시간표 문제의 유전자 알고리즘을 이용한 해결에 관한 연구

안 종 일†

요 약

본 논문은 인공지능의 한 연구 분야인 다중 제약을 갖는 대학의 시간표 작성 문제를 해결하는 것으로서, 이를 위해 두 강좌 간의 시간 충돌 제약과 요일 충돌 제약을 동시에 표현 가능하도록 2-유형 에지(edge) 그래프를 정의하였다. 또한 이를 유전자 알고리즘으로 해결하는 방법을 제안하고 무작위 탐색의 효율을 높이기 위해 국부 탐색을 수행하는 방법을 소개하였다. 본 논문에서 제안된 방법의 실험결과가 무작위 탐색과 비교하여 탐색 비용을 10000번의 반복회수에서 평균 약 71% 덜 한 것으로 나타났다

A Study of Genetic Algorithm for Timetabling Problem

Jong-Il Ahn†

ABSTRACT

This paper describes a multi-constrained university timetabling problem that is a one of the field of artificial intelligent research area For this problem, we propose the 2-type edge graph that is can be represented time-conflict and day-conflict constraints simultaneously. The genetic algorithms are devised and considered for it. And we describe a method of local search in traditional random operator for its search efficiency. In computational experiments, the solutions of proposed method are average 71% costs that are compared with solutions of random method in 10,000 iterations

1. 서 론

다중 제약 조건을 갖는 작업 스케쥴링 문제의 대표적인 응용 분야는 대학의 시간표 작성 문제이다. 지금까지 시간표 문제를 모델링하기 위한 다양한 방법이 제안되었다. 주로 응용되는 방법으로는 제약 조건의 관계를 그래프로 정의하여 이에 대한 최소 착색을 목표로 하는 그래프 착색 문제(Graph Coloring Problem : GCP)가 있다[1-5, 7].

제약 조건의 표현 방법으로는 각 노드를 강좌로 표현하고 두 강좌가 서로 제약 조건으로 인해 동일한 시간에 배정이 불가능하다라는 조건이 있다면 에지(edge)로 연결하는 것이다. 착색의 목표는 에지로 연결된 두 노드가 서로 동일한 색으로 착색될 수 없다는 조건을 만족하는 최소의 색 집합을 구성하는 것이 된다.

그러나 그래프 착색 방법의 단점은 제약 조건의 표현력에 있다. 그래프 착색 문제에서 두 노드를 연결하는 에지의 의미는 두 강좌가 동일한 시간에 배정될 수 없다는 것이다[2].

시간표의 문제는 이러한 제약 조건 이외에도 여러 가지 다른 제약 조건들이 있다. 그 중 하나가 두 강좌의 특성상 동일한 요일에 배정될 수 없다는 조건이다. 동일한 시간에 배정이 될 수 없다는 제약 조건에는 동

† 준 회 원 : 용인송담대학 컴퓨터소프트웨어과 전임강사
논문접수 : 1999년 10월 11일, 심사완료 : 2000년 5월 8일

일한 요일에 배정 가능하다라는 전제 조건을 내포하고 있고 반대로 동일한 요일에 배정이 불가능하다라는 제약 조건에는 동일한 시간에 배정이 불가능하다라는 전제 조건을 내포하고 있으므로 두 제약 조건의 유형은 서로 비동질적이라고 말할 수 있다. 이러한 비동질성을 갖는 제약 조건들을 그래프 착색 문제로 정의하기 위해서는 전통적인 그래프 착색 문제에 대한 변형이 요구된다.

기존의 연구에서는 이러한 문제를 단계별로 해결하고자 하였다. 즉, 먼저 요일로 과목을 분류하는 단계를 수행한 후 동일한 요일로 배정된 과목 그룹에서 시간을 배정하는 두 번째 단계를 수행하였다[12]. 이 방법의 단점은 일단 요일의 배정이 끝나면 요일에 대한 과목 변동이 없다는 점이므로 탐색 공간을 지나치게 축소한다는 단점을 갖는다.

