

# 공항의 계류장 관리 스케줄링 및 조정을 위한 전문가 시스템

## Ramp Activity Expert System for Scheduling and Co-ordination

조근식\*, 양종윤\*\*

Geun-Sik Jo\* and Jong-Yoon Yang\*\*

### 요 약

이 연구에서는 항공기의 주기 문제를 해결하여 주는 스케줄링 시스템과 그 조정을 위한 전문가 시스템 (RACES : Ramp Activity Co-ordination Expert System)을 설계 및 개발한 내용을 기술하고 있다. RACES는 공항에서 매일 발생하는 출발편 및 도착편 항공기를 브릿지(bridge)와 스팟(spot)에 배정하기 위해 인간 전문가(human expert)로부터 습득한 해당 분야의 지식(도메인 지식) 및 휴리스틱(heuristic)을 지식 베이스로 갖고 있다. 이 RACES는 브릿지/스팟과 항공기 간에 내적 관계, 예를 들어 승객 및 공항의 그라운드 핸들링(ground handling)등과 같은 복잡하며 동적인 제약조건 들로부터 발생하는 복잡한 스케줄링 문제를 수반한다. 매일 발생하는 600편 정도의 항공기에 대한 주기장 관리 스케줄링이 인간 전문가에 의해 수행되어졌을 경우에는 약 4~5시간이 소요되는 반면 RACES에 의해 수행되어졌을 경우에는 약 20초 정도의 시간이 소요되었고 RACES로부터 얻어진 스케줄링 결과는 해당 분야의 전문가들로부터 인정되었다. RACES는 또한 예외적인 상황이 발생했을 경우에 스케줄의 부분적인 조정을 처리하도록 설계되었다. 하루의 스케줄링이 완료된 후 항공기의 변경 및 지연 메시지는 도메인 전문가의 지식을 바탕으로 스케줄링에 반영되어 스케줄이 조정되어야 한다. 동적 재스케줄링(reactive scheduling) 단계는 도메인 전문가의 지식 모델 분석을 통해 사용자 그래픽 인터페이스의 규칙과 시나리오로써 효과적으로 나타내어진다. 항공편의 변경 및 취소로 인해 발생하는 항공기 배치의 조정은 현재 스케줄에 반영되어야 하기 때문에 이러한 항공기 배치의 조정은 동적 재스케줄링을 위해 메인 프레임으로부터 RACES에게 통보되어야 하며 부분적인 재스케줄링을 처리하는 것에는 불규칙적인 요소들이 많기 때문에 RACES에 의해 스케줄의 조정이 반 자동적으로 수행된다.

### Abstract

In this paper, we have described the Ramp Activity Coordination Expert System (RACES) which can solve aircraft parking problems. RACES includes a knowledge-based scheduling problem which assigns every daily arriving and departing flight to the gates and remote spots with the domain specific knowledge and heuristics acquired from human experts. RACES processes complex scheduling problem such as dynamic inter-relations among the characteristics of remote spots/gates and aircraft with various other constraints, for example, custome and ground handling factors at an airport. By

\*인하대학교 전자계산공학과 (Dept. Computer Science and Eng., Inha Univ.)

\*\*한국전자통신연구원 컴퓨터·소프트웨어 연구소 (Electronics and Telecommunications Research Institute, CSTL)

· 논문번호 : 98-1-7

· 접수일자 : 1998년 8월 17일

user-driven modeling for end users and knowledge-driven near optimal scheduling acquired from human experts, RACES can produce parking schedules of aircraft in about 20 seconds for about 400 daily flights, whereas it normally takes about 4 to 5 hours by human experts. Scheduling results in the form of Gantt charts produced by the RACES are also accepted by the domain experts. RACES is also designed to deal with the partial adjustment of the schedule when unexpected events occur. After daily scheduling is completed, the messages for aircraft changes and delay messages are reflected and updated into the schedule according to the knowledge of the domain experts. By analyzing the knowledge model of the domain expert, the reactive scheduling steps are effectively represented as rules and the scenarios of the Graphic User Interfaces (GUI) are designed. Since the modification of the aircraft dispositions such as aircraft changes and cancellations of flights are reflected to the current schedule, the modification should be notified to RACES from the mainframe for the reactive scheduling. The adjustments of the schedule are made semi-automatically by RACES since there are many irregularities in dealing with the partial rescheduling.

