

# 유전알고리즘을 이용한 판넬블럭조립공장의 일정계획시스템\*

최형림\*\* · 류광렬\*\*\* · 조규갑\*\*\*\* · 임호섭\*\*\*\*\* · 황준하\*\*\*\*\*

## A Scheduling System for Panel Block Assembly Shop in Shipbuilding using Genetic Algorithms

Hyung Rim Choi\*\*, Kwang Ryel Ryu\*\*\*, Kyu Kab Cho\*\*\*\*,  
Ho Seob Lim\*\*\*\*\*, Jun Ha Hwang\*\*\*\*\*

### 요 약

본 논문은 조선업에서 평블럭을 생산하는 판넬블럭조립공장의 일정계획 문제를 해결하기 위한 유전알고리즘의 적용 방안을 제시하고 있다. 판넬블럭조립공장의 일정계획은 작업장별 부하 평준화와 각 작업장내에서의 일자별 부하 평준화라는 두가지 목표를 가지고 있다. 이러한 목표를 달성하기 위해 본 논문에서는 유전알고리즘을 계층적으로 나누어 적용하였다. 상위단계 유전알고리즘은 작업장별 부하 평준화를 담당하며 하위단계 유전알고리즘은 상위단계 유전알고리즘의 결과를 바탕으로 각 작업장내에서 일자별 부하 평준화의 최적화를 담당한다. 실험 결과, 유전알고리즘에 의한 일정계획이 수작업에 의한 방법보다 처리시간이 짧게 소요되고 부하 평준화의 측면에서 더 좋은 해를 얻을 수 있음을 확인하였다.

Keywords : Shipbuilding Scheduling, Genetic Algorithms

\* 이 논문은 1995년도 동아대학교 학술연구조성비에 의하여 연구되었습  
\*\* 동아대학교 경영정보학과 부교수  
\*\*\* 부산대학교 컴퓨터공학과 조교수  
\*\*\*\* 부산대학교 산업공학과 교수  
\*\*\*\*\* 동아대학교 경영정보학과  
\*\*\*\*\* 부산대학교 컴퓨터공학과

## 1. 서론

하나의 선박은 유사한 크기를 가진 수백개의 블록들로 이루어진다. 각 블록은 평블럭과 곡블럭의 두가지 유형으로 분류되며 평블럭은 판넬블럭조립 공장에서, 곡블럭은 곡블럭조립공장에서 각각 조립된다. 본 논문은 유전알고리즘을 이용한 판넬블럭조립공장의 일정계획 방법을 제시한다.

현행 판넬블럭조립공장의 일정계획은 일정계획 전문가의 경험에 의존해서 수작업으로 수립되고 있는데 수작업으로 일정계획을 수립하는데는 적어도 일주일 정도가 소요되며 이렇게 해서 작성된 일정계획이 최적이라는 보장도 없다. 특히 판넬블럭조립공장의 일정계획 수립시 최적화를 위해 효과적으로 적용가능한 경험적 휴리스틱이 거의 존재하지 않으므로 휴리스틱이나 지식기반시스템으로 일정계획 시스템을 개발하기 어렵다. 본 논문에서는 이러한 문제를 해결하기 위해 유전알고리즘을 적용하였다. 일반적인 휴리스틱 알고리즘들은 주어진 문제의 지식에 의존하기 때문에 그 적용범위가 매우 제한적인 반면 유전알고리즘은 자연계에서 관찰되는 적자생존의 유전법칙에 바탕을 둔 탐색 알고리즘이어서(Goldberg, 1989 ; Holland, 1975) 주어진 문제에 대한 지식을 얻기 힘든 경우에도 큰 효과를 발휘할 수 있다는 장점이 있다. 본 논문에서는 판넬블럭조립공장의 일정계획을 위해 유전알고리즘을 두단계로 나누어 적용하였다. 첫번째 단계는 각 작업장에 계획 대상 블록들을 공평하게 분배하는 단계로서 블록 할당 모듈이 이를 담당하며, 두번째 단계는 각 작업장별로 할당된 블록들을 대상으로 계획기간에 걸쳐서 일자별로 부하를 평균화하는 단계이다.

개발된 일정계획 시스템의 성능평가를 위한 실험을 위해 실제 조선사의 341개 블록에 대한 데이터를

사용하였다. 실험 결과, 실행시간과 부하 평균화 측면에서 현업 전문가에 의해 수행되는 것보다 훨씬 좋은 결과를 얻을 수 있었다. 전문가에 의해서 일주일 정도 소요되던 일정계획이 SUN Sparc Station 20을 사용하여 개발된 시스템에서는 2.5시간 정도 소요되었으며 부하 평균화 수준도 수작업에 의한 일정계획 결과보다 훨씬 개선되었음을 확인할 수 있었다.

## 2. 유전알고리즘의 개관

### 2.1. 유전알고리즘의 원리

유전알고리즘에서는 후보해를 하나의 bit string으로 부호화하여 표현하며 다수 후보해들의 집합으로 이루어진 집단(population)을 가지고 탐색을 시작하여 유전법칙에 따라 세대 변천을 거듭해감에 따라 점차적으로 최적에 가까운 해를 찾아내게 된다. 이러한 유전알고리즘 기법은 문제 해결에 필요한 이렇다할 만한 휴리스틱이 없는 경우의 문제 해결에 탁월한 역량을 발휘하고 있다.

유전알고리즘은 초기 모집단에서 출발해서 유전법칙을 이용하여 새로운 세대로 집단을 계속 진화시켜 나가는데 세대변천은 도태(selection), 교차(crossover), 변종(mutation)의 세가지 유전연산에 의해서 이루어진다. 도태는 집단 내의 각 후보해(string)를 평가하여 적합도(fitness)를 계산한 후 적합도가 높을수록 많은 수의 자기복제를 허용함으로써, 차세대에 그 세력이 확장될 수 있게 한다. 교차는 도태이후 변화된 세력판도하에서 집단내의 후보해들로 하여금 무작위적으로 모두 짝을 짓게 한 후 짝들이 서로 일부 bit string을 교환하게 함으로써, 새로운 쌍의 string을 만들어 낸다. 변종은 집단에 무작위성을 부여하기 위해 string내의 임의의 한 bit를 0에서 1로 혹은 1에서

0으로 바꾸어 준다. 탐색의 방향이 지나치게 무작위적으로 바뀌지 않아야 하므로 변종은 매우 조심스럽게 낮은 확률로 적용된다. 이상의 세가지 기본 유전연산을 바탕으로 보다 복잡하고 고도화된 연산기법들이 개발되어 활용되고 있으나, 이들이 궁극적으로 추구하는 것은 적합도가 높은 string들간의 유사성을 파악하고 선택적으로 잘 조합함으로써 최적의 string, 즉 최적에 가까운 해를 찾고자 하는 것이다.

