

항공교통관제체계의 시뮬레이션에 대한 연구

윤 석 준*

〈 목 차 〉

- | | |
|----------------------|------------------------|
| I. 서 론 | V. 연구용 시뮬레이터 개발을 위한 기반 |
| II. 항공교통관제 | 연구 계획 |
| III. 관련 연구 사례 | VI. 향후 계획 |
| IV. 항공교통관제 훈련용 시뮬레이터 | |

요 약

당 연구의 궁극적인 목적은 미국 FAA(Federal Aviation Administration)의 NSC(National Simulation Capability) 프로그램과 같이 공항과 항공관제(Air Traffic Control: ATC) 체계의 용량과 안정성을 분석하여 공항주변에서의 항공 안전사고에 대한 근본적인 해결방안을 수립하는 것이다. 지역적으로 분산된 비행 시뮬레이터들과 컴퓨터가 생성한 항공기 object들이 ATC 시뮬레이터와 연동되면 가상합성공간이 창출되고, 이러한 가상공간 내에서 항공안전에 대한 체계적이고 종합적인 연구가 수행될 수 있다. 당 연구과제의 주제는 항공관제체계와 상용항공기들의 시뮬레이션, 분산처리 시뮬레이션 등을 포함한다. 이러한 연구 주제들은 항공 안전문제를 근본적으로 해결하기 위하여 요구되는 연구개발용 항공관제 시뮬레이터를 개발하기 위한 기반을 제공할 것이다. 당 연구에서는 항공 안전문제를 보다

* 세종대학교 항공우주공학과 부교수

본질적으로 접근하기 위한 관련자료의 조사와 분석, 비행운동의 모델링, 관제사의 기능과 작업의 분석과 모델링, 항공관제 시뮬레이션 시스템의 개념설계 등이 수행된다. 또한, PC를 기반으로 한 연구개발용 항공관제 시뮬레이션 시스템의 원형이 개발되고, 미 국방성의 High Level Architecture가 또한 항공관제 시뮬레이션에 적용된다.

I. 서론

지금까지의 각종 항공 운항사고들을 분석해 보면, 대부분의 경우 공항 부근지역의 출발/접근 구역에서 이착륙 중에 발생하고 있음이 드러난다. 더욱이 그 사고들은 항공기 자체의 결함에 의한다고 하기보다 주로 인재에 기인함을 알게 된다. 항공교통관제(Air Traffic Control: ATC)는 바로 이러한 운항 안전사고들을 방지하기 위하여 지상이나 공중에서 이동하는 모든 항공기에 대하여 지상이동 유도단계, 이착륙단계, 항로비행단계 등 출발 공항에서 목적 공항에 착륙하여 엔진작동을 정지할 때까지 모든 비행단계를 적절하게 통제하고 비행상황을 철저히 감시하며, 아울러 공항 및 항로상의 기상상태, 활주로를 비롯한 공항 항공시설의 운용상태, 각종 항공 보안시설의 운용상태, 비행공역의 통제상황 등 비행하는데 필요한 각종 항공정보를 적시에 정확하게 지원하는 것을 주임무로 한다. 본 논문은 기존 국내 공항 항공교통관제의 업무, 절차, 처리능력 등을 분석하고 안전성을 평가하기 위한 항공교통관제 시뮬레이터의 설계에 대하여 다루고 있다. 이를 위하여 우선 항공교통관제에 사용되는 기본용어들을 살펴보고, 관제업무 내용과 절차[1]에 대하여 간략히 설명한다. 또한, 이 분야에서 선도적인 연구를 수행하고 있는 미국 NASA, Ames의 관련 연구 사례[2, 3]와 국내에 도입되어 사용되고 있는 관제사 교육용 시뮬레이터의 구성을 살펴본다. 마지막으로 항공 운항 안정성평가와 연구를 위한 항공교통관제 시뮬레이터의 설계를 위한 기반연구계획을 소개한다.

II. 항공교통관제

가. 주요 관제용어 해설

항공교통관제를 이해하기 위하여 ICAO에서 정의하는 전문용어[1]들을 소개한다. 단, 그 적용범위를 국내로 국한한다.

■ 비행정보구역(flight Information Region: FIR): 국제민간항공기구(ICAO)에 의해 가맹 국가별로 할당된 하늘의 구역으로서 비행정보업무 및 경보업무를 제공하는 일정한 범위의 공역이다. 한국의 관할구역 명칭은 “대구비행정보구역(TAEGU FIR)”이며 수평범위는 그림1과 같다. 수직범위는 지표 또는 수면으로부터 무한대까지이다.

■ 관제구(Control Area: CTA): 항공로 지표 또는 수면으로부터 200m(700ft) 이상 높이의 공역으로서 항공로 상을 항행하는 항공기의 지속적인 안전과 질서있는 항행을 확보하고자 관제를 실시하기 위하여 건설교통부 장관이 지정한 공역을 말한다. 그림1에서와 같이 국내 수평/수직 관할 범위는 대구 FIR과 일치하며, 22,000ft를 기준으로 하여 “대구 저고도 관제구역”(700ft ~ 22,000ft)과 “대구 고고도 관제구역”(23,000ft ~ 무한대)으로 구분하여 대구 저고도 관제구역은 동부, 서부, 중부, 남부 등 4개의 sector로 대구 고고도 관제구역은 동부 및 서부 등 2개의 sector로 분할하여 대구 항공교통관제소(TAEGU Air Traffic Control Center: 대구 ACC)에서 운영하고 있다.

■ 접근관제구역(Terminal Control Area: TMA): 관제구(CTA)의 일부분에 속하는 구역으로서 비행장에 이착륙하는 계기비행 항공기의 체공, 계기 접근과 출발 시의 상승 및 강하비행에 필요한 공역을 말한다. 한국의 TMA는 14개소(서울, 제주, 포항, 원주, 강릉, 예천, 대구, 광주, 사천, 김해, 중원, 해미, 오산, 군산)로 분할되어 있으며, 대구 ACC로부터 사용 관제구역, 관제 상한고도 및 관제업무의 범위 등을 협정에 의거 위임받아 관할 접근관제소(Approach Control Office: APP)에서 운영하고 있다. 수평/수직범위는 관할하는 비행장의 위치, 항공교통량 및 항공기의 종류 등에 따라 달라지며, 한국 APP의 경우 3,500ft~ 22,000ft까지의 상한고도를 운영하고 있다.