## I. 서 론

항공기의 주기 문제는 출발과 도착하는 항공기를 여러 가지 요구에 만족하도록 브릿지와 스팟에 배정하는 스케줄링 문제이다. 이러한 항공기의 주기 문제는 Job-shop 스케줄링으로 분류될 수 있는 스케줄링 문제의 한 종류이다. 게다가, 이 문제에서는 시간적 추론 메커니즘의 특징을 갖고 있다. 항공기의 도착과 출발의 횟수가 증가하거나 운영하는 단계에서 예기치 않은 사건이 발생한다면 스케줄링은 좀 더 동적이 되고 더 어렵게 되며 항공 교통의 증가와 승객의 요구 또한 항공기 주기 스케줄링 결과를 얻는데 어렵게 하는 요인이 된다. 실제로 스케줄링 전문가는 수동으로 매일 매일의 스케줄을 작성하게 된다. 전문가는 주어진 날짜 동안 항공기 스케줄(flight schedule)의 결과를 얻기까지는 복잡한 수 작업을 해야 하고 시간까지 소비하게 된다.

전통적으로 연구자들은 이런 종류의 문제를 풀기 위해 수학적 프로그래밍 기법을 사용해 왔다. 산업 공학에서 사용하는 Operational Research가 대표적인 예이다. 그러나 수학적인 변수만으로 이루어진 도메인 지식과 제약조건의 모델링(constraints modeling)은 매우 어렵다. 최근에는 많은 연구자들이 이러한 문제를 푸는데 constraint directed reasoning, 전문가 시스템, Constraint Satisfaction Problems(CSP)과 같은 인공지능 기법의

사용을 제안했다[4],[7]-[9]. 인공지능 기법은 복잡한 스케줄링 문제를 모델링 하는데 있어서 수학적 프로그래밍보다 더 나은 융통성과 효율성을 제공한다.

많은 전문가 시스템이 이런 방면에서 소개되고 있는데 특히 항공 산업에서 실용적인 전문가 시스템이 개발되어 왔다. American Airline사는 복잡한 항공 교통과 자원을 효과적으로 관리할 수 있는 Gatemanager라는 시스템을 개발했다[2]. COSYTEC사는 APACHE라는 automatic ramp scheduling system을 개발했다[3]. Texas State 대학교에서 개발한 GATES시스템은 New York에 있는 JFK공항의 브릿지 배정과 추적을 제어한다[11].

## II. 주기장 스케줄링 관리 문제

본 연구에서 개발한 RACES(Ramp Activity Coordination Expert System)[7]는 브릿지나 스팟에 도메인에 특정한 지식과 휴리스틱을 이용해 날마다 항공기를 배정한다. 시스템은 해가 될 수 있는 가능한 탐색공간을 제약조건을 통해 탐색 전에 미리 축소시키는 도메인 여과 기법을 사용하여 다양하고 제한된 탐색 공간에서의 불일치성을 제거할 수 있었다. 사용자가 최적화된 해를 찾기 위해서 RACES는 제약조건을 만족시키는 효율적인 휴리스틱 스케줄링 방법을 이용한다.

RACES는 항공기 스케줄, 항공기 형태, 스팟의 특성 그리고 그라운드 핸들링 등의 상태를 고려하여 최적에 가까운 스케줄링의 결과를 산출한다. 항공기는 주어진 제약조건에 만족하는 적합한 스팟이나 브릿지에 배정되어야 한다. 또한 브릿지나 스팟은 크기나 급유시설에 의해 구별되어진다. RACES는 제약조건에 종류에 따라 그림 1과 같이 크게 3부분으로 구분되는 3차원 형태의 제약조건 해석기와 같이 보여질 수 있다. 그리고 3차원 공간을 간트 차트의 형태의 2차원 공간으로 표현이 가능하기 때문에 RACES는 하루 전에 미리 다음 날의 항공기 주기 스케줄을 표현하는 간트 차트를 산출하여 사용자에게 제공한다.

