 통신 구조는 최적의 임무할당을 보장하지 못하지만 통신 문제나 사고로 인하여 무인기의 일부가 임무수행이 불가능 하더라도 복수 무인 시스템 전체적으로는 임무수행에 치명적인 영향을 주지 않는 강건성을 갖는다.

2. 임무관리센터 처리 기반 복수 무인기 모의실험 시스템

두 번째 구성은 각각의 무인기의 정보를 임무관리센터에서 취합하고 복수 무인기 알고리즘을 계산하여 각 무인기에 명령을 전달하는 방식이다. 무인기에 탑재되는 비행 제어 컴퓨터(FCC)와 복수 무인기 알고리즘 프로그램 간의 호환이 어려운 경우 이러한 구성을 통해 필요 정보만을 전달하는 방식으로 알고리즘 검증이 가능하다. 이 구성은 X-Plane 상에서 각각의 무인기의 위치 정보를 MATLAB/Simulink 상의 UDP 통신을 통해 임무관리센터로 보내게 된

다. 임무관리센터에서는 전달받은 무인기들의 정보 취합하여 이를 바탕으로 각각 무인기에 인가할 임무를 선정하게 되며, 선정된 임무로 가기 위해 적용된 경로생성법을 통해 가야할 지점에 대한 경로점(waypoint)을 찾게 된다. 주어진 경로점으로 가기 위한 대상 무인기의 오토파일럿 시스템을 구성하였으며 UDP 통신을 통해 X-Plane 상의 조종면 입력으로 전달되도록 하였다. 이 통신 구조는 임무할당에 필요한 취합 정보를 가지고 임무관리센터에서 각 무인기의 임무 및 경로를 계산이 가능하며 통신 문제로 인해 모든 무인기의 정보가 취합되지 않은 경우에도 가지고 있는 정보만을 가지고도 임무 수행이 가능하다.

V. 모의실험 수행결과 및 분석

본 시뮬레이션은 실제 실험에 적용하기 전에 이상적인 UDP 통신 환경에서의 복수무인기 시스템에 적용된 알고리즘의 실시간성을 확인하고 알고리즘과 모의 환경 간의 적합성을 파악하는데 주된 목적이 있다. 즉, UDP 통신을 통해 전송되는 데이터의 양과 전송 주기, 알고리즘의 실행 주기에 따라 시뮬레이션과 무인기 자동 비행 상에 어떤 문제가 발생하는지, 모의 환경의 규모와 임무의 개수, 위치 등이 무인기 성능에 따라 어떤 결과 양상을 보이는지 확인할 수 있다.

임의의 임무를 가지는 시뮬레이션을 선정하여 알고리즘을 검증하였다. 여기서 사용된 알고리즘은 협상 기반 알고리즘으로 알고리즘의 타당성 검증으로써 수행하였다. 환경은 다음과 같이 영종도에 위치한 인천국제공항과 강화도 주변의 실제 지형에 대한 시뮬레이션으로 가정하였다. 그림 7의 섬 지형을 장애물로 인식하여 회피를 하도록 하였으며 해상과 강화도 위에 존재하는 임의의 정찰 임무를 선정하여 각 임무 당 한 번의 정찰을 수행하고 모든 정찰 임무가 완수되면 각각 무인기의 지정된 도착 지점으로 가도록 시나리오를 정하였다.

추가로, 동적인 환경의 변화를 주어 복수 무인기 임무할당 알고리즘이 이러한 환경 변화에도 적용이 가능하다는 것을 확인하였다. 환경의 변화로 세 대의 무인기 중 한 대(가운데 2번 무인기)가 도중에 추락한다는 설정과 기존에



그림 7. 협상 기반 알고리즘을 적용한 모의실험 도식화.

Fig. 7. Mimetic Diagram of Simulation applied by Negotiation based Algorithm.

알려지지 않은 장애물(기존에 알려진 섬 지형 외 해상 장애물)이 갑작스럽게 생성되는 상황을 만들어 이러한 경우의 임무할당을 동적으로 수행하는 지에 대해 확인하였다.

그림 10은 협상 기반 임무할당 알고리즘을 적용한 모의 실험 결과이다. 가운데의 2번 무인기가 4번 임무를 수행하고 추락하는 상황에서 나머지 무인기들이 이 외의 임무를 모두 완수하고 각각의 도착지점으로 가는 것을 확인하였으며, 위와 같은 시나리오에 대해서 무인기 별로 할당된 임무가 그림 7과 같이 이루어짐을 알 수 있다.