## 2.2. 유전알고리즘을 이용한 일정계획 연구

Fang 등은 유전알고리즘의 교차와 변종을 조작하여 Job-shop scheduling, Rescheduling, Open-shop scheduling 문제를 벤치마크하였다(Fang *et al.*, 1993). 또한 Bruns는 교차와 변종을 일정계획에 맞게 변형 시켜서 일정계획용 지식이 부가된 유전알고리즘을 적용하였다(Bruns, 1993). 이러한 국외의 연구에서는 주로 유전 연산자를 조작하여 소규모 실험실 수준의 일정계획 문제에 적용하였으며, 실제 일정계획 문제에 적용한 사례는 아직 발표되지 않고 있다.

한편 국내에서는 유전알고리즘을 이용하여 실제계의 일정계획 시스템을 개발한 사례가 있다. 류광렬 등은 실제 자동차 부품공장을 대상으로 유전알고리즘을 적용하여 복잡한 일정계획 문제를 성공적으로 풀 수 있음을 보였다. 즉, 문제해결의 가장 중요한 관건이 되는 유전자 표현방법으로 직접적인 방식과 간접적인 방식의 절충행을 택하여 최적에 가까운 일정계획을 구하고 있어 충분히 실용화가 가능함을 확인하였다(류광렬 등, 1995). 김석준은 수행시간의 합을 최소화시키는  $1:n$ 의 작업일정계획 문제를 해결하기 위하여 병렬 유전알고리즘을 이용하였다(김석준, 1992).

그러나 본 논문의 대상인 조선공업에서의 판넬조

립공장을 대상으로 한 매우 복잡한 일정계획 문제에 유전알고리즘을 적용한 사례는 전혀 없는 실정이다.

## 3. 대상문제

### 3.1. 판넬블럭조립공장의 일정계획

선박의 건조 과정은 생산계획 초기단계의 블럭분할(Block division)로부터 시작된다. 블럭분할에 의해 하나의 선박은 수백개의 블럭으로 분할되며 각각의 블럭은 적게는 한 개, 많게는 네 개의 단위블럭들로 구성된다. 각 블럭은 블럭조립공장에서 독립적으로 조립 생산되고 생산된 몇개의 블럭들은 선행탑재 과정을 통해 보다 큰 블럭으로 만들어지며 이렇게 만들어진 블럭들은 도크로 옮겨져 탑재 과정을 통해 하나의 완전한 선박으로 조립된다(Cho *et al.*, 1996 ; Lee *et al.*, 1995).

조선에서의 생산 일정계획은 계획시점과 계획기간의 범위에 따라 대일정계획, 중일정계획, 작업일정계획 등으로 분류된다. 대일정계획은 장기계획으로서 유사선종의 정보를 이용하여 수립되는데, 대개 장기적 관점에서의 부하 분석을 통해 주요 자재관리, 설비 및 인력관리용으로 활용된다. 중일정계획은 블럭분할 정보를 이용하여 수립되며, 작업장간의 부하 조정과 강재발주 그리고 강제적재계획을 위해 사용된다. 마지막으로 작업일정계획은 각 블럭의 공작도 정보를 이용하여 수립되며 세부 작업장 배정을 통한 작업장간 부하 평준화와 각 블럭의 착수일자 및 완료일자 조정을 통한 일자별 부하 평준화를 최적화하여 실제 현장에서의 작업관리를 위해 사용된다. 본 논문에서는 상기 생산 일정계획중에서 작업일정계획을 대상으로 한다.

작업일정계획 시스템은 중일정계획 시스템으로부

더 1개월간의 조립대상 블럭들과 각 블럭에 대한 착수일자 및 완료일자를 입력받고 공정설계 시스템 (Cho *et al.*, 1996)의 결과로 생성된 공수를 입력받는다. 착수일자로부터 완료일자까지의 기간을 흔히 공기라 부르며, 공수는 각 블럭을 완성하는데 투입되어야 할 manhour를 의미한다. 각 블럭에 대한 일일 작업공수는 공수를 공기로 나눔으로써 산출된다. 예를 들어, 공기가 7일인 블럭의 공수가 140일 경우에는 계획상 하루에 20공수씩 작업하는 것으로 계산된다. 단, 일정계획 시스템 사용자의 판단에 따라 토요일 작업시간은 평일 작업시간과 다르게 책정될 수 있다.

본 논문의 작업일정계획 시스템은 중일정계획에 의해 주어진 작업물량에 대하여 여러가지 제약조건을 만족시키면서 세부 작업장 사이의 부하 평준화와 계획기간 동안의 일자별 부하 평준화를 이룰 수 있는 최적의 작업일정계획을 수립하는 것을 목표로 하고 있다.

대개 월간계획 대상 물량은 340여개 정도의 블럭들로 구성되는데, 이 블럭들에 대해 각 작업장의 여건과 작업인원의 능력을 고려하여 일정계획을 수립하여야 한다. 현재 선각내업의 작업일정계획은 일정계획 담당자의 경험에 의존해서 수립되고 있다. 그런데 계획 대상 블럭의 수가 많고 선행 작업장의 물량 수급 상황 등 다양한 제약조건에 대한 고려가 어렵기 때문에 각 작업장마다 전체적인 부하를 균등하게 배분하기가 어렵고 일자별 부하 평준화도 제대로 이루어지지 않고 있는 실정이다. 수작업에 의한 일정계획 결과를 보면 1개월의 일자별 부하가 상당히 고르지 못하다. 그래서 잔업을 하는 작업장이 있는가 하면 경우에 따라서는 작업물량이 부족해서 익일의 물량을 앞당겨 생산하는 작업장도 생기게 된다. 결국 좋지 못한 일정계획으로 인해 회사 전체적으로 막대한 손실이 발생하게 된다.