■ 관제권(Control Zone: CTR): 비행장 및 그 주변의 공역으로서 항공교통의 안전을 위하여 건설교통부 장관이 지정한 공역이다. 계기비행을 위한 이착륙 절차가 수립되어 있고 관할하는 항공교통 관제기관이 있어야 하며 유자격자에 의한 정시 기상관측 및 특별 기상관측이 실시되고 있는 비행장에 대하여 지정하며, 관제탑(Aerodrome Control Tower: TWR)에서 관장한다. 수평범위는 비행장 중심점으로부터 반경 5NM(9.3Km)의 범위 내에 설정하며, 수직범위는 각 비행장의 설정에 따라 하부한계 및 상부한계의 고도를 상이하게 설정한다.

■ **항공로(ATS Route)**: 항공기의 비행에 적합하도록 전방향 표지시설(VOR) 등 지상에 설치된 항공보안 무선시설의 전파를 이용하거나, 특정 지점을 서로 연결하여 설정한 회랑 형태의 공중통로를 “항공로”라 하며 국내 항공로는 필요에 따라 각 국가별로 설정하고 있으나, 국제 항공로의 경우 모든 인접 국가와 동지역을 비행하는 항공사가 소속되어 있는 국가들의 동의를 받아 국제 민간항공기구의 항공항행계획(Air Navigation Plan: ANP)에 수록한 후 해당 국가와 협의하여 설정하며, 그 폭은 일반적으로 육상의 경우 10NM(18Km), 해상은 50NM(90Km)로 구성되어 있다. 한국의 항공로는 관악산에 위치한 안양 무선표지소(SEL VORTAC)를 기점으로 동쪽, 서쪽, 남동쪽, 남서쪽 방향 등으로 설정되어 있으며, 대구 ACC에서 관장하고 있다.

■ **시계비행규칙(Visual Flight Rules: VFR)**: 조종사의 육안 식별에 의하여 모든 장애물과 지형, 지물 등을 회피하면서 일정 고도를 유지하여 비행하는 방식이다. 항공법 상 최저 안전고도는 지상 장애물로부터 150m ~ 300m 이상을 유지하도록 규정하고 있으며, 시계비행을 위해서는 비행계획서를 사전에 제출하여야 하며 관제권(CTR) 또는 관제구(CTA) 등 교통상황이 복잡한 지역에서는 관제기관의 지시에 따라 비행하여야 한다.

■ **계기비행규칙(Instrument Flight Rules: IFR)**: 외계를 전혀 보지 않고 항공기에 장착된 계기만을 보고 비행하는 방식이다. 항로 및 공항 주변을 비행하는 항공기를 관제하기 위하여 정해진 비행규칙으로서 전적으로 관제기관의 관제업무를 제공받아 비행하므로 안전성 확보가 수월하여 모든 민간 항공기는 이 방식을 따르도록 규정하고 있다. 이 방식으로 비행하기 위해서는 정밀 고도계, 외기 온도계, 자이로 선회계, 전후좌우지시계, 방향지시계, 속도계, 승강계, 식별 트랜스폰더 등 항공기 내에 각종 정밀 장비가 장착되어야 할 뿐 아니라, 비행장에는 계기착륙시설(Instrument Landing System: ILS) 및 각종 등화시설 등이 설치되어 있어야 한다. 계기비행은 이론적으로 기상상태에 영향을 받지 않고 전천후 비행이 가능하다.

■ **계기착륙장치(Instrument Landing System: ILS)**: 계기 비행하는 항공기에 계 착륙 접근 비행로를 설정하여 위치를 지시해 주는 무선 표시 Beacon 방법으로 운용되는 NDB, VOR, TACAN 등과 착륙 활주로 중심선의 방위각, 활주로 접지 활공각도 및 활주로 끝으로부터의 거리 등을 지시해 주는 무선 송신장치의 방법으로 운용되는 ILS 등으로 계기착륙 유도장치를 구별할 수 있으며, 최종 접근비행

과정에서 항공기를 활주로까지 유도하는 착륙 지원장치를 말한다. NDB, VOR, TACA 등을 비정밀 계기착륙장치, ILS를 정밀 계기착륙장치라 하며, ILS를 구성하는 장비로서는 Localizer(방위각), Glide Slope(활공각), Marker(위치 표시) 등이 있다(그림2 참조).

나. 항공교통관제업무

항공법 제70조(항공교통의 지시) 제1항의 규정에 의거 관제권 또는 관제구에서의 이륙, 착륙의 순서 및 시기와 비행방법 등에 관한 지시와 항공기 간의 충돌 방지, 항공기와 장애물 간의 충돌 방지, 항공교통의 촉진 및 질서 유지를 위하여 행하는 일체의 업무를 말하며, 행정권한 위임 규정에 의거 지방 항공청장이 시행하고 있다. 관제업무는 다음과 같이 비행장 관제업무, 접근 관제업무, 항로 관제업무 등 세 가지로 구별된다.

■ 비행장 관제업무(Aerodrome Control Services): 비행장 내의 이동지역 및 비행장 주위에서 운항하는 항공기와 당해 비행장의 업무에 종사하는 자에 대하여 행하는 업무로서 관제탑(Aerodrome Control Tower: TWR)에서 담당하며 주요 업무는 다음과 같다.

- 비행장에서 이착륙하는 모든 항공기에 대하여 이착륙 순서를 지정해 주고 이륙 및 착륙 허가를 발부한다.
- 당해 비행장 및 그 주변 공역에서 비행하는 모든 항공기 간의 분리를 유지시켜 공중충돌을 방지한다.
- 항로, 비행고도, 레이더 식별코드 등을 포함한 항로비행 허가서를 출발 항공기에게 중계한다.
- 항공기상, 활주로 상태, 접근절차 등 비행정보를 이착륙하는 항공기에게 제공한다.
- 활주로, 유도로 등 공항 특정지역 내의 항공기, 차량 및 인원의 이동을 통제한다.

관제탑 내 각 근무 좌석별 담당업무는 다음과 같다.

- 국지 관제석(Local Control Position): 모든 항공기의 이착륙 허가 및 통과 항공기는 물론, 착륙대 내의 인원, 차량 및 장비 등을 통제한다. 또한, 레이더 조연 업무를 제공한다.
- 지상 관제석(Ground Control Position): 지상 이동 항공기에 대한 유도 허가 및 통제업무를 수행한다. 또한, 이동지역 내의 인원, 차량, 장비 등을 통제한다.
- 계기비행 허가 중계석(Clearance Delivery Position): 계기비행 허가 중계와 엔진시동 시간, 표준계기 출발절차, 항로상의 비행 제한사항 등을 발부한다.
- 비행정보 관제석(Flight Data Position): 비행 계획서와 계기비행 허가서를 접수하고 통보하며, 항공기상 및 비행정보를 수집하고 전파한다. 공항 자동정보 방송장치(ATIS)를 운용한다.