제약조건을 해결하는 데는 그 성질에 따라 세 가지의 제약조건을 고려해야 한다. 첫번째는 스케줄링 작업 동안에 반드시 만족되어야 하는 강한(strong-hard) 제약조건이다. 만약 이 제약조건이 위배된다면 결과는 더 이상 유효하지 않게 된다. 두번째는 특정한 환경하에선 위반될 수도 있는 약한(weak-hard) 제약조건이다. 이 제약조건은 사용자와 상호 대화 시에 위반될 수도 있다. 하지만 RACES에 의해서는 위배되지는 않는다. 마지막, 세번째는 특별한 항공기와 특별한 시간에 적용될 수 있는 지키면 좋지만 경우에 따라서는 안 지켜도 무방한 약한 제약조건(soft-constraint)이다. 스케줄링

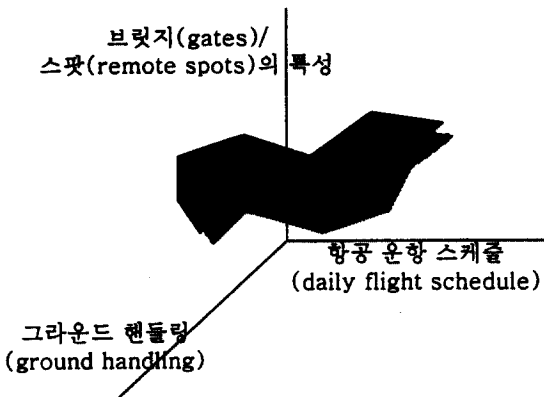


그림 1. 항공기 주기 문제의 3차원적 제약조건 해석 모델

Fig. 1. Three dimensional semantic model for aircraft allocation problem.

동안 soft-constraint가 체크되고 가능하면 만족시키기 위해 시도하고 만약 만족되지 않는다면 이 조건은 RACES에 의해 보류된다.

RACES는 두 개의 상이한 지식 기반의 시스템으로 분류된다. 그 중 하나는 3장에서 설명하게 될 하루의 스케줄을 산출하기 위해 그 전날에 산출하는 것이고 다른 하나는 4장에서 보게 될 운영하는 당일 날에 스케줄을 조정하는 것이다.

### III. 초기 스케줄의 생성

우리가 도메인 전문가로부터 얻은 지식의 문서화를 완성했을 때 그 문서는 데이터베이스에 대한 기술을 제외하고도 약 20 페이지에 달했다. 더욱이 그러한 지식은 너무 도메인에 특정하기 때문에 일반인들이 그러한 지식을 이해하기가 힘들다. 이 장에서는 RACES 시스템에서 전개되고 있는 스케줄링 전략을 기술하겠다. RACES는 당일의 스케줄을 조정하기 위해 당일의 주기 스케줄의 결과를 나타내는 간트 차트를 산출한다.

#### 3-1 도메인 여과에 의한 일관성 유지

스케줄링 프로시저는 시간 제한의 제약조건을 만족시키기 위해 연속적인 시간값들을 이산적인 시간값들로 결합시킨다. 연속적인 시간 도메인을 다루기 위해서는 연속적인 시간값들을 이산적인 시간요소들로 분해되며, 또한 브릿지와 스팟의 사용 가능한 모든 시간들은 시간 키(time keys)에 의해 분류된다. 시스템이 스케줄링을 처리할 때는 세 가지 단계를 거쳐 도메인을 여과하여 제약조건을 위반하는 요소들을 제거하는데 그림 2는 항공기 스케줄을 위한 도메인 여과 처리의 예를 보여주고 있다.