### 3.2. 대상 공장의 구성

본 논문에서는 실제 조선사의 판넬블럭조립공장을 대상으로 하여 일정계획 시스템을 개발하였다. 이 조립공장은 배의 구조중 주로 중앙부 평블럭을 조립하는 공장으로서 크게 3개의 작업장(A, B, C)으로 구성된다. 그림 1에서 볼 수 있는 바와 같이 A 작업장은 여러개의 판(plate)을 용접하는 자동판계장과 블럭을 조립하는 2개의 흐름라인 작업장(A#1, A#2)으로 구성된다. B 작업장은 자동판계장, 2개 흐름라인 작업장(B#1, B#2) 그리고 1개의 고정정반(B#3)으로 구성되며, C 작업장은 4개의 고정정반(C#1, C#2, C#3, C#4)으로 구성되어 있다. 각각의 흐름라인 작업장 혹은 고정정반은 각각의 작업팀이 담당하고 있다. 따라서 A 작업장에는 2개의 작업팀이 있으며, B 작업장과 C 작업장에는 각각 3개, 4개의 작업팀이 있다.

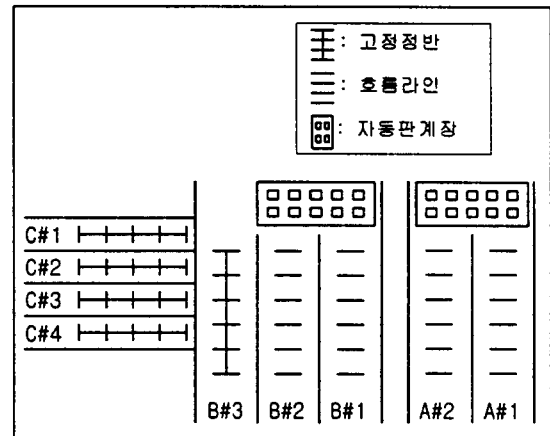


그림 1. 대상 공장 배치도

### 3.3. 작업일정계획의 목표 및 제약조건

그림 2는 중일정계획 시스템으로부터 팀의 구분 없이 A 작업장에 할당된 블럭들의 일자별 부하 그래프의 예를 보여주고 있다. 현재 중일정계획에서는

A, B, C 작업장 전체로 물량을 할당하며 팀별로는 할당하지 않고 있다. 따라서 그림 2는 각 팀으로 블럭들을 배정하기 전의 A 작업장 전체 물량에 대한 중일정계획 결과이기 때문에 일자별 부하의 측면에서는 만족할만한 결과를 보여주지 못하고 있다.

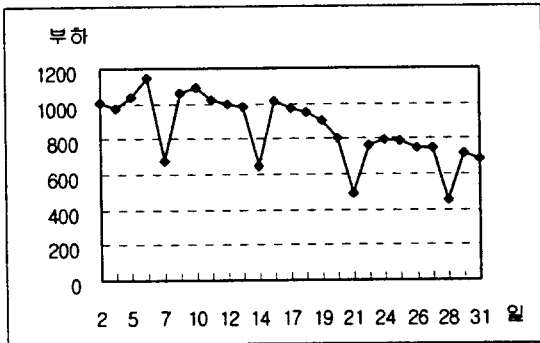


그림 2. 중일정계획에 의한 A 작업장의 부하 그래프

따라서 작업일정계획 시스템은 각 작업장별로 할당된 물량을 해당 작업팀에 공평하게 배분하는 역할을 담당한다. 이러한 목표를 달성하는 과정에서 반드시 고려해야 할 다수의 제약조건들이 존재하는데, 본 연구에서는 이러한 제약조건들을 크게 세가지로 분류하였는바, 기술적 제약조건(Technical constraints), 선호되는 제약조건(Preferred constraints), 조정가능 제약조건(Relaxable constraints) 등이 그것이다.

기술적 제약조건은 각 작업장의 시설, 장비 등과 관련된 제약조건으로 반드시 지켜져야 하는 것이다. 예를 들면, 자동판계장의 일일판계량과 흐름 불가능 블럭에 관한 제약조건이 있다. 선호되는 제약조건은 가능하면 지켜야 하는 조건으로 생산효율과 연관된 제약조건이다. 이에겐 전문화 작업장과 PS 제약조건이 있다. 조정가능 제약조건은 일정계획 담당자가 회사의 정책에 따라서 조정할 수 있는 제약조건으로 착수일자, 완료일자, 1일 작업시간 등이 있다. 각각에

대한 세부 내용은 다음과 같다.

- 착수일자 및 완료일자 조정 : 각 블럭의 착수일자는 선행작업장과의 여유일이 2일 정도이며 완료일자는 후행작업장과의 여유일이 1일 정도이다. 따라서 착수일자는 2일, 완료일자는 1일 범위 내에서 조정이 가능하다. 단, 각 블럭의 착수일자는 작업일정계획 기간을 벗어날 수 없다. 착수일자 및 완료일자 조정은 일자별 부하 평준화를 위한 가장 중요한 요소이다.

- 자동판계장의 일일판계량 : 블럭의 작업기반이 되는 판을 용접하는 작업을 판계라 부르며 이 작업은 자동판계장에서 이루어진다. A 자동판계장의 일일 최대판계 개수는 9개이고 B 자동판계장의 일일 최대판계 개수는 16개로서 B 작업장과 C 작업장의 조립작업을 위한 물량을 공급한다. 그리고 각 블럭의 판계 개수는 블럭의 형상에 따라 1~4개 정도이다. 착수일자 및 완료일자 조정을 통한 부하 평준화가 잘 되더라도 자동판계장의 일일 생산 능력을 초과하는 일정계획이 수립되어서는 안 된다.

- 흐름 불가능 블럭 : 블럭 형상 문제로 인해 흐름라인을 따라 흐를 수 없는 블럭들은 흐름 라인으로 배정되어서는 안 된다. 따라서 B 작업장의 경우 흐름 불가능 블럭은 고정정반 작업장인 B#3으로 배정되도록 해야 한다. 블럭 형상에 대한 정보는 공정설계 시스템의 공정설계 결과로 생성된다.

- 전문화 작업장 : 블럭 형상이 corrugation인 블럭은 C#1 팀에서 작업되어야 한다.

- PS 제약조건 : 선박은 대개 좌우 대칭형이므로 선박을 구성하는 블럭들에도 대칭되는 쌍이 있다. 쌍을 이루는 블럭들에는 동일 블럭 이름을 부여하되 이름 끝에 좌현과 우현을 나타내는 P(Port)와 S(Starboard)를 붙여 각각을 구별하고 있다. 이들 두 블럭은 항상 서로 다른 라인에서 동시에 조립되는 것이 바람직하다.

결국 작업일정계획의 목표는 이상의 제약조건을 만족하면서 팀별 부하 평준화와 각 팀내에서 일자별 부하 평준화를 도모하고자 하는 것이다.

