

## 유전알고리즘을 이용한 강의시간표 작성 시스템 설계

강명주<sup>o</sup>

<sup>o</sup>청강문화산업대학 컴퓨터게임과

e-mail: mjkkang@ck.ac.kr

## Design A Timetable System Using A Genetic Algorithm

Myung-Ju Kang<sup>o</sup>

<sup>o</sup>Dept. of Computer Game, ChungKang College of Cultural Industries

### ● 요약 ●

본 논문에서는 유전 알고리즘을 이용한 강의시간표 작성 시스템 설계 방법을 제안한다. 강의시간표는 교과목의 강의시간, 강의실, 교수자 등의 정보를 시간표 테이블에 할당하는 문제이면서 스케줄링 문제이다. 강의시간표로써의 의미를 가지기 위해서는 강의실 중복 회피, 교수의 강의시간 중복회피와 같은 필수 제약조건(Hard Constraint)을 만족해야 한다. 또한, 강의시간표 문제는 NP-Complete 문제로 알려져 있으며 기존의 Exact 알고리즘으로는 최적 해를 구하는 것이 힘들다. 따라서 본 논문에서는 휴리스틱 알고리즘 중의 하나인 유전 알고리즘을 교과목의 강의시간 배정 부분과 강의실 배정 부분을 각각 별도의 부시스템(Sub-system)으로 나누어 적용하는 방법을 제안한다. 실험 결과 강의시간표로써의 의미를 가질 수 있는 필수 제약 조건을 만족시킬 수 있음을 알 수 있었다.

키워드: 시간표(timetable), 유전알고리즘(genetic algorithm)

### I. 서론

강의시간표 문제는 일주일 단위로 교과목, 강의실, 교수자 등의 자원에 대한 다중 제약 조건을 스케줄링하는 다중목적(Multi-Objectives) 문제이면서 자원할당 문제로서 NP-Complete 문제로 알려져 있다. 일반적으로 NP-Complete 문제는 문제의 크기가 커질수록 해결법이 폭발적으로 커짐으로써 기존의 exact 알고리즘으로는 최적의 해를 구하는 것이 어렵다. 따라서 타부탐색(Tabu Search), 유전자 알고리즘(Genetic Algorithm)과 같은 휴리스틱 알고리즘을 이용한 근사 최적해(Near-Optimal)를 구하는 것이 일반적이다.

지금까지 시간표 문제를 해결하기 위한 방법으로 그래프 컬러링(Graph Coloring)을 이용한 방법[1], 수학적 프로그래밍(Mathematical Programming), 타부탐색(Tabu Search) 방법[2], 그리고 유전자 알고리즘(Genetic Algorithm) 방법[3] 등이 사용되어 왔다.

그러나 이러한 기존의 방법들은 개설되는 교과목과 교육환경, 강의실 정보 등을 하나로 처리함으로써 처리속도와 복잡도가 높아진다는 단점이 있다.

따라서 본 논문에서는 이러한 문제를 해결하기 위해 교과목-교수정보 할당 처리와 강의실 할당 처리를 부 시스템(Sub-System)으로 분할하여 처리하는 강의시간표 작성 시스템 설계를 제안한다.

### II. 시간표 문제

시간표 문제는 특정한 조건을 만족시키면서 교과목들을 시간표 테이블 상에 맵핑시키는 문제이다[4]. 시간표 문제는 다양한 제약 조건을 포함하고 있다. 이 제약조건에는 반드시 지켜야 하는 필수 제약조건(Hard Constraint)과 반드시 지킬 필요는 없으나 시간표로써의 효율을 높이기 위해 최대한 만족시켜야 하는 부가 제약조건(Soft Constraint)이 있다.

필수 제약조건에는 강의실 중복이나 교수자의 시간 중복 등을 회피해야 하는 조건들이 있다. 부가 제약조건에는 교수자의 연속 강의 회피, 점심시간 확보 등이 해당된다.

본 논문에서는 C대학의 3개의 전공으로 구성된 스쿨(총 학생 수 1,350명)을 모델로 하였으며, 다음의 제약조건을 만족하는 시간표 작성 시스템을 제안한다.