첫번째 단계에서 RACES는 항공기와 항공 운항 스케줄 제약조건에 의해 도메인을 여과한다. 두 번째 단계에서는 토잉과 같은 그라운드핸들링 제약조건에 의해 도메인을 여과하고 마지막 단계에서는 브릿지와 스팟의 제약조건에 의해 도메인을 여과한다. 이러한 처리의 결과로 RACES는 항공기의 주기 스케줄을 위한 탐색 공간을 상당히 줄일 수 있다.

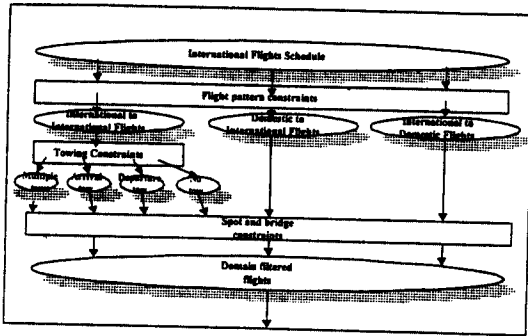


그림 2 스케줄링 전 상태의 도메인 여과  
Fig. 2. Domain reduction prior to scheduling.

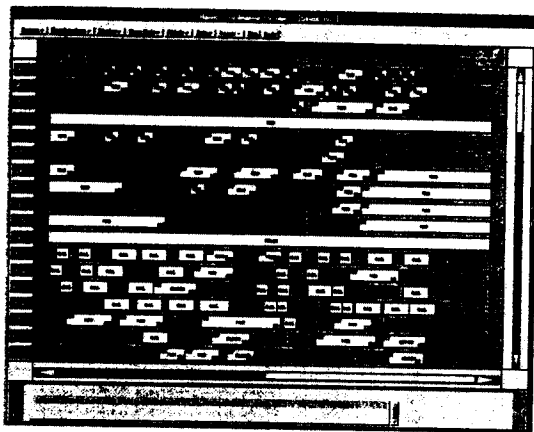


그림 3. RACES에서의 초기 스케줄의 생성  
Fig. 3. Intial scheduling in RACES.

3-2 최적 스케줄링에 가까운 지식 접근 방식

RACES는 사용자의 요구에 의한 최적 해를 고려한다. 이 때에 중요하게 고려되어야 할 사항은 RACES가 공항의 포화로 인해 브릿지나 스팟에 아직 배정되지 못한 대기중인 항공편이 최소화되도록 하는 것이다. 또한 스케줄링 중에 RACES는 공항에서 토잉이 가장 적게 일어나도록 시도한다. 만약 시스템이 부득이하게 대기중인 항공편을 초래한다면 그 항공편을 좀 더 작은 항공기가 되도록 하는 것이다. 만일 대기중인 항공편에 상대적으로 큰 항공기가 포함되어 있다면 큰 스팟들은 대개 충분히 있지 않기 때문에 이러한 항공기들을 자동화된 스

케줄 후에 수(manual) 작업으로 할당하는 것은 어렵게 된다.

RACES에는 두 개의 휴리스틱 스케줄링 방법이 있다. 하나는 주기를 위해서 시간 간격(time-span)에 중심을 두어 최적(best-fit)의 배정을 하는 방법이고 다른 하나는 항공기 크기에 중심을 둔 배정 방법이다. 각각의 방법들은 사용자 방식의 최적 해를 찾는데 있어서 장점과 단점들을 가지고 있는데 이 두 방법을 사용할 경우에 스케줄링 동안의 어떤 단계에서는 하나의 방법이 다른 방법과 충돌을 일으킬 수 있는 문제가 발생한다. 본 연구실에서 개발한 RACES는 시간 간격에 중심을 둔 방법과 크기에 중심을 둔 방법 사이의 충돌을 피하기 위해서 경험에 의해 두 방법 사이에 균형(trade-off) 위치를 찾아 각각의 방법의 장점을 최대한으로 이용하여 개발하였다.