## 4. 판넬블럭조립공장의 일정계획 시스템

### 4.1. 시스템 구성

본 논문의 작업일정계획 시스템은 그림 3과 같이 데이터 생성 모듈, 블럭 할당 모듈, 메인 스케줄링 모듈, 출력 생성 모듈로 구성된다.

데이터 생성 모듈은 일정계획 수립에 필요한 각종 데이터를 수집, 변환하는 모듈로서 크게 타시스템으로부터 제공되는 정보와 사용자가 직접 입력하는 정보를 받아들인다. 타시스템으로부터 제공되는 정보는 중일정계획 시스템과 공정설계 시스템으로부터 자동으로 입력되며 구체적인 입력자료는 표 1에 제시되어 있다. 사용자가 입력하는 자료에는 작업일수, 출근율, 작업인원, 1일 작업시간(평일/토요일), 목표

능률 등이 있다. 출력 생성 모듈은 바차트, 부하 분석표 등의 출력물을 생성한다.

본 논문에서 유전알고리즘이 적용된 모듈은 블럭 할당 모듈과 메인 스케줄링 모듈이다. 블럭 할당 모듈에서는 상위단계 유전알고리즘, 메인 스케줄링 모듈에서는 하위단계 유전알고리즘을 적용하였다. 상위단계 유전알고리즘에서는 중일정계획의 착수일자와 완료일자를 준수하면서 1일 단위로 팀간의 부하를 평준화하기 위해 각 블럭들을 팀으로 배정하기 때문에 월 전체적인 일자별 부하 평준화가 이루어지는 것이 아니라, 일자를 기준으로 각 팀의 작업자 1인에게 할당되는 작업량이 비슷하도록 계획한다. 하위단계 유전알고리즘에서는 각 블럭의 착수일자와 완

표 1. 타 시스템으로부터 제공되는 입력자료

입력자료	자료 제공원
호선	공정설계시스템
선종	
선형	
블럭 번호	
블럭의 위치	
블럭의 조립단계	
관계 유형	
관계수	
블럭 중량	
블럭 길이	
블럭 폭	
블럭 높이	
블럭 형상	
공수	
흐름작업 가능 여부	
공기	중일정계획시스템
대조 이관 블럭	
중일정계획	

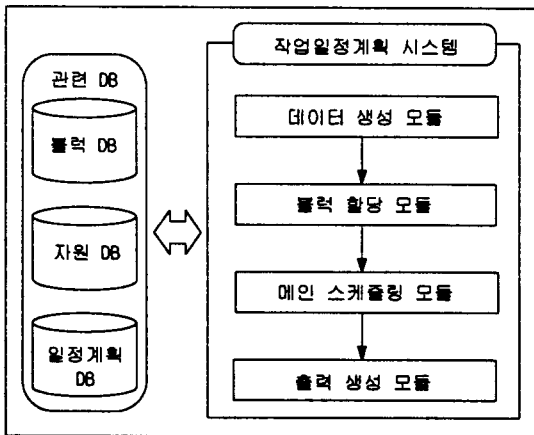


그림 3. 작업일정계획 시스템 구성도

료일자를 허용범위내에서 조정함으로써 월 전체적으로 일자별 부하 평준화를 이루도록 한다.

이러한 블럭 할당 모듈과 메인 스케줄링 모듈에 대한 자세한 내용은 다음절에서 계속해서 설명하기로 한다.

#### 4.2. 계층적 유전알고리즘의 적용

대상 문제인 평블럭조립을 위한 작업일정계획은 팀별 부하 평준화와 각 팀내에서의 일자별 부하 평준화라는 두 가지 목표를 가지고 있다. 본 논문에서는 이러한 두가지 목표를 달성하기 위한 모듈을 각각 독립적으로 구현하였다. 상위단계 유전알고리즘인 블럭 할당 모듈에서는 팀별 부하 평준화를 담당하고 하위단계 유전알고리즘인 메인 스케줄링 모듈에서는 각 팀내에서의 일자별 부하 평준화를 담당한다. 단일 유전알고리즘을 이용하여 이 문제를 해결할 경우의 가장 큰 문제점은 각 블럭의 팀 배정과 착수일자 및 완료일자 조정을 한꺼번에 생각해야 하므로 탐색공간이 너무 커서 만족할만한 해를 얻기 위한 계산시간이 매우 길어진다는 것이다. 계층적 유전알고리즘은 이러한 단점을 보완하여 팀별 부하 평준화를 먼저 최적화한 다음 일자별 부하를 평준화함으로써 허용시간 내에 만족할만한 해를 탐색할 수 있다.

상위단계의 유전알고리즘은 A, B, C 각각의 작업장에 대한 계획이 독립적으로 수행되지만 하위단계의 유전알고리즘은 A 작업장과 B, C를 묶은 작업장을 대상으로 두 개의 유전알고리즘이 독립적으로 수행된다. 이것은 자동관계장의 물량 수급에 관계된 문제로서 A 작업장의 물량을 공급하기 위한 자동관계장은 따로 있으나 B와 C 작업장은 하나의 자동관계장을 공유하고 있기 때문이다.

블럭 할당 모듈의 목표는 팀별 부하 평준화이다. 블럭 할당 모듈은 계획 대상 블럭들을 팀 배정원칙과 전문화 작업장 등의 제약조건을 고려하고 부하를 평준화하여 팀별로 할당한다. 예를 들면 A 작업장은 A#1, A#2라는 두 팀으로 구성되므로 A 작업장의 계획 대상 블럭들을 A#1팀과 A#2팀으로 각각 배정한다. 각 일자별로 부하를 평준화하기 때문에 팀 배정후의 부하는 월 전체적으로 평준화가 이루어지지 않지만 그림 4에서와 같이 1일을 기준으로 봤을 때 각 팀의 부하는 공평하게 배분됨을 알 수 있다. 다음으로는 블럭 할당 모듈에서 염색체의 구성, 수행과정 및 평가에 대해서 설명한다.