■ 접근관제업무(Approach Control Services): 관제권 또는 접근관제구역에서 비행하는 계기비행 항공기 및 이들 공역 내의 특별관제구역에서 비행하는 항공기에 대하여 행하는 업무로서 비행장 관제업무 외의 관제업무를 말하여 접근관제소(Approach Control Office: APP)에서 담당한다. 지역 여건 및 공역 운영 실정에 따라 1개 또는 다수의 공항을 관할하기도 하며 주요 담당업무는 다음과 같다.

- 항로관제소(ACC)로부터 인수한 입항 항공기를 레이더로 유도, 분리, 감시하여 목적 공항의 관제탑에 인계한다.
- 공항에서 출발한 항공기를 관제탑으로부터 인수받아 레이더로 유도, 분리, 감시하여 항로에 진입시킨 후 항로관제소 또는 인접 접근관제소에 인계한다.
- 시계 및 계기 비행 항공기에 대한 비행감시와 교통조연 업무를 수행한다.
- 목적 공항의 기상, 사용 활주로, 접근 종류, 활주로 상태 등의 비행정보를 제공한다.

우리 나라의 경우, 전국 14개소 APP의 대부분이 공항 감시레이더(ASR) 이용한 레이더 관제를 수행하고 있으며 APP 내의 각 근무 좌석별 담당업무는 다음과 같다.

- 접근 관제석(Approach Control Position): 관할구역 내로 진입하는 계기비행 항공기와 체공 및 특수상황 항공기들을 관제한다. 또한, 비행금지 및 제한구역을 레이더로 감시한다.

- 착륙 관제석(Arrival Control Position): 최종 접근로를 포함한 착륙도중의 항공기와 접근에 실패한 항공기들을 관제한다.
- 출발 관제석 (Departure Control Position): 표준계기로 출발하는 항공기를 관제하며, 고도 제한사항에 대한 준수 여부를 감시하고 접근 항공기와의 충돌을 방지한다.
- 비행정보 관제석 (Flight Data Position): 비행진행 노트 작성, 인접 관제기관과의 협조 및 주파수 배정, 각종 비행정보의 수집 및 전파 등의 업무를 수행한다.

■ 항로 관제업무(Area Control Services): 관제구 중 접근관제구역을 제외한 기타 지역 및 항로에서 비행하는 모든 계기비행 항공기에 대한 관제업무를 제공하며, 관할 FIR 내외로 입출항하는 항공기에 대한 비행정보 제공 및 수색, 구조를 위한 경보업무를 수행한다. 항로 관제소(Area Control Center: ACC 또는 ARTCC)에서 담당하며 주요업무는 다음과 같다.

- 목적 공항까지의 사용항로, 순항고도, 식별코드 배정 등 항로 비행허가를 항공기에게 제공한다.
- 출발 공항의 관할 접근관제소로부터 인수한 항공기를 레이더로 유도, 분리, 감시하여 목적 공항의 관할 접근관제소 또는 인접 국내의 항로관제소에 인계한다.
- 항로 상에서의 항공기 간 공중충돌 방지를 위하여 레이더로 분리유지 및 감시 업무를 수행한다.
- 목적 공항의 항공기상, 활주로 상태, 사용 활주로, 접근종류 등의 비행정보를 제공한다.

다. 비행방식별 항공기 관제절차

출발지 공항에서 목적지 공항까지 비행하는 항공기에게 제공되는 필요한 관제절차 및 비행정보 교환절차는 시계비행 방식과 계기비행 방식으로 구별된다. 본 논문에서는 계기비행 방식에 대하여만 소개하며 그림3과 4에 이착륙 시의 관제절차가 도시되어 있다.

Ⅲ. 관련 연구 사례

항공교통관제에 영향을 미치는 요소들을 정량적으로 파악하고, 효과적인 관제절차를 개발하기 위한 노력들은 미국[2,3], 캐나다[4,5,6], 네덜란드[7] 등을 중심

으로 근래에 들어와 활발히 시도되고 있다. 본 논문에서는 이들 중 가장 선도적인 위치에 있는 미국 NASA의 관련 연구 프로그램을 소개한다.

미국의 NASA, Ames Research Center는 FAA와 공동으로 'Surface Development and Test Facility (SDTF)'라 명명된 항공교통관제 연구용 시뮬레이터를 개발 중에 있는데, 미국에서도 가장 크고 바쁜 항공교통관제 시뮬레이터를 그대로 옮겨놓은 정도의 규모와 기능을 자랑하고 있다. 그 용도는 미래의 항공교통관제와 공항 내 지상교통에 대한 연구, 개발, 통합, 시험, 인증, 설치 등을 위한 하나의 시험대를 제공하는 것이다. 예를 들어 다음과 같은 사항들을 연구할 수 있다.

- 공항과 항공관제체계의 운항관제처리 용량 평가
- 기존 국내 공항의 관제절차 평가를 통한 개선방안 제시
- 최신 자동관제체계(TRACON: Terminal Radar Approach Control) 도입에 따른 연구
 - EDP(ExDeparture Path) tool의 사용시 관제체계 효율성을 극대화하기 위한 관제절차의 개선에 대한 연구
 - 공항 지상교통 유동을 극대화하기 위한 SMA(Surface Movement Advisor)의 효과적인 사용법 연구
- 항공관제체계 시뮬레이션 환경과 비행훈련용 시뮬레이터들을 분산 대화형 통신망을 통하여 연동하여 현실감이 충실한 가상합성환경을 창출함으로써 조종사와 관제요원(dispatcher, agents, controller 등)들에 대한 사실적인 훈련환경 제공
- 항공관제탑의 컴퓨터-인간 인터페이스에 대한 인간공학적 연구
 - 그래픽, 문자, 레이더 정보에 대한 GUI 환경 설계원칙에 대한 연구
 - 관제사의 작업량 평가
 - 현 관제절차에 대한 이해
 - 항공 안전사고에 대한 원인규명 및 평가
 - 공항 주변의 소음, 배기(Emission) 등 환경 평가

SDTF는 다음과 같은 요소들로 구성된다. (그림5 참조)