IV. 지식기반의 동적 재스케줄링

스케줄링을 마친 후에 실제적으로 스케줄링이 적용되어지는 날이 임박했을 때 주위 환경의 변화에 따른 예상하지 못한 사건이 발생할 수 있다. 항공기 주기 장소와 시간 슬롯(time-slot)은 이러한 실제 항공기의 운항 지연이나 항공 스케줄의 변경과 같은 스케줄의 갑작스러운 변화에 대해 주기 스케줄이 알맞게 조정되도록 해야 한다.

4-1 동적 재스케줄링에서의 전문가 모델

그림 4에서 메타 모듈은 메시지 핸들링 모듈(message handling agent)에서 전달되는 메시지에 따라 어떤 모듈이 활성화되는가에 대한 지식을 가지고 있다. 지면 관계상 다음의 메시지 핸들링 모듈과 동적 재스케줄링 모듈은 간략하게 설명하겠다.

메시지 핸들링 모듈이 하는 일은 운항에 관한 일관성을 나타내는 슬링(sling)을 체크하고 사이클(cycle)을 검출하고 관련 있는 메시지를 묶어준다.

동적 재스케줄링 모듈(Reactive scheduling agent)의 작업은 실제적인 작동 중에 주위환경의 변화에 따른 메시지에 의한 재스케줄링이다. 이러한

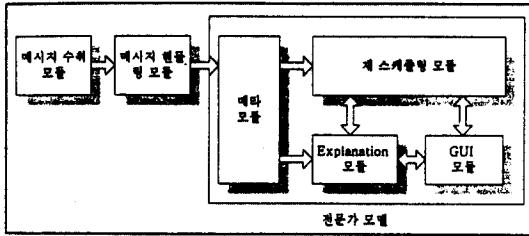


그림 4. 재 스케줄링에서의 전문가 모델  
 Fig. 4. Expert system model for reactive scheduling.

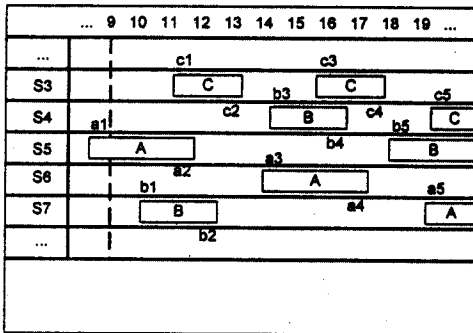
작업은 새로운 대기(stand-by) 항공기, 바들(bars)을 만드는 것과 이러한 바들을 배정하는 두 개의 서로 다른 작업으로 분리될 수 있다. 동적 재스케줄링 모듈의 가장 중요한 작업은 새로운 대기 바들을 간트 차트에 배정하는 것인데 이러한 작업은 복잡하고 섬세한 도메인 지식이 필요하다. 조정되는 규

칙의 변경 등과 같은 것은 이러한 처리과정을 위해 구현되었다.

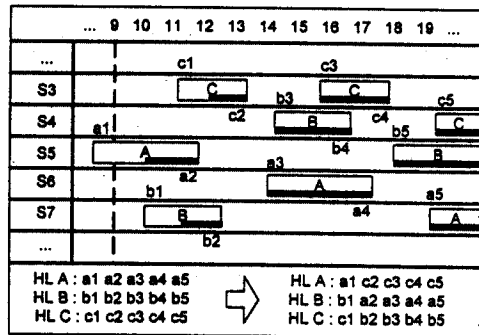
4-2 상호작용적인 그래픽 사용자 인터페이스

항공기의 운항 지연, 수리, 고장 등으로 스케줄상으로는 운행 중이어야 하는 항공기가 운행할 수 없는 경우가 종종 발생한다. 이런 경우에는 다른 항공기가 그 항공기의 운행을 대신하게 되는데 이러한 상황 발생을 AC(AirCRAFT) 변환이라고 한다. 예를 들어 그림 5를 보기로 하자.