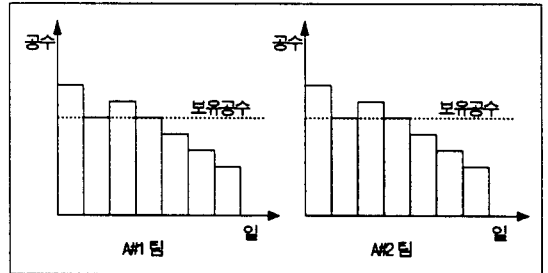


그림 4. 블럭 할당 모듈 수행후의 A 작업장의 부하상태

##### ① 염색체의 구성

블럭 할당 모듈에서의 염색체는 그림 5와 같이

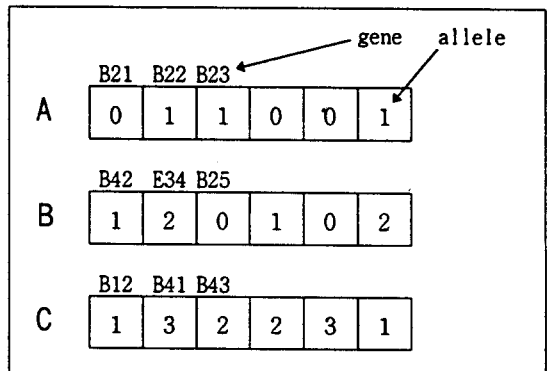


그림 5. 블럭 할당 모듈의 염색체 구성

##### 1) 상위단계 : 블럭 할당 모듈

블럭을 나타내는 gene으로 구성된다. A, B, C 각 작업장의 염색체는 월간 작업일정계획 대상 블럭수 만큼의 gene으로 구성된다.

각 작업장마다 allele의 구성은 팀수에 따라서 다르게 결정된다. A 작업장은 2개의 팀으로 구성되므로 1 bit이면 되지만, B, C 작업장은 각각 3개팀, 4개 팀으로 구성되므로 2bit로 나타낸다.

② gene map의 구조와 사상

그림 6에는 allele에 의해서 어떻게 gene이 실제 블럭과 사상되는지를 나타내고 있다.

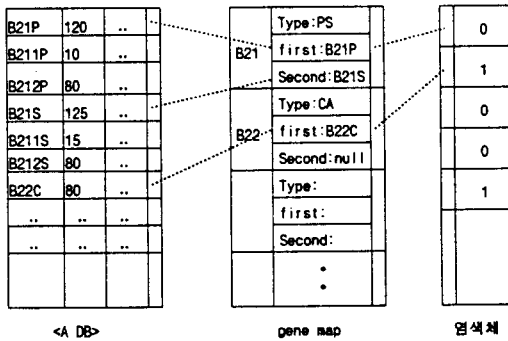


그림 6. gene map의 구조와 사상

그리고 표 2에는 gene map에 의한 작업장 배정에 관한 원칙이 정리되어 있다. 예를들어 A작업장의 경우 allele이 0이고, 블럭의 type이 “PS”인 경우에는 first인 B21P 블럭은 A#1에 배정되고, second인 B21S 블럭은 A#2에 배정된다. 반대로 allele이 1이고, 블럭의 type이 “PS”인 경우에는 first인 B21P는 A#2에 배정되고, second인 B21S 블럭은 A#1에 배정된다. B22 블럭의 경우에는 allele이 1이고, 블럭의 type이 “CA”인 경우에는 first에 연결된 B22C는 A#2에 배정된다. 만약에 allele이 0이면 first에 연결된 B22C는 A#1에 배정된다. 이렇게 엇갈리게 사상을 하는 이유는 PS 제약조건을 만족시켜 좌현(P) 블럭과 우현(S) 블럭이 서로 다른 두개의 작업장에서 동시에 조립될

수 있도록 하기 위함이다.

표 2. 작업장 배정 원칙

작업장	allele	gene map		
		type	first 작업장	second 작업장
A	0	PS	A#1	A#2
		CA	A#1	null
	1	PS	A#2	A#1
		CA	A#2	null
B	0	PS	B#1	B#2
		CA	B#1	null
	1	PS	B#2	B#1
		CA	B#2	null
	2	PS	B#3	B#3
		CA	B#3	null
C	0	PS, CA	C#1	null
	1	PS, CA	C#2	null
	2	PS, CA	C#3	null
	3	PS, CA	C#4	null

③ 초기 모집단의 생성

초기 모집단은 중일정계획의 결과를 근거로 구성하였다. 그 이유는 완전히 새로이 모집단을 구성하는 것보다 중일정계획을 기반으로 하는 것이 보다 빠르게 원하는 해를 구할 수 있는 것으로 판단하였기 때문이다.

④ 평가(Evaluation)

A 작업장은 유전자의 해석에 따라 A#1팀 또는 A#2팀으로 배정될 수 있고 PS 제약조건을 만족하기 위해 같은 이름의 블럭은 항상 다른 팀에 배정되도록 지정된다. 각 블럭의 팀 배정 결과에 따라 그림 7과 같이 각 팀의 load builder를 이용하여 1개월간 부하를 쌓는다.



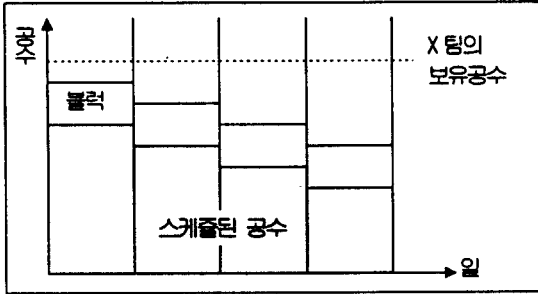


그림 7. X팀의 load builder

B 작업장은 유전자의 해석에 따라 B#1팀, B#2팀, B#3팀으로 배정될 수 있으며 흐름 불가능 블럭 제약조건을 만족하기 위해 흐름 불가능 블럭들은 유전알고리즘을 적용하기 전에 미리 추출하여 B#3팀으로 배정함으로써 유전알고리즘의 탐색대상에서 제외한다. 그리고 PS 제약조건을 만족하기 위해 A 작업장과 마찬가지로 같은 이름의 블럭은 항상 다른 팀에 배정되도록 하였다. C 작업장은 유전자의 해석에 따라 C#1팀, C#2팀, C#3팀, C#4팀으로 배정될 수 있으며 전문 작업장 제약조건을 만족하기 위해 블럭 형상이 corrugation인 블럭들을 미리 C#1팀에 배정하여 유전알고리즘의 탐색대상에서 제외한다.