■ 관제탑의 물리적 환경

- 관제탑의 360° 창 밖 시계
- 24ft 반경의 관제탑 실내공간
- 시뮬레이션 대상 관제탑에 적합하게 재배치가 가능한 콘솔 (Console)
- 관제탑 내의 관제사들과 관제 지원석들을 위해 요구되는 장비 일체
- 레이더 디스플레이
- 지대공, 공대지, 지대지의 라디오, 전화, 인터컴 등을 구현하는 디지털 통신 시스템

■ 공항 교통 시뮬레이션

- 공항, 지형, 건물 등에 대한 사진 수준의 3차원 그래픽 모델
- 항공기, 헬리콥터, 트럭, 응급차량, 정비차량 등의 이동표적에 대한 사실적 3차원 그래픽 모델
- 공항 운용의 현재와 과거 데이터에 기초한 시나리오
- 돌발상황을 삽입할 수 있는 일상 운용의 시뮬레이션

■ 환경 요소

- 국립 기상청에서 입수한 공항 부근의 기상기록
- 낮/일몰/밤, 음영, 기상 등의 환경 효과
- 항공기 도장, 빛, 환경적 소음

■ 공항 이용자

- 항공교통 관제사
- 공항 운용자 (Local, Ground, Flight Data Clearance, Supervisors등)
- 항공사 (램프 매니저, 관제사, 조종사)

■ 시험 환경

- 시뮬레이션에서 돌발상황과 시스템 고장 효과를 유발할 수 있는 기능
- 시뮬레이션의 모든 운용과 응답을 연속적으로 기록
- 검토와 작업분석을 위한 시뮬레이션 재현 기능
- 데이터 수집, 분석, 포매팅
- 시뮬레이션 도중 관제탑 시뮬레이터 실내에서 발생하는 이벤트들에 대한 영상/소리 기록과 편집

IV. 항공교통관제 훈련용 시뮬레이터

앞에서는 항공교통관제 연구용 시뮬레이터에 대하여 그 목적과 구성에 대하여 살펴보았는데, 여기에서는 항공교통관제 시뮬레이터의 구성에 대한 보다 깊은 이해를 위하여 국내에 도입되어 관제사 훈련용으로 공항공단의 청주 항공기술훈련원에서 사용 중인 캐나다 ATS Aerospace Inc. 시뮬레이터의 구성[8, 9]에 대하여 간략히 소개하기로 한다.

당 항공교통관제 훈련용 시뮬레이터는 그림6에서와 같이 관제탑 관제장치(Tower Controller) 1대, 지상 관제장치(Ground Controller) 1대, 비행 중인 항공기들을 시뮬레이션하기 위한 모의 비행조종장치(Air Pilot) 3대, 비행 또는 지상 이동 중인 항공기들을 나타내기 위한 모의비행조종장치 1대, 훈련상황 설정과 감시를 위한 교관석(Exercise Coordinator/file server) 1대, 3채널 프로젝터 방식의 영상장치, 이러한 하부 시스템들을 연동시키기 위한 통신망서버 1대 등으로 구성된다. 구성 컴퓨터들은 모두 DOS 방식의 486 PC 또는 586 PC 이며, 3대의 영상 컴퓨터(Image Generator) 만이 Tellurian AT200이다.

V. 연구용 시뮬레이터 개발을 위한 기반연구 계획

항공관제체계의 시뮬레이션은 다음과 같은 시험 평가 및 연구환경을 제시하게 되므로 항공운항 안전문제들을 상당부분 해결해 주는 해법이 된다:

가. 최종 연구목표 및 연구내용

당 과제의 최종 연구목표는 항공관제체계에 대한 안정성 평가와 효율적인 관제 절차 연구 및 개발을 위한 항공관제체계 시뮬레이션 설계기술의 기반 구축으로 종합되며, 항공관제체계의 종합적인 시뮬레이션 능력을 개발하기 위해서는 다음과 같은 세부 연구항목들이 포함되어야 한다.

- PC와 Windows NT를 기반으로 한 실시간 kernel 연구 및 executive 개발
- Fast Simulation 기법 연구
 - 시간 상수 증가법 (Increasing the Time Constant)

- 컴퓨터 집중 처리 (Computer Intensive Processing)
- 이벤트 중심 시뮬레이션 (Event Based Simulation)

■ Scenario Executive 개발

- High Level Architecture(10)와 호환인 분산 대화형 실시간 통신환경 개발
- HLA RTI(Run-Time Infrastructure)를 활용한 시뮬레이션 Infra 구축

- HLA 규약 조사 및 연구
- HLA를 적용한 응용연구

* 항공기 모델 개발

- 비행 운동
- 자동 항법제어 모델
- 정보/음성 통신

■ 대기 및 기후 모델 개발

■ 레이더 모델 개발

■ 항로 에디터 개발

■ 관제사 (Air Traffic Controllers: Dispatcher, Agents, Air Traffic Manager) 모델 개발

- 관제 이벤트 (Controller Events)
- 비행 형상 및 충돌 (Flight Path and Conflicts)
- 관제업무

■ ATC 시뮬레이션 통제 시스템의 GUI 환경 개발

당 과제에서 수행할 기반연구의 내용은 다음과 같다.

■ 항공기 모델 개발

비행계획에 따른 항공기의 비행을 시뮬레이션 하기 위해서는 항공기의 운동을 모델링 하여야 한다. 김포공항에 취항하는 다양한 기종의 항공기들에 대한 특성 매개변수들을 조사하고 추출하여 비행운동 모델에 적용함으로써 국내에서 취항하는 대부분의 기종들에 대한 사실적인 시뮬레이션이 가능하도록 한다. 항공기 비행모델은 독립된 조종사의 비행훈련용 시뮬레이터에 적용되어 사용될 경우는 그 모델이 엄밀하여야 하지만, 관제탑의 레이더에 나타나는 항공기의 움직임을 묘사하기

위해서는 단순한 모델로 접근할 수가 있고, 주어진 컴퓨터의 용량으로 다수의 항공기들을 동시에 시뮬레이션할 수 있을 것이다. 당 연구에서 요구되는 항공기의 모델은 이처럼 2단계로 구별하여 개발될 것이며, 고급 모델은 PC를 기반으로 H/W적으로 단순하게 구현되는 비행 시뮬레이터에 적용되고, 저급 모델은 또 다른 PC를 이용하여 동시에 다수의 항공기들에 대한 움직임의 묘사에 사용된다.