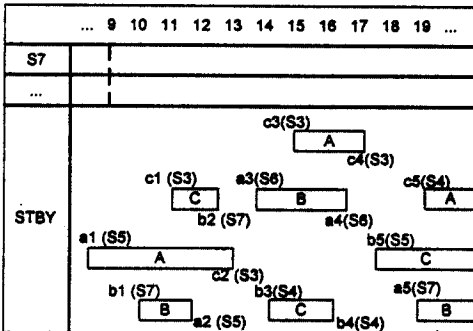
그림 5에서 대문자는 항공기에 대한 등록된 항공기 고유 번호(HL number)를 나타내고 소문자는 비행 번호를 나타낸다. 초기 스케줄에서는 항공기 A는 운항편 a1로 도착한 뒤 운항편 a2 출발하게 되어 있었고 비슷하게 항공기 B는 운항편 시간 b1에 도착한 뒤 운항편 시간 b2에 출발하게 되어 있었다. 마찬가지로 C는 c1으로 도착 c2로 출발 예정이



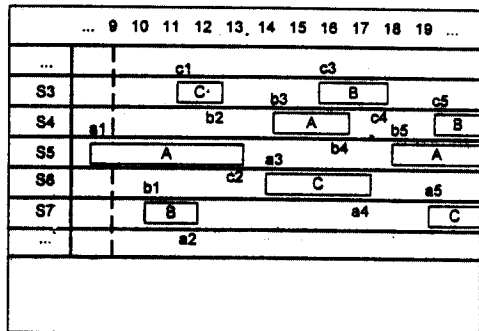
(a) 초기 스케줄 결과



(b) AC 변환 메시지



(c) 새로운 stand-by bars and 변환된 HL numbers



(d) 새로운 스케줄 결과

그림 5. 초기 스케줄 조정을 위한 각각의 단계  
 Fig. 5. Stage of reactive scheduling.

었다. 만약 항공기 A를 B, B를 C, C를 A로 순환적으로 바꾼다면 A는 a1으로 도착해서 c2로 출발하고 B도 b1도착 a2출발, C도 c1도착 b2로 출발할 것이다.

그림 5에서 본 바와 같이, 각 항공기들은 도착과 출발에 대한 시간 슬롯을 가지고 있다. 만약 어떤 항공기의 운항 스케줄이 바뀌면 원래 해의 시간 막대(solution bar)의 길이는 바뀔 것이다. 이것은 초기 스케줄의 결과인 시간 막대의 조정이 필요하다라는 것을 의미한다. 항공기 스케줄의 변환은 두 대의 항공기 또는 세 대 또는 그 이상의 항공기 사이에 발생할 수 있다. 이것은 그림 5에서 볼 수 있듯이 일반적으로 순환적으로 발생한다. 따라서 완벽한 자동 조정은 때때로 적합하지 않고 자동화하기에는 너무도 복잡할 때가 있다. 지능형 상호작용 GUI는 사용자에게 수동적인 조정을 제공한다.

## V. 결 론

CHIP(Constraints Handling In PROLOG)으로 구현한 RACES(Ramp Activity Coordination Expert System)은 70개의 GUI 메뉴를 가진 약 5만 라인의 Prolog code로 구성되어 있다. RACES는 숙련된 인간 전문가의 문제 해결 과정과 유사한 방법을 이용하며 특정한 도메인 지식과 경험을 표현하고 처리한다. 시스템이 스케줄링할 때, RACES는 도메인 여과 기법을 이용하여 탐색 공간을 줄이고 균형 스케줄링 휴리스틱(trade-off scheduling heuristics)을 이용하여 사용자 방식의 최적에 가까운 스케줄링(user-driven near optimal schedule)을 제공한다. 시스템의 정확성을 검증하기 위해 약 120일 동안의 실제 항공사 운행 데이터를 가지고 RACES를 운영하였고 그 결과는 도메인 전문가에 의해 분석되어 증명되어져 현재 대한항공에서 다른 시스템들과 연계되어 실무에 적용되고 있다. 최근 RACES는 실제 환경에서 김포공항의 계류장의 부족으로 인하여 약 70%를 재스케줄할 수 있는 능력을 가지게 되었다. 동적 재스케줄링은 실제 스케줄링 시스템을 이용하기 위한 개발자의 가장 중요하게 고려해야 할 주제 중의 하나이다. 그

러나 시스템이 그 자체에 의해 재스케줄링을 완벽하게 지원할 수는 있으나 사람과 같은 융통성을 완전히 발휘하기 위해서는 GUI를 통한 시스템과의 상호 작용이 필수적이다. 따라서 몇몇 조정은 GUI를 통한 사용자의 수작업에 의한 조정이 병행되어지고 있다. 그리고 상호작용 GUI는 올바른 결론을 도출하기 위한 사용자를 지원하는 역할을 한다.