#### 1) 필수 제약조건

- 하나의 강의실에 강좌를 중복으로 할당하지 말아야 한다.
- 교수는 시간을 중복하여 할당되지 말아야 한다.
- 교과목의 속성과 강의실의 속성이 일치하여야 한다.
- 동일한 학과의 동일한 학년의 교과목들은 중복되지 않아야 한다.
- 교수의 강의 요일과 시간이 미리 지정되면 이를 준수해야 한다.
- 수강신청 인원과 강의실 규모가 적정하여야 한다.

2) 부가 제약조건

- 점심시간이 확보되어야 한다.
- 동일한 교수가 연속 강의를 하지 않아야 한다.
- 교양과목은 다른 과목과 시간이 중복되지 않아야 한다.

III. 시간표 문제를 위한 유전 알고리즘

1. 유전 알고리즘

유전 알고리즘은 해집합으로 구성된 모집단을 이용한다. 모집단은 한 세대를 구성하는 개체들의 집합으로, 유전 연산자를 이용한 진화과정에 적용된다. 초기 세대의 모집단은 랜덤하게 생성되고, 각 개체들은 평가를 거쳐 우수과 열성이 결정된다. 평가 결과가 우수한 개체는 다음 세대로 진화하기 위한 선택 가능성이 높아진다. 선택된 개체들은 유전 연산자에 의해 다음 세대로 진화하게 된다. [그림 1]은 일반적인 유전 알고리즘을 나타내고 있다[5-8].

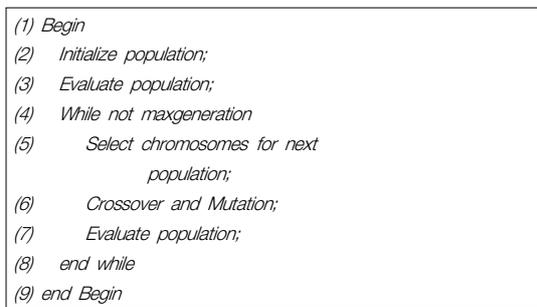


그림 1. 유전자 알고리즘  
Fig. 1. Genetic Algorithm

2. 시간표 문제 해법을 위한 유전 알고리즘

[그림 2]는 교과목 시간 배정 부분과 강의실 배정 부분을 각각 별도의 진화과정을 통해 병렬로 처리하는 방법을 나타내고 있다.

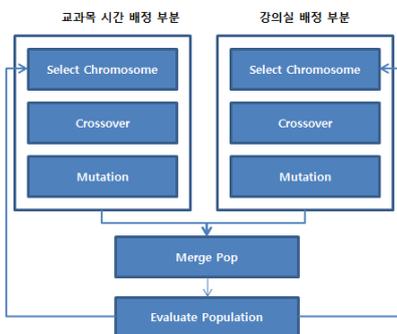


그림 2. 시간표 문제 해법을 위한 병렬 유전 알고리즘  
Fig 2. Parallel Genetic Algorithm for Timetable Problem

3. 염색체 인코딩

유전 알고리즘에서의 염색체는 주어진 문제의 구성요소들을 모두 포함하는 하나의 해집합으로서 비트스트링으로 구성된다.

강의 시간표 작성에 필요한 요소들은 교과목, 강의 요일 및 시간, 강의실, 그리고 교수자 등이다.

따라서 본 논문에서는 교과목-강의시간-교수자 정보와 강의실 정보를 비트스트링 염색체로 인코딩하였다. 병렬처리를 위하여 각각을 별도의 염색체로 구성하여 유전 연산자를 수행하고 근사 최적해로 수렴하여 진화해 가도록 설계하였다.