■ 대기 및 기후 모델 개발

다양한 기후조건 하에서의 항공관제체계에 대한 시뮬레이션을 위하여 공항 주변의 이착륙 시에 항공기의 안전에 절대적인 영향을 미치는 하강 기류(down wash)나 비행 중의 난류(Turbulence) 등에 대한 대기 및 기후 모델들을 비행운동에 적용하여 항공기의 비행과 항공관제체계를 사실적으로 시뮬레이션 한다.

■ Scenario Executive 개발

SE(Scenario Executive)는 fast-time 시뮬레이션의 초기화 과정에서 전체 시뮬레이션을 주관한다. SE는 비행계획 경로를 주컴퓨터에 load하며, 항공기가 이러한 경로를 따라 실제의 비행영역 sector들을 통과하는 시간을 제공하기 위하여 적합한 변환 프로그램을 불러들인다. 또한, SE는 비행계획을 세우기 위하여 지상 지연 루틴(ground delay routine)을 부른다. 예를 들어, 부산에서 서울로 비행하는 항공기의 비행계획을 수립하는데, 서울 김포공항이 예정 시간에 착륙을 허용할 수 없는 상황이라 판단되면 비행을 계획하지 않는다. 시뮬레이션 도중에서의 제어는 사용자가 편집한 시간표에 따른 scripted events에 의하여 제공되며, 자동적인 활주로 폐쇄, sectors 통합, 레이더 불능 등의 기능을 수행한다.

■ 항로 에디터 개발

항로 에디터는 비행경로를 계획하며, beacons과 sectors의 경계점 같은 위치에 대한 통과 예상 시각을 제공한다.

■ 관제사 (Air Traffic Controllers: Dispatcher, Air Traffic Manager 등) 모델 개발

Fast-time 시뮬레이션의 경우 loop 내에 인적 요소를 포함하지 않으므로 항공관제사들의 기능을 시뮬레이션 하여야 한다. 임의의 관제 시나리오는 다수의 관제사들을 포함하며 그들의 작업은 본래가 이벤트 중심(event based)이기 때문에 객체지향형 이벤트 중심 모델링이 가능하다. 관제사의 판단은 반면에 법적 규정에 따

르기 때문에 AI (Artificial Intelligence)에 일종인 규정 중심(rule based)의 모델링이 적절하다. 관제사에는 dispatcher와 Air Traffic Manager 그리고 다양한 controllers(예, oceanic, en-route, terminal 등)가 포함된다.

■ ATC 시뮬레이션 통제시스템의 GUI 환경 개발

ATC 시뮬레이션 구성 요소들을 빈틈없이 통합하고 관제하기 위하여 일종의 서보인 CSATCS(Control Station for Air Traffic Control Simulator)이 요구된다. CSATCS의 H/W로는 Windows NT를 사용하는 PC를, GUI 환경은 MS Visual C++/Basic을 기본 저작도구로 사용할 계획이다. 관제사 훈련용 시뮬레이터의 사양을 기본으로 하여 당 과제의 연구목적에 적합하게 개발된다. ATC 시뮬레이션은 CSATCS에 의하여 시작되고 종료된다.

■ PC/Windows NT를 기반 한 실시간 kernel 연구 및 fast-time/real-time executives 개발

당 연구개발의 최종적인 가시적 결과는 일종의 연구개발용 및 훈련용 항공관제체계 시뮬레이터이다. 훈련용 시뮬레이터의 경우는 실시간으로 항공관제체계 구성요소들을 시뮬레이션하는 것으로 충분하지만, 연구개발용인 경우는 필요에 따라 오랜 시간이 소요되는 실제의 관제과정을 단시간 내에 시뮬레이션을 통하여 결과를 도출하고 그 결과와 과정을 분석하게 된다. 이를 위하여 소위 fast simulation이 요구되는데, 항공관제 시뮬레이터에서는 연구개발을 목적으로 fast simulation을, 연구개발과 관제사와 조종사의 훈련을 위하여 실시간 시뮬레이션을 모두 수행할 수 있어야 한다. 한편, 항공관제체계의 경제적인 시뮬레이션을 위하여 PC와 Windows NT 환경을 채택하는데, 불행히도 Windows NT는 실시간 운영체계가 아니다. Windows NT를 통합 시스템의 운영체제로 사용할 경우 실시간 운영체제는 별도의 독립된 실시간 kernel을 사용하는 방식과 Windows NT kernel에 실시간 기능을 더하는 방식으로 구별할 수 있다. 이러한 방식들을 실시간성, 기능성, 편의성, 유지보수성 등을 기준으로 비교, 분석하여 항공관제체계의 시뮬레이션에 적합한 실시간 운영체제를 선정하고 이 환경 하에서 시뮬레이션 작업들을 관장하고 시간을 할당하는 real-time과 fast-time executives를 개발한다.

■ 대화형 분산 시뮬레이션을 위한 통신 기반 기술에 관한 연구

이 연구의 목표는 다중 사이트에 분산되어 처리되는 클라이언트/서버구조 [11,12]의 ATC 시뮬레이터를 지원하기 위한 통신 기반 구조를 연구하는데 있다.

분산시스템 구조를 따르는 ATC 시뮬레이터를 구성하는 다양한 클라이언트와 서버 간에 이루어지는 통신은 클라이언트와 서버의 상대적인 위치에 따라 크게 다음과 같은 세 가지로 나누어 볼 수 있다.

- 클라이언트와 서버가 같은 프로세스로 이루어지는 경우
- 클라이언트와 서버가 독자적인 프로세스로 이루어지지만 하나의 컴퓨터 시스템에 함께 존재하는 경우
- 클라이언트와 서버가 네트워크로 연결된 두 개의 별개의 컴퓨터 시스템에 존재하는 경우

첫 번째의 경우는 통신이 전혀 필요없어 통신 오버헤드를 최소화할 수 있지만 이것은 독립형 fast-time simulation에만 적용할 수 있는 극히 제한적인 방법이고, 일반적인 클라이언트/서버 기반 ATC 시뮬레이터를 위해서는 두 번째 혹은 세 번째 경우와 같이 클라이언트/서버가 위치하게 된다. 클라이언트와 서버의 배치가 위의 세 가지 경우 중 어느 것이냐에 따라 통신 오버헤드가 크게 달라지므로 네트워크로 연결된 다중의 컴퓨터에서 분산시스템 구조의 ATC 시뮬레이터를 구현하기 위해서는 통신 오버헤드를 최소화할 수 있는 방법이 연구되어야 한다. 즉, ATC 시뮬레이터를 구성하고 있는 여러 클라이언트와 서버 간의 유기적인 관계를 고려하여 통신 오버헤드를 최소화하기 위한 클라이언트/서버 배치에 관한 연구가 이루어져야 한다. 또한, 이들 통신을 효율적이고 일관되게 지원하기 위해서는 통신 작업을 계층화된 구조로 설계하기 위한 연구가 이루어져야 할 것이다. 일반적인 구조로서 다음과 같이 세 개의 계층으로 이루어진 통신구조를 생각해 볼 수 있으며 이들 각 계층에 대한 연구가 이루어져야 한다.