블럭 할당 모듈에서의 평가함수는 load builder의 결과에 따른 각 팀별로의 부하 평균화 정도를 평가한다. 평가함수는 식(1)과 같다. 이 평가함수는 load builder의 결과에 대한 1일단위의 1인당 부하가 팀간에 어느정도 평균화되었는가를 검사한다.

$$OF_{ba} = \sum_{i=1}^{TWD} \left| \sum_{j=1}^{TWN} \left( \frac{\sum_{j=1}^{TWN} (L_{ji}/NM_j)}{TWN} - \frac{L_{ji}}{NM_j} \right) \right| \quad ..(1)$$

i : 작업일

j : 팀

$L_{ji}$  : j 팀의 i일 부하

$NM_j$  : j 팀의 작업인원

TWD : 총작업일수

TWN : 총작업팀수

다음으로는 제약조건외의 위배정도에 따라 표 3과 같이 벌점(penalty)을 부여한다. 벌점 부여 점수의 산정근거는 결과값을 산출해가면서 여러번의 실험을 한 후에 좋은 해를 유도할 수 있도록 부여하였다.

표 3. 제약조건 위배에 따른 벌점

B#1		C#1,2	
제약조건	벌 점	제약조건	벌 점
블럭중량	1,000	블럭길이	1,000
		블럭 폭	1,000
		선관계 블럭 배정	1,000

이상의 평가함수의 결과값과 벌점을 합해서 적합도 (fitness)를 구한다. 적합도 수식은 식(2)와 같다.

$$FITNESS_{ba} = Min(OF_{ba} + P_{ba}) \quad .. (2)$$

$FITNESS_{ba}$  : 블럭 할당 모듈의 적합도

$OF_{ba}$  : 블럭 할당 모듈의 평가함수 결과 값

$P_{ba}$  : 블럭 할당 모듈의 벌점

적합도가 최소인 값을 찾을 때까지 도태(selection), 교차(crossover), 변종(mutation)의 유전 연산을 계속적으로 수행해서 최적에 가까운 해를 도출한다.

## 2) 하위단계 : 메인 스케줄링 모듈

메인 스케줄링 모듈의 목표는 팀내에서의 일자별 부하를 평균화하는 것이다. 팀에 할당된 블럭들을 대상으로 스케줄링 전략과 자동판계장의 제약조건 등을 고려하여 각 블럭의 착수일자과 완료일자를

조정한다. 그림 8과 같이 블럭 할당 모듈 수행후에는 팀간의 부하 평준화가 어느정도 이루어지더라도 월 전체적으로는 일자별 부하가 고르지 못한 상태이다. 메인 스케줄링 모듈에서는 블럭 할당 모듈로부터 가장 좋은 검색체를 입력 받아 스케줄 전략을 반영하여 일자별 부하를 평준화시키고, 자동판매장과 각 팀의 블럭들에 대한 착수일자와 완료일자를 확정한다. 따라서 그림 8의 메인 스케줄링 모듈 수행 후의 형태와 같이 일자별 부하 평준화를 이루는 것이 메인 스케줄링 모듈의 목표이다.

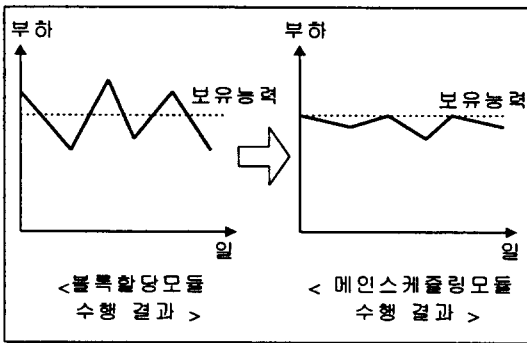


그림 8. 메인 스케줄링의 목표

계속해서 메인 스케줄링 모듈에서 검색체의 구성, 수행과정 및 평가에 대해서 설명한다.

① 검색체의 구성

그림 9의 검색체 구성은 표 4의 착수완료일 조정

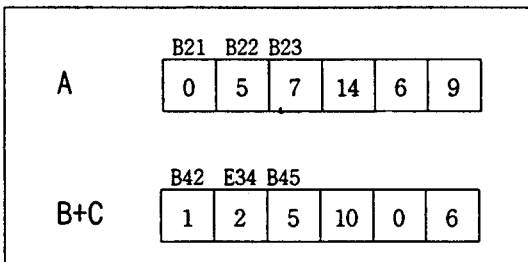


그림 9. 메인 스케줄링 모듈의 검색체 구성

전략값을 반영하여 구성된다. 이러한 전략은 공기가 -2일 % %3일 정도의 범위내에서 조정된다. A 작업장의 검색체는 A 작업장에 할당된 계획 대상 블럭 수만큼의 gene으로 구성되며, B와 C 작업장의 검색체는 B와 C 작업장에 할당된 블럭수의 합에 해당되는 블럭수만큼의 gene으로 구성된다.

② gene map의 구조와 사상

그림 10은 allele에 의해서 어떻게 gene이 실제 블럭과 사상되는지를 나타내고 있다. 그리고 표 4에는 착수완료일 조정 전략값이 제시되어 있다. 예를들어 A 작업장의 경우, allele이 0이면 B21 블럭의 착수일자는 2일이 앞당겨져 작업이 이루어지고, 완료일자는 1일이 앞당겨진다. 그래서 공기는 1일이 늘어난다.

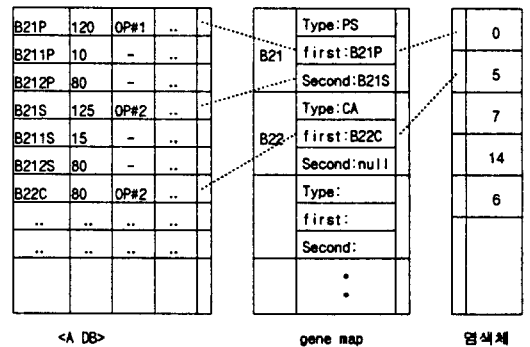


그림 10. gene map의 구조와 사상

③ 초기 모집단의 생성

블럭 할당 모듈에서 생성된 최적해 gene이 메인 스케줄링 모듈에서의 초기 모집단이 된다.

④ 평가

메인 스케줄링 모듈에서는 앞에서 밝힌 바와 같이 자동판매장의 최대능력분석을 위해 A 작업장과 B, C를 묶은 작업장으로 나누어 두개의 독립적인 유전

알고리즘이 수행된다. 작업장에 관계없이 착수완료일 조정 전략은 착수일자 및 완료일자 조정 제약조건에서 밝힌 바와 같이 착수일 조정 5가지(-2, -1, 0, 1, 2)와 완료일 조정 3가지(-1, 0, 1)로 총 15가지 경우가 나올 수 있다. 또한 PS 제약조건을 준수하기 위해서 쌍을 이루는 블럭들은 착수일자와 완료일자의 조정 정도가 같도록 하여 같은 날에 착수 및 완료를 할 수 있도록 하였다. 염색체의 해석에 따라 각 팀의 load builder에 1개월간 부하를 쌓게 된다.