우측의 3번 무인기가 6번 임무를 수행한 후 아직 수행이 되지 않은 3번 임무로 향하다가 좌측 1번 무인기가 2번 임무를 완수하였기 때문에 마지막으로 남은 3번 임무에 대해 3번 무인기와 경합을 벌이게 된다. 그 결과 더 가까운 거리에 있어 상대적으로 임무 수행에 적은 시간이 소비될 것이라고 판단되는 1번 무인기가 3번 임무를 할당 받게 되고 더 이상 수행할 임무가 없는 3번 무인기는 미리 지정된 자신의 도착 지점으로 선회하는 것을 볼 수 있다. 그리고 처음에 고려하지 않은 장애물에 대해서도 이를 인지한 순간부터 이를 회피하도록 경로를 생성하는 것을 확인하였다.



그림 8. 모의실험 실행화면 (X-Plane 실행화면).

Fig. 8. View of Simulation Execution (X-Plane).

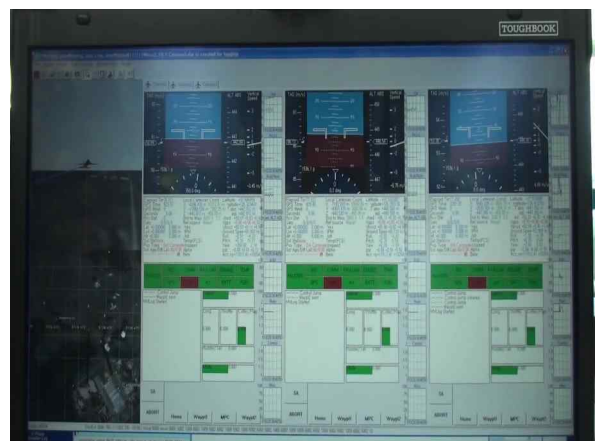


그림 9. 모의실험 실행화면 (임무모니터링센터 실행화면).

Fig. 9. View of Simulation Execution (Mission Monitoring Center).