- 통신소프트웨어 계층의 세부적인 사항에 관계없이 각각의 클라이언트와 서버가 통신하도록 해주기 위한 클라이언트/서버별 스텝: 클라이언트/서버가 발생하는 메시지를 인코딩하고 디코딩하며 적절한 목적지로 라우트 해 주는 역할.
- TCP와 socket 등의 통신 시스템의 구체적인 세부사항을 다루어 줌으로써 스텝 구현을 용이하게 하기 위한 공통적인 통신 시스템
- 운영체제에 의해 구현된 TCP/IP

그리고, 개발하고자 하는 ATC 시뮬레이터의 프로토타입은 네트워크로 연결된

PC상에 설치되어 테스트될 예정이다. 따라서, 네트워크 구성이 클라이언트/서버 기반 ATC 시뮬레이터의 성능에 어떻게 영향을 미치는지를 연구함으로써 개발하게 될 ATC 시뮬레이터 프로토타입을 효과적으로 가동할 수 있는 네트워크 구성을 디자인하고 그 성능을 검증할 필요가 있다.

궁극적으로, 특정 항공교통제어절차를 시뮬레이션하기 위해서는 ATC 시뮬레이터와 비행시뮬레이터와 같은 별개의 독자적인 시뮬레이터들 간에 연동이 이루어져야 한다. 미국방성에서는 서로 다른 사이트에서 구축되어 작동하는 별개의 시뮬레이터들이 연합하여 대형의 대화형 실시간 분산 시뮬레이션 시스템을 구축할 수 있도록 하기 위해 시뮬레이터 간에 호환성을 증진하기 위한 시뮬레이션 기반구조로서 HLA(High Level Architecture)라는 구조를 표준화하였다. HLA RTI(Run Time Infrastructure)는 상호 연동될 시뮬레이터 간에 데이터를 주고 받을 수 있는 서비스를 제공하는 소프트웨어이다. ATC 시뮬레이터와 비행 시뮬레이터 등의 연동을 위해서는 HLA와 HLA RTI에 관한 연구 및 이를 본 연구의 ATC 시뮬레이션 체계에 적용하기 위한 방안에 대한 연구가 이루어져야 할 것이다.

■ ATC 시뮬레이션을 위한 컴포넌트 기반 클라이언트-서버 시스템의 설계

ATC 시뮬레이터는 많은 기능들이 포함된 매우 복잡한 시스템으로, 구조화되어 설계되지 않으면 유지/보수에 드는 비용이 지나치게 많이 된다. ATC 시뮬레이터는 여러개의 독립적이며 유기적인 컴포넌트로 나뉘어질 수 있다. 각 컴포넌트가 서로 다른 언어나 툴(tool)로 개발될 수 있다면, 시스템의 개발이 용이할 뿐 아니라, 시스템을 사용해 가면서 기능을 추가/삭제하는 것도 용이하게 된다. ATC 시뮬레이터는 다른 여러 ATC 시뮬레이터들과 연동되어 실행될 수 있도록 확장 가능해야 한다. 여러 시뮬레이터를 동시에 실행시키기 위해서는 고성능 컴퓨터나 이기종 PC (또는 워크스테이션) 클러스터와 같은 병렬/분산 컴퓨팅 환경이 필요하며, 이러한 환경에서 실행되기 위해서 ATC 시뮬레이터의 각 요소가 기반 컴퓨터시스템과는 독립적으로 구축될 필요가 있다.

따라서, 당 연구는 ATC 시뮬레이터의 각 기능을 유기적인 개체로 분리하여, 각 기능이 클라이언트-서버 형태로 구축될 수 있도록 한다. ATC 시뮬레이터는 매우 복잡한 시스템으로써, 시뮬레이터를 구성하는 요소들을 구조화하지 않으면 시스템 자체를 이해하기 어렵고, 유지/보수하기가 힘들게 된다. 이를 해결하기 위한 한가

지 방법은 ATC 시뮬레이터의 각 요소를 가능한 한 독립적이며 유기적인 관계를 갖는 모듈들로 나누어 설계하는 것이고, ATC 시뮬레이터는 그 특성상 이런 방식의 설계가 적합한 시스템이다. ATC 시뮬레이터의 각 요소는 분산 대화형 모드에서 실행되는 것으로, 이는 서비스를 제공하는 서버와 서비스를 제공받아 일을 처리하는 클라이언트들로 설계되는 것이 바람직하다.

본 연구에서는 우선 설계하고자 하는 ATC 시뮬레이터에 포함될 기능들을 파악하고, 이들간의 유기적인 관계를 분석한다. 이 관계를 바탕으로 하여 전체 시스템이 어떤 방식의 클라이언트-서버 쌍으로 나뉘어야 시스템의 각 요소 간의 통신 오버헤드를 줄이면서 신속하게 실행될 수 있는가를 분석한다. 시스템의 각 요소는 클라이언트-서버 형태로 구성되는데, 이 경우 클라이언트와 서버 간의 작업량을 적절하게 분배하는 것이 시스템의 효율에 큰 영향을 미친다. 따라서 본 연구에서는 ATC 시뮬레이터의 각 요소에 대해서 클라이언트와 서버 간의 작업분배가 어떤 형태로 이루어져야 하는가를 분석하려고 한다. 유사한 방식으로 구성될 수 있는 요소끼리 그룹을 정하여, 각 그룹에 대해 클라이언트-서버의 작업량을 정한다. 클라이언트와 서버는 빈번한 통신을 통해 자료를 주고받고, 일을 처리해주는 분산시스템의 특정한 형태이다. 따라서 클라이언트와 서버 간의 인터페이스는 체계적으로 잘 정의/정리되어야 한다. 잘 정의된 인터페이스는 후에 시스템에 다른 기능을 추가/삭제하는데도 큰 도움을 주게 된다. 본 연구에서는 클라이언트 서버형태의 시스템 구축에 기반이 되는 클라이언트-서버 간의 인터페이스를 정의하여 효율적인 클라이언트-서버 체계를 마련하도록 할 계획이다.