표 4. 착수완료일 조정 전략

allele	착수일자 조정	완료일자 조정	실제 조정폭(일)
0	-2	-1	+1
1	-2	0	+2
2	-2	1	+3
3	-1	-1	0
4	-1	0	+1
5	-1	1	0
6	0	-1	-1
7	0	0	0
8	0	1	+1
9	1	-1	0
10	1	0	-1
11	1	1	0
12	2	-1	+1
13	2	0	-2
14	2	1	-1

평가함수는 식(3)과 같으며 블럭 할당 모듈에서 팀간의 부하 평준화 정도를 검사한 것과는 달리 한 팀 내에서 일자별로의 부하 평준화가 어느 정도 좋은가를 검사한다.

$$OF_{ms} = \sqrt{\sum_{j=1}^{TWN} \frac{\sum_{i=1}^{TWD} (AL_{ji} - AC_{ji})^2}{TWD}} \left\{ \sum_{j=1}^{TWN} \frac{\sum_{i=1}^{TWD} (AL_{ji} - AC_{ji})^2}{TWD} \right\}^2 \quad ..(3)$$

- $i$  : 작업일
- $j$  : 팀
- $AL_{ji}$  :  $j$  팀의  $i$ 일 부하
- $AC_{ji}$  :  $j$  팀의  $i$ 일의 보유공수
- $TWD$  : 총작업일수
- $TWN$  : 총작업팀수

즉, 그림 11과 같이 load builder의 일자별로 보유공수와 실제 쌓인 공수의 차를 최소화시킴으로써 전체적으로 부하 평준화를 이룰 수 있도록 조정한다.

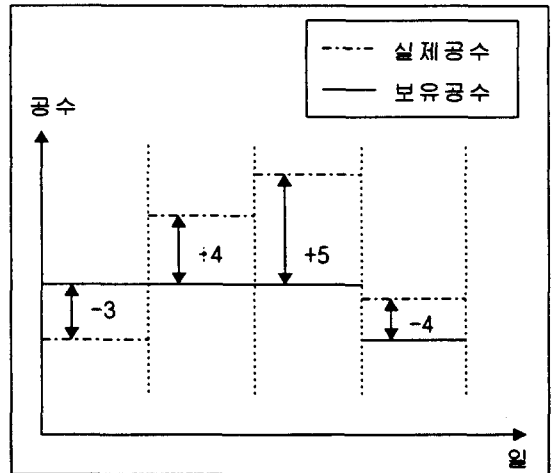


그림 11. 메인 스케줄링 모듈의 부하 평가

자동판계장의 최대능력을 초과한 계획에 대한 별점은 위배정도에 따라 표 5와 같이 부과하였다. 이 값은 여러번의 실험을 통해서 결정되었다. 최종 평가값은 평준화 정도를 검사하기 위한 평가값과 자동판계장의 제약조건 위배에 따른 별점을 합하여

계산한다.

표 5. 제약조건 위배에 따른 벌점

작업장	제약조건	벌 점
A	자동관계능력 초과	초과 판넬수×12
B+C	자동관계능력 초과	초과 판넬수×50

이상의 평가함수의 결과값과 벌점을 합해서 적합도를 구한다. 적합도 수식은 식(4)와 같다.

$$FITNESS_{ms} = Min(OF_{ms} + P_{ms}) \quad .. (4)$$

$OF_{ms}$  : 메인 스케줄링 모듈의 평가함수

$P_{ms}$  : 메인 스케줄링 모듈의 벌점

적합도가 최소인 값을 찾을때까지 유전 연산을 계속 수행한다. 최적해로 판명된 gene이 실제 블럭과 사상되어서 착수일자와 완료일자가 확정된다.

## 5. 실험

실험 대상 물량으로는 실제 조선사의 95년 10월분 데이터를 사용하였으며 계획기간동안의 블럭수는 A 작업장 96개, B 작업장 97개, C 작업장 148개이다. 프로그램 개발 환경은 Sun Sparc Workstation 20의 UNIX 운영체제에서 유전알고리즘 개발도구인 GAucsd(Grefenstette, 1992)와 C 언어를 사용하였다.

1개월 물량을 계획하는데 소요되는 시간을 살펴 보면 집단크기가 2000, 세대수가 300인 경우의 블럭 할당 모듈 수행 시간은 0.5시간이 소요된다. 메인 스케줄링 모듈 수행시간을 보면 집단크기가 1000이고, 세대수가 500인 경우에는 2시간이 소요된다. 집단과 세대의 크기는 클수록 좋은 결과를 얻을 수

있지만 검색시간이 많이 걸리므로 적당한 집단크기와 세대수를 선정하도록 하였다. 그리고 유전알고리즘을 수행할 때 사용되는 비율 중에서 교차비율(crossover rate)은 0.98, 변종비율(mutation rate)은 0.000053으로 설정하여 실험을 하였다. 이 비율은 여러번의 실험을 통하여 설정된 값이다.

A 작업장의 A#1팀에 대한 블럭 할당 모듈 수행 후의 부하 그래프와 메인 스케줄링 모듈 수행후의 부하 그래프를 보유공수 및 최대 보유공수와 비교한 결과는 그림 12와 같다. A#1팀의 블럭 할당 모듈 수행후에는 부하가 초순에는 높게 나타나다가 하순 경우에는 적정 능력보다 낮게 나타나고 있다. 하지만 메인 스케줄링 모듈을 수행하고 난 이후에는 최대 능력이하로 부하의 평준화가 이루어졌음을 알 수 있다.

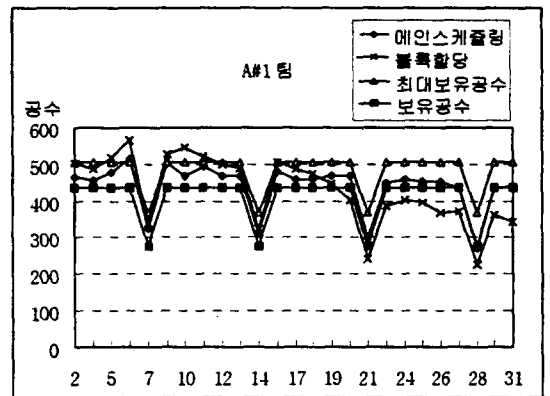


그림 12. A#1팀의 일정계획 결과 부하 그래프

A 작업장의 일정계획 결과, 자동관계장의 부하를 보면 그림 13과 같다. 그래프 상에서는 일일 최대 판계수보다 낮게 나타나고 있어 자동관계장 제약조건을 만족하고 있음을 알 수 있다.