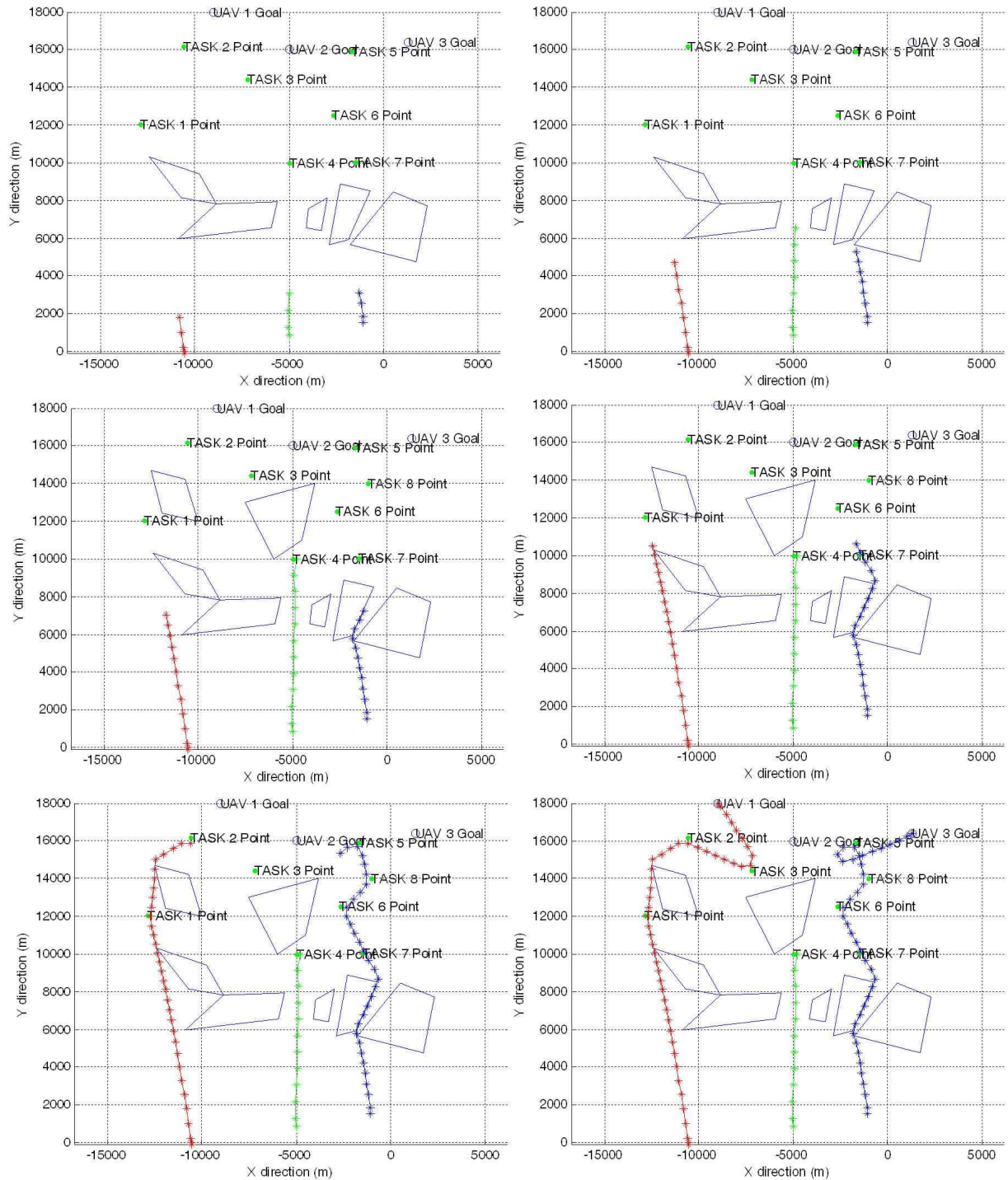


그림 10. 협상 기반 알고리즘을 적용한 모의실험 수행결과. (t = 80 sec, 160 sec, 240 sec, 380 sec, 620 sec, 900 sec).

Fig. 10. Simulation Result applied by Negotiation based Algorithm. (t = 80 sec, 160 sec, 240 sec, 380 sec, 620 sec, 900 sec).

연산시간이 높지 않고 가벼운 알고리즘인 협상 기반 임무할당을 적용하여 모의실험을 수행한 결과 실시간 수행 시간이 0.5초 이내로, 실시간 계산에도 무리가 없었다. 임무모니터링센터 기반 환경의 경우 주고받는 데이터의 크기가 UDP 통신에 큰 영향을 주지 않는 것으로 확인되었다.

특히, 무인기의 성능에 비해 빠르게 임무가 변경되는 경우나 환경에서는 시간 지연에 의한 임무 계획과 수행에 어

려움이 있을 수 있을 수 있으므로 위와 같이 환경을 모의하여 실시간 적용 가능성을 판단할 필요가 있다.

본 시뮬레이션 결과를 통해 제시한 복수무인기 시스템이 동적인 환경 변화에서도 시나리오에 따라 원활한 수행을 할 수 있음을 확인하였으며 알고리즘 계산 시간에 따른 지연 문제가 발생하지 않음을 확인하였다. 이를 바탕으로 실제 비행에서도 제안한 복수무인기 시스템이 적용가능하다는 것을 판단할 수 있었다.

VI. 결론

본 논문에서는 객체지향적인 데이터 구조를 갖는 계층화된 복수무인기 시스템을 제안하였다. 객체지향적인 데이터의 구조는 무인기 상태, 임무 및 작업, 임무 환경으로 객체화하였고, 각 무인기와 임무 관리 센터간의 통신 및 정보공유 방법을 계층적인 단계를 설정해 주어 추상화하였다.

이렇게 추상화한 복수무인기 시스템을 구현하기 위하여 복수무인기 시스템을 검증하기 위한 시나리오를 상정하였으며, 모의실험을 위한 시스템을 구축하였다. 구축한 시스템은 X-Plane과 MATLAB/Simulink를 연동한 통합적인 운용체계로서, 각 소프트웨어가 지니고 있는 장점을 결합하여 실제 실험 환경과 거의 동일한 조건에서 개발한 기법들의 가능성과 성능을 타진할 수 있음을 모의실험 검증을 통해서 확인하였다. 본 연구를 통해서 복수 단위의 무인기에 대한 유기적인 운용 기술을 개발하고, 무인기 간의 통합 운용 기법 개발에 응용할 수 있을 것으로 기대된다.