4. 유전 연산자

4.1 교차 연산

교차 연산은 2개의 부모 염색체를 교배하여 새로운 자식 염색체 2개를 생성하는 연산자이다. 본 연구에서는 1점 교차 연산 방법을 사용한다. 1점 교차 연산은 비트스트링에서 임의의 한 위치를 기준으로 서로 교차하는 방법이다. 다음은 1점 교차 연산 방법을 나타내고 있다[5].

```

p1 : 1111011010 | 11100011010
p2 : 1010001010 | 00001100100

s1 : 1111011010 | 00001100100
s2 : 1010001010 | 11100011010
    
```

4.2 돌연변이 연산

돌연변이 연산은 유전 알고리즘에서 실질적인 진화 동작을 수행하는 교차연산과는 달리 선택된 개체에서 하나의 비트에 변화를 주어 지역최소해(Local Minim)를 탈피할 수 있도록 하는 연산자이다. 본 논문에서는 돌연변이 연산으로 반전방법(Inversion Method)을 사용한다. 반전방법은 비트 스트링에서 임의의 한 셀에 대해 0이면 1로, 1이면 0으로 값을 바꿔주는 방법이다. 다음은 돌연변이 연산 방법을 나타내고 있다[5].

```

s : 111101101000001100100
s' : 1111011010100001100100
    
```

4.3 선택 연산

(1) 목적함수

시간표 문제에서는 필수 제약조건을 반드시 만족해야 한다. 즉, 강의실 중복 회피, 교수의 시간 중복 회피, 교과목 속성과 강의실 속성의 일치를 만족해야 한다. 따라서 본 논문에서 적용되는 목적함수는 다음과 같다.

$$\text{Minimize } C = \sum_{i=1}^n \sum_{j=i+1}^n (D_{ij} + T_{ij} + R_{ij})$$

where  $i, j \in \{\text{과목1, 과목2, ...}\}$ ,

$D$ : 요일,  $T$ : 강의시간,  $R$ : 강의실

(2) 적합함수

적합함수는 다음 세대로 진화시키기 위해 부모집단으로부터 개체를 선택하기 위한 함수이다. 따라서 가능한 우성 형질을 갖는 개체를 선택할 필요가 있다. 시간표 문제에서의 우성 형질을 갖는 개체는 C의 값이 작은 개체가 해당된다. 따라서 적합함수는 다음과 같이 정의할 수 있다.

$$F(k) = \left(\frac{1}{C(k)}\right)^m$$

여기서 C(k)는 k번째 염색체의 목적함수를 나타내고, m은 스케일링(Scaling)인자로서 0보다 큰 정수이다. 이 스케일링 인자는 우성과 열성의 차이를 크게하여 우성 형질을 갖는 염색체가 선택될 수 있는 확률을 높여준다.

(3) 선택연산

선택연산은 부모 모집단으로부터 다음 세대로 진화할 개체를 선택하는 연산이다. 본 논문에서는 Roulette Wheel 선택 방법을 사용한다. 다음은 Roulette Wheel 선택 방법을 사용하기 위한 계산식이다.

$$P(k) = \frac{F(k)}{\sum_{i=1}^n F(i)}$$

여기서 P(k)는 k번째 개체의 상대적인 적합함수 값으로서 염색체의 상대적인 우성 정도를 나타낸다.

IV. 실험 결과

1. 실험 환경

본 논문에서는 윈도우7 운영체제에서 Visual Studio 2008을 이용하여 시뮬레이션을 수행하였다. 시뮬레이션에 사용된 교과과정은 3개의 전공으로 구성된 C대학의 한 스텝의 3년제 교과과정이다. 전체 학생 수는 1,350명이며, 전체 교과목 수는 분반을 포함하여 191개이고, 전체 강의실 수는 40개로 구성되어 있다. 각 강의실은 교과목의 특성 및 속성에 따라 17가지로 구분되어 있으며, 교과목의 속성에 따라 강의실이 배정되어야 한다.

유전 알고리즘에 적용된 교차연산율은 0.8이며 돌연변이 연산율은 0.05이다. 전체 모집단은 50으로 적용하였다.

2. 실험 결과

실험결과 시간표로써 의미를 가지기 위한 필수 제약조건은 모두 만족하였으나, 부가제약 조건은 일부 만족되지 않는 결과를 얻었다.