계층적 유전알고리즘을 이용한 일정계획 시스템의 수행 결과와 기존방식에 의한 각 작업장의 부하 평준화 정도를 비교하기 위해 표 6에는 수작업에 의한

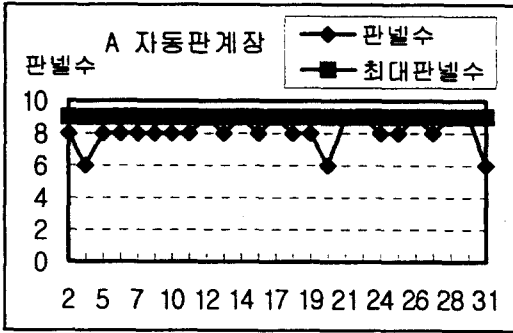


그림 13. A 작업장의 자동판계장 부하

방식과 단계별 유전알고리즘 수행 결과 계산된 표준편차를 표기하였다. 수작업에 의한 방식에 비해서 메인 스케줄링 후의 표준편차는 상당히 낮은 값으로 나타나고 있다. 표준편차가 낮다는 의미는 평균에서 흩어짐의 정도가 적다는 것으로서 부하 평균화가 좋을 것을 나타내는 것이다. 따라서, 하위단계 유전알고리즘을 이용한 일정계획 시스템을 수행한 후에 좋은 일정계획이 수립되고 있음을 알 수 있다.

표 6. 일정계획 결과 비교(표준편차)

팀	수작업 일정계획	일정계획 시스템	
		블럭할당모듈	메인스케줄링모듈
A#1	95.63	93.79	66.82
A#2	91.74	92.18	66.20
B#1	69.69	60.62	57.95
B#2	74.20	57.44	55.53
B#3	54.67	58.86	49.77

예를 들어, A 작업장의 A#1팀에 대한 수작업에 의한 방식과 메인 스케줄링 모듈 수행후의 부하 평균화 정도에 대한 부하 그래프를 살펴보면 그림 14와 같다. 수작업에 의한 경우에 있어서는 토요일을 제외한 변동폭이 600에서 375으로 나타나고 있으며, 시스템에 의한 경우에는 524에서 415로 나타나고

있다. 따라서 메인 스케줄링 모듈 수행후의 부하 평균화가 수작업에 의한 방식에 비해서 상당히 좋게 나타남을 알 수 있다.

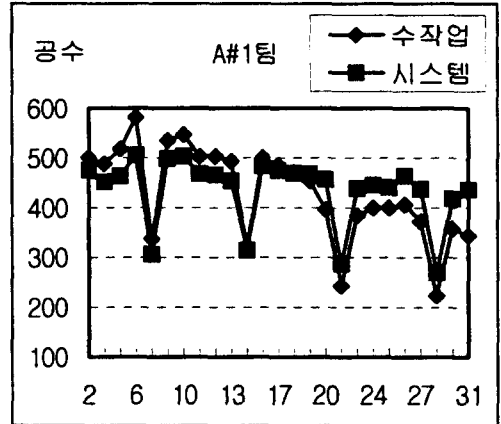


그림 14. 본 시스템과 기존방식의 결과 비교

## 6. 결 론

본 논문에서는 선각블럭조립공장의 일정계획 문제를 해결하기 위한 유전알고리즘의 적용 방안을 제시하였고 이를 실제 조선사의 판넬블럭조립공장에 대한 일정계획 문제에 적용해 봄으로써 기존의 수작업을 대체할 수 있는 시스템의 실용화가 가능함을 보였다. 특히, 팀별 부하 평균화와 각 팀내에서의 일자별 부하 평균화라는 두가지 목표에 대해 각각의 문제를 해결할 수 있는 유전알고리즘을 설계하고 상위단계에서는 팀별 부하 평균화의 최적화 과정을, 하위단계에서는 일자별 부하 평균화 과정을 거침으로써 비교적 짧은 시간 내에 복잡한 문제에 대한 좋은 해를 탐색할 수 있음을 보였다.

앞으로의 연구를 통해 실행시간을 더 단축시킬 수 있는 알고리즘 개발과 함께 Hill climbing 등의 휴

리스트틱 방법을 개발하여 서로의 비교 연구가 이루어져야 할 것으로 생각된다. 그리고 계층적 구조의 일정계획 문제에서 필연적으로 야기되는 Reactive scheduling 문제에 대한 해결책을 본 논문에서는 제시하지 못했는데 이에 대한 연구도 뒤따라야 할 것으로 생각된다.

### 참 고 문 헌

1. 김석준, 이채영, "작업 일정계획문제 해결을 위한 유전알고리즘의 응용," *한국경영과학회지*, 제17권, 3호(1992), 1-12.
2. 류광렬, 박성훈, 김갑환, 홍봉희, "유전 알고리즘을 이용한 자동차 부품공장의 작업 일정계획 수립," *Proceedings of The 21st KISS Fall Conference*, 제21권, 2호(1995), 335-338.
3. Bruns, R., "Direct Chromosome Representation and Advanced Genetic Operators for Production Scheduling," *ICGA-93 Conference* (1993), 352-359.
4. Cho, K. K. et al., Computer Aided Process Planning System for Block Assembly Shop Using CBR, *Proceedings of The 24<sup>th</sup> North American Manufacturing Research Conference*, Ann Arbor, Michigan, (1996), 157~162.
5. Fang, H., Ross, P., Corne, D., "A promising Genetic Algorithm Approach to Jop-Shop Scheduling, Rescheduling, and Open-Shop Scheduling Problems," *ICGA-93 Conference* (1993), 375-382.
6. Goldberg, D.E., *Genetic Algorithms in Search, Optimization & Machine Learning*, Addison-Wesley, 1989.
7. Grefenstette, J.J., and N. N. Schraudolph, *A Users Guide to GAUCSD 1.4*, University of California, San Diego, July, 1992.
8. Holland, J.H., *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor, 1975.
9. Lee, J. K. et al., Intelligent Scheduling Systems for Shipbuilding, *AI Magazine*, Vol.16, No.4 (1995), 78-94.