ATC 시뮬레이터를 컴포넌트 기반 시스템으로 구축하여 각 컴포넌트가 서로 다른 환경에서 개발/실행될 수 있도록 하며, 시뮬레이터의 기능을 추가/삭제하는데 용이하도록 시스템을 설계한다. ATC 시뮬레이터는 유기적 관계를 갖는 여러 개의 컴포넌트로 나뉘어질 수 있는 매우 복잡한 시스템이다. 이 시스템의 각 컴포넌트가 서로 다른 환경에서 개발되고 실행될 수 있도록 설계한다면, 더욱 정교한 ATC 시뮬레이터의 개발, 유지, 보수가 용이하게 될 것이다. 각 컴포넌트는 개발하는데 쓰인 언어, 도구 등 뿐 아니라, 그 컴포넌트가 실행될 컴퓨팅 환경도 다를 수 있다. 즉 이기종 분산 컴퓨팅 환경에서 실행이 가능하다는 뜻이다. 이렇게 이기종 분산 환경에서 신속하게 실행될 ATC 시뮬레이터를 구축하기 위하여, 본 연구에서는 우선

각 컴포넌트가 어떤 언어, 도구를 써서 개발되어야 신속한 처리가 가능한지를 분석하고, 나아가 이기종 환경에서 개발된 컴포넌트들이 어떻게 효율적으로 자료/명령을 전달하면서 실행될 수 있는지, 그 방법을 제시하려고 한다. 컴포넌트 간의 통신은 다양한 수준에서 이루어질 수 있으나, 여기서는 시스템의 유지, 보수, 부분적 추가, 삭제를 용이하게 하기 위하여, 사용자 수준에서 쓰기 편한 컴포넌트 간의 통신방법을 정의하고, 이를 테스트하여 그 성능을 분석해 보려고 한다. 컴포넌트 기반으로 설계된 이러한 시스템은 각 컴포넌트의 수정/보완이 용이하고, 서로 다른 환경에서 구축된 컴포넌트들을 통합하여 전체 시스템이 구축될 수 있으며, 여러 개의 ATC 시뮬레이터를 연결하여 작동시킬 수 있도록 확장성을 보장한다.

Ⅵ. 향후 계획

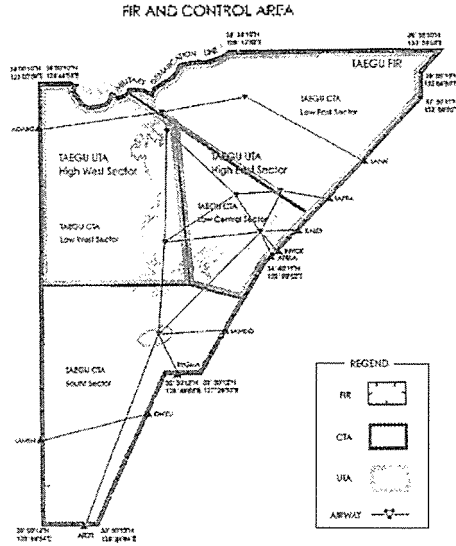
이상과 같이 항공교통관제 시뮬레이터의 필요성, 관련 연구개발사례 및 훈련용 시뮬레이터의 구성 예, 세종대학교에서 준비중인 항공교통관제 연구 및 평가용 시뮬레이터의 목적 및 내용 등에 대하여 살펴보았다. 궁극적으로는 NASA의 SDTF와 같은 항공교통관제 시뮬레이터가 우리에게도 요구되나, 당 대학에서는 최소 시스템의 연구개발부터 단계적으로 접근하고자 한다. 이를 위하여 현재 기본 시뮬레이터의 요구사항 분석과 개념설계를 자체적으로 수행 중이다.

[참고문헌]

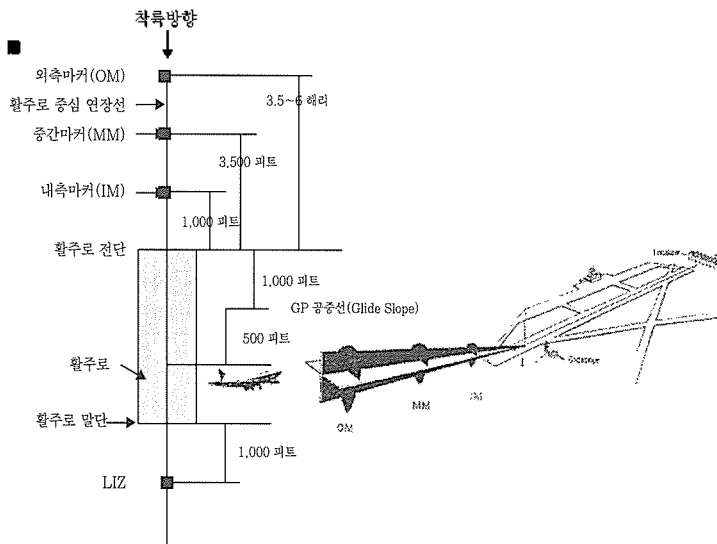
1. 항공교통관제기본, 한국공항공단 항공기술훈련원, 1998. 3.
2. Richard F. H., "Validating NASA's Surface Development and Test Facility, Air Traffic Control Tower Simulator", Proceedings of AIAA Flight Simulation conference, pp. 181-189, Aug., 1997.
3. <http://www.tc.faa.gov/act500/nsc/nsc1.html>.
4. Siksik D. N., Prager R., and Muise J., "A Rule Based Approach to Fast Time Simulation of an Air Traffic Controller", Proceedings of AIAA Flight Simulation conference, pp. 119-128, Aug., 1996.

5. Hached M. and Beauchesne G., "Aircraft Performance Modeling for Air Traffic Control Simulation Applications", Proceedings of AIAA Flight Simulation conference, Aug., 1995.
6. Air Traffic Control Manual of Operations, Transport Canada, April 1993.
7. <http://www.nlr.nl/public/facilities/f150-01/index.html>.
8. Hardware Technical Manual for the Korea Airports Authority Air Traffic Control training Simulator, Doc. # 9516-H-MA-TE-X-00, ATS Aerospace Inc..
9. Visual Tower and Radar Simulator Acceptance Test Specifications, Doc. # 9516-C-SP-TS-X-0B, ATS Aerospace Inc..
10. HLA web site, <http://www.dms0.mil>.
11. Alex Berson, "Client/Server Architecture," McGraw-Hill, 1992.