참고문헌

- [1] Office of the Secretary of Defense, "Unmanned Aircraft Systems (UAS) Roadmap 2005-2030," U.S., Report, 2005.
- [2] T. Schouwenaars, B. D. Moor, E. Feron, and J. How, "Mixed integer programming for multi-vehicle path planning," *Proc. of European Control Conference 2001*, Porto, Portugal, pp. 2603-2608, Sep. 2001.
- [3] A. Richards, Y. Kuwata, and J. How, "Experimental demonstrations of real-time MILP control," *Proc. of AIAA Guidance, Navigation, and Control Conference and Exhibit 2003*, Austin, U.S., 2003.
- [4] P. B. Sujit, A. Sinha, and D. Ghose, "Team, game, and negotiation based intelligent autonomous UAV task allocation for wide area applications," *Studies in Computational Intelligence*, vol. 70, pp. 39-75, 2007.
- [5] A. Viguria, I. Maza, and A. Ollero, "Distributed service-based cooperation in aerial/ground robot teams applied to fire detection and extinguishing missions," *Advanced Robotics*, vol. 24, pp. 1-23, 2010.
- [6] L. Pimenta, N. Michael, R. C. Mesquita, G. Pereira, and V. Kumar, "Control of swarms based on hydrodynamics models," *Proc. of International Conf. on Robotics and Automation*, Pasadena, U.S., pp. 1948-1953, May 2008.
- [7] P. Cheng and V. Kumar, "A almost communication-less approach to task allocation for multiple unmanned aerial vehicles," *Proc. of International Conf. on Robotics and Automation*, Pasadena, U.S., pp. 1384-1389, May 2008.
- [8] R. T. Vaughan, "Stage: a multiple robot simulator," University of Southern California, U.S., Report, 2000.
- [9] A. Goktogan, E. Nettleton, M. Ridley, and S. Sukkarieh, "Real time Multi-UAV simulator," *Proc. of International Conference on Robotics and Automation*, vol. 2, pp. 2720-2726, Taipei, Taiwan, Sep. 2003.
- [10] S. Kim, S. Cho, S. B. Cho, and C. B. Park, "Develop-

ment of operation network system and processor in the loop simulation for swarm flight of small UAVs," *Journal of Institute of Control, Robotics and Systems (in Korean)*, vol. 18, no. 5, pp. 433-438, 2012.

- [11] S. Moon, E. Oh, and D. H. Shim, "Study on an object-oriented swarm intelligence system for multiple UAVs," *Proc. of Korean Society of Aeronautical & Space Sciences Fall Conference 2010 (in Korean)*, pp. 371-374, Jeju, Korea, Nov. 2010.
- [12] S. Moon, E. Oh, and D. H. Shim, "An integral framework of task assignment and path planning for multiple unmanned aerial vehicles in dynamic environments," *Proc. of International Conference on Unmanned Aerial Systems 2012*, Pennsylvania, U.S., Jun. 2012.
- [13] S. Moon, E. Oh, D. You, and D. H. Shim, "An integrated task assignment and path planning method and a simulation-based verification using X-plane for multiple unmanned aerial vehicles," *Proc. of Korean Society of Aeronautical & Space Sciences Fall Conference 2011 (in Korean)*, Pyeongchang, Korea, pp. 1047-1051, Nov. 2011.



문 상 우

2009년 카이스트 항공우주공학과 졸업. 2011년 동 대학원 석사. 2011년~현재 공군사관학교 항공우주기계공학과 강사. 관심분야는 복수무인기의 군집 시스템 및 임무할당, 무인기 경로계획법.



오 은 미

2010년 부산대학교 항공우주공학과 졸업. 2012년 카이스트 항공우주공학과 석사. 2012년~현재 한국항공우주연구원 교통항법기술연구센터 연구원. 관심분야는 CNS/ATM.



유 동 일

2007년 인하대학교 항공우주공학과 졸업. 2010년 카이스트 항공우주공학과 석사. 2010년~현재 동 대학원 박사과정. 관심분야는 무인전투기 자율 교전, 무인기 자동이착륙 유도 및 제어시스템.



심 현 철

1991년 서울대학교 기계설계학과 졸업. 1993년 동 대학원 석사. 2000년 University of California, Berkeley Mechanical Engineering 공학박사. 2007년~현재 KAIST 항공우주공학과 부교수. 관심분야는 무인항공기 및 제어.