## 참 고 문 헌

- [1] Albert E. Croker, Vasant Dhar, A Knowledge Representation for Constraint Satisfaction Problems, *IEE transaction on Knowledge and data Engineering*, vol. 5, no. 5, October, pp. 740-752, 1994.
- [2] American airlines, GateManager, *Precision Technologies Inc.*, April, 1993.
- [3] APACHE : Constraint Logic Programming Solves Aircraft Parking Program, *The Magazine of the Advanced Systems and Software*, vol. 7, no. 9, September, 1991.
- [4] Baptiste Pierre, Legeard Bruno, Manier Marie-Ange, Varnier Christiphe, A scheduling Problem Optimization Solved with Constraint Logic Programming, *Artificial Intelligence*, vol. 42, pp. 200-231, 1994.
- [5] Bing Liu, Problem Acquisition in Scheduling Domain, *Expert System with Applications*, vol. 6, pp. 257-265, 1994.
- [6] Daniel Frost, Rina Dechter, In search of the best constraint satisfaction search, *AAAI'94*, pp. 301-306, 1994.
- [7] Geun-Sik Jo, Jong-Jin Jung and Chang-Yoon Yang, *Expert System for Scheduling in an Airline Gate Allocation*, *Expert Systems with Applications*, Elsevier Science Ltd, vol. 13, no. 4 pp. 275-282, 1998.
- [8] M.Dincbas, P.Van Hentenrych, H. Simonis, A. Aggoun, T. Graf, Applications of CHIP to industrial and engineering

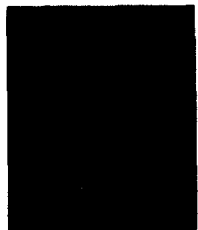
problems, *In First Int. Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert System*, Tullahoma, Tennessee, June 1988.

[9] M.S. Fox, Constraint-Directed Search : A Case Study of Job-Shop Scheduling, Research Notes in Artificial Intelligence, *Fitman Publishing*, 1987.

[10] Patrick Prosser, Domain filtering can degrade intelligent backtracking search, *International Joint Conference on Artificial Intelligence*, pp. 262-267, 1993.

[11] Robert P. Brazile and Kauthleen M. Swigger, GATES : An Airline Gate Assignment and Tracking, *IEEE Expert*, pp. 33-39, 1988.

**조 근 식 (趙根植)**



1982년 : 인하대학교 전자계산학과 (이학사)  
1985년 : Queens College /CUNY Computer Science전공(이학석사)  
1991년 : City University of New

York Computer Science전공(이학박사)

1988년 4월~1991년 2월 : Logic Based Systems Lab. 연구원

1997년 3월~1998년 8월 : 인하대학교 전자계산공학과 학과장

1996년 1월~1996년 12월 : 한국정보과학회 논문지 편집위원

1991년 3월~현재 : 인하대학교 전자계산공학과 부교수

1994년 1월~현재 : 한국전문가시스템학회 학술이사

1997년 1월~현재 : 한국 전문가 시스템 학회 논문지 편집위원장

관심분야 : 전문가시스템, 지식기반 스케줄링, Intelligent Agent, Constraint Satisfaction Problems

**양 종 윤 (梁種允)**



1996년 : 인하대학교 전자계산공학과(공학사)  
1998년 : 인하대학교 대학원 전자계산공학과(공학석사)  
1998년~현재 : 한국전자통신연구원 컴퓨터비전팀 연구원

관심분야 : 전문가시스템, 에이전트, 컴퓨터비전