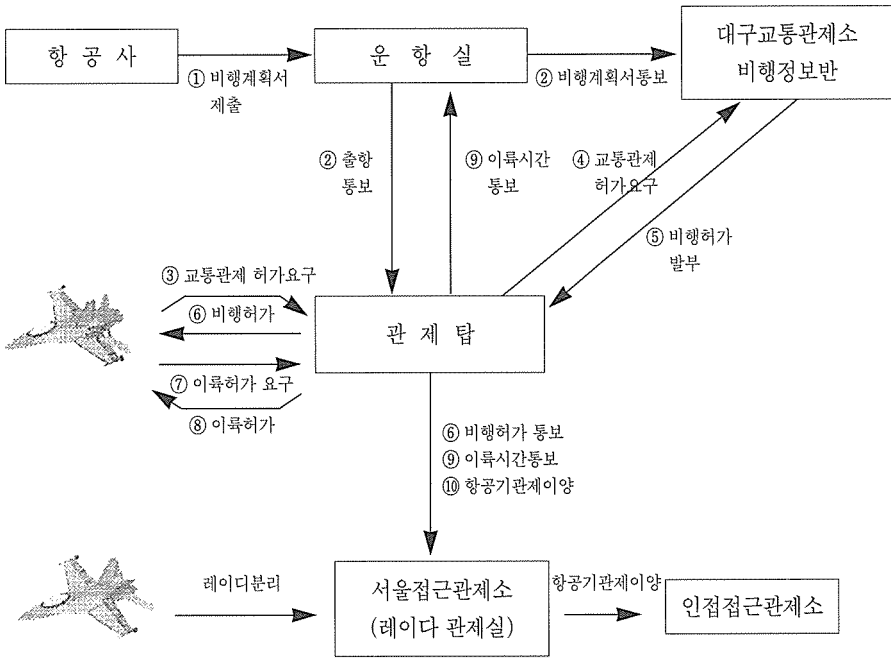
〈그림 1〉 한국 비행정보구역도 (대구 FIR)와 관제구



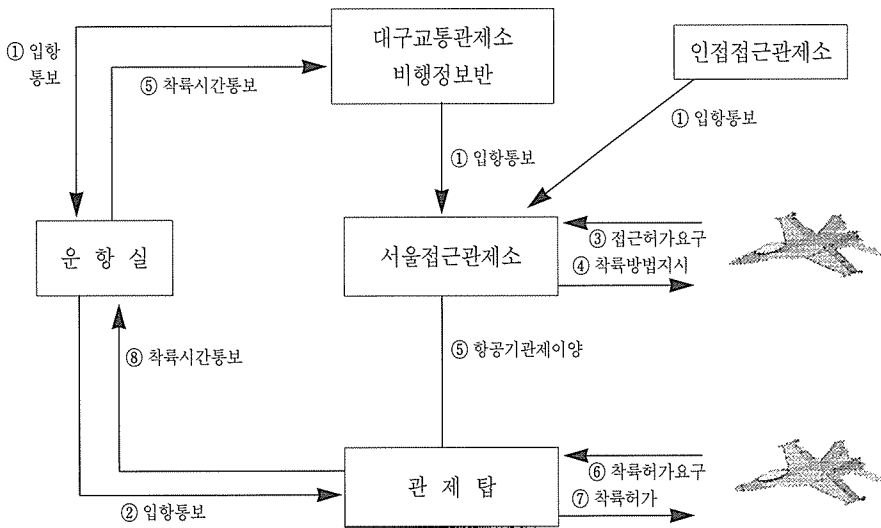
〈그림 2〉 계기착륙시설 (ILS)의 구성도



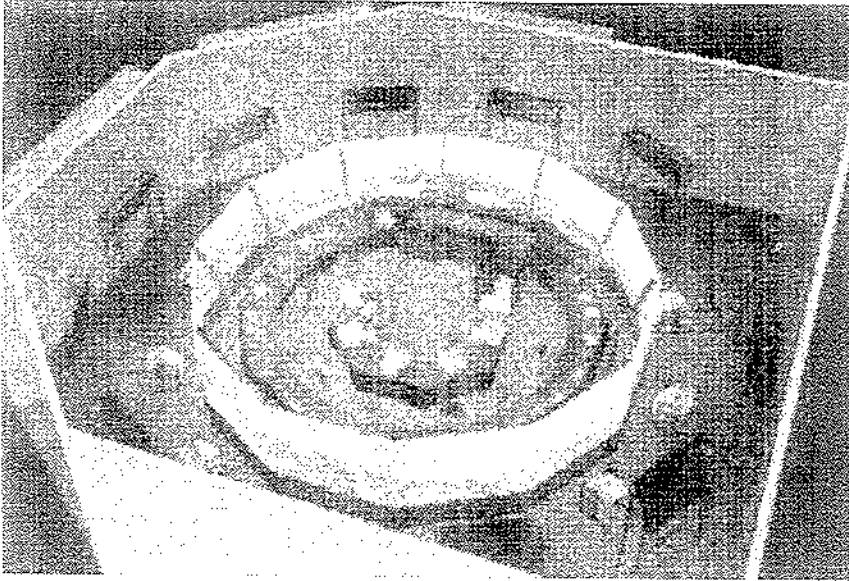
〈그림 3〉 계기비행절차 도해 (이륙)



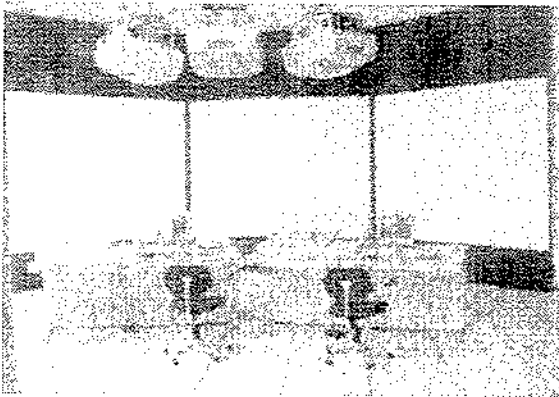
〈그림 4〉 계기비행절차 도해 (착륙)



〈그림 5〉 NASA Surface Development and Test Facility (SDTF)의 형상



〈그림 6〉 한국공항공단의 항공교통관제 훈련용 시뮬레이터



〈그림 7. 항공교통관제 훈련용 시뮬레이터 시스템 구성도〉