[그림 3]은 본 연구를 통해 작성된 시간표의 예를 나타내고 있다.

그룹별 강의 시간표(Group 3)

[1학년]

시간/요일	월요일	화요일	수요일	목요일	금요일
9:00~12:00	운영체제 ( 프로그래밍 PC실 #5)	인터랙티브저작도구 ( 프로그래밍 PC실 #3)	콘텐츠프로그래밍기초 ( 프로그래밍 PC실 #5)		
12:00~15:00	프로그래밍 기초(C언어) ( 프로그래밍 PC실 #1)				
15:00~18:00		인터랙티브저작도구 ( 프로그래밍 PC실 #3)			콘텐츠프로그래밍기초(Java) ( 프로그래밍 PC실 #3)
18:00~21:00					

[2학년]

시간/요일	월요일	화요일	수요일	목요일	금요일
9:00~12:00	게임 톨 프로그래밍 ( 프로그래밍 PC실 #2)	캐주얼 게임 제작 기초 ( Mac 실 #1)	캐주얼 게임 제작 기초 ( Mac 실 #1)	웹만화제작실습 ( 2D 그래픽 PC실 #3)	웹만화제작실습 ( 2D 그래픽 PC실 #4)
12:00~15:00	디지털만화원고기법 ( 2D 그래픽 PC실 #2)	앱제작기초 ( Mac 실 #1)	게임 알고리즘 ( 프로그래밍 PC실 #3)		
15:00~18:00	디지털만화원고기법 ( 2D 그래픽 PC실 #1)	앱제작기초 ( Mac 실 #1)	그래픽 프로그래밍 기초 ( 프로그래밍 PC실 #4)	네트워크 프로그래밍 기초 ( 프로그래밍 PC실 #1)	
18:00~21:00					

그림 3. 실험 결과 - 강의 시간표 예  
Fig 3. Simulation Result - An Example of Timetable

## V. 결 론

본 논문에서는 유전 알고리즘을 이용한 강의시간표 작성 시스템 설계 방법을 제안하였다.

시간표 작성 문제는 교과목, 강의실, 교수자, 강의 요일 및 시간 등 여러 가지 속성을 고려하여야 하는 NP-Complete 문제이다.

유전알고리즘은 휴리스틱 알고리즘의 하나이며 NP-Complete 문제에서 근사 최적 해를 구하는데 많이 사용되는 방법이다.

본 논문에서는 기존의 적용된 시간표 문제 해법과는 달리 교과목 시간 배정과 강의실 배정을 별도의 부 시스템으로 분리하여 유전 연산자를 적용하는 방법을 사용하였다. 적용된 교과목 수가 191개, 강의실 수가 40개인 환경에서 실험한 결과 필수 제약 조건을 모두 만족하는 결과를 얻을 수 있었다.

## 참고문헌

- [1] Werra D., Chalhal, "An interactive System for Constructing Timetables on a PC," European Journal Operation Research 40.
- [2] Hertz, A., "Tabu Search for Large Scale Timetabling Problem," European Journal of Operation Research 54.
- [3] Wilhelm E, Jurgen Keppler, "A Genetic Algorithm Solving a Weekly Course-Timetabling Problem," Proceedings of the First International Conference on the Practice and Theory of Automated Timetabling.
- [4] J. I. Ahn, S.H. Jo, "Research of a Freedom Rate for Timetabling Problem," Journal of The Korea Society of Computer Industry, Vol. 10, No. 5, 2009
- [5] Myung-Ju Kang, "Graph Transformation Decoding Method in Genetic Algorithm for Undirected Rural Postman Problem," Journal of The Korea Society of Computer and Information, Vol. 12, No. 1, 2007
- [6] Goldberg, D. E., Genetic Algorithm in Search, Optimization & Machine Learning, Addison-Wesley, 1989.
- [7] Ladd, S. R., Genetic Algorithms in C++, M&T Books, 1995.
- [8] Michalewicz Z., Genetic Algorithms + Data Structures = Evolution Programs, Second, Extended Edition, Springer-Verlag, 1994.