본 논문은 이러한 다중 제약 조건을 갖는 시간표 문제를 해결하기 위한 새로운 그래프 착색 문제의 제안한다. 이러한 문제를 해결하기 위해 본 연구에서는 2-유형 에지 그래프(2-Type Edge Graph)를 정의한다. 또한 이 그래프에 대한 착색의 방법으로 유전자 알고리즘을 이용한 착색의 방법을 소개한다.

논문의 순서는 2장에서 2-유형 에지 그래프를 정의하고 3장에서는 유전자 알고리즘을 이용하여 이 그래프를 착색하는 방법을 기술한다. 4장에서는 유전자 알고리즘의 방법을 실험하고 5장에서 결론을 맺는다.

2. 2-유형 에지 그래프 착색 문제의 정의

전통적인 그래프는 $G=(V, E)$로 구성되고, 여기서 V는 노드의 집합, E는 에지(노드 간의 제약 조건)의 집합이다. 제약 조건 문제를 그래프 착색 문제로 정의할 수 있다면 그로부터 다양한 정보를 얻을 수 있다. 우선, 문제를 구조적으로 표현하게 되어 문제가 갖는 복잡도를 알아낼 수 있다는 점과 이에 따라 풀이 방법에 경험적 지식을 적용할 수 있도록 해준다[10].

대학 시간표의 경우 두 강좌가 동일한 시간에 배정될 수 없다는 것이 가장 대표적인 제약 조건 중의 하나이며 이와 더불어 두 강좌가 같은 요일에 배정될 수 없다는 조건이 자주 등장한다[9]. 이러한 두 가지 유형의 제약 조건을 전통적인 그래프로 표현하는 데에는 한계가 있다. 따라서 본 논문에서는 이 두 가지 제약 조건을 모두 표현할 수 있는 2-유형 에지 그래프로 문제를 정의한다.

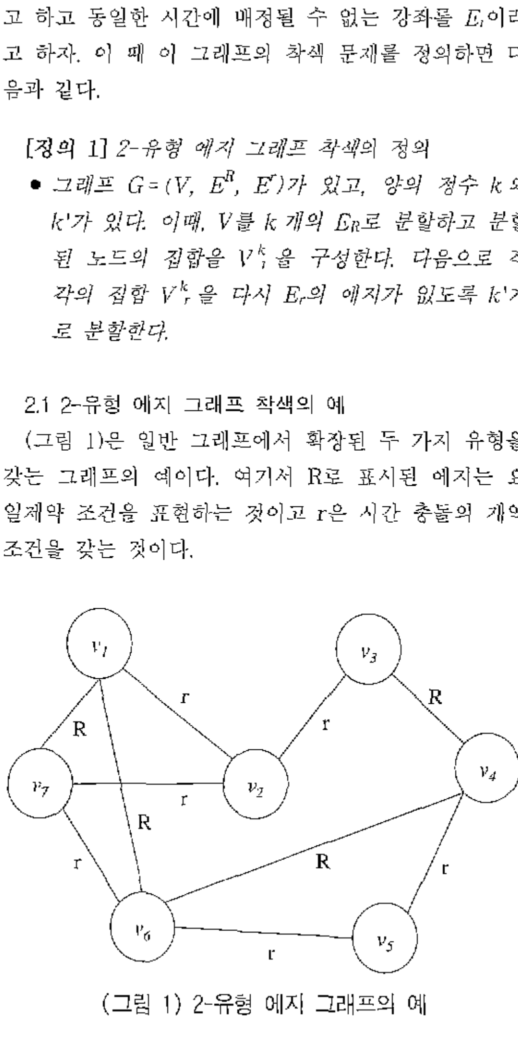
그래프 $G=(V, E^R, E^r)$로 표현하고 이때 그래프로 표현하고자 하는 강좌의 집합을 V라고 하고 동일한 요일에 배정될 수 없는 강좌간의 연결 에지를 E_R이라고 하고 동일한 시간에 배정될 수 없는 강좌를 E_r이라고 하자. 이 때 이 그래프의 착색 문제를 정의하면 다음과 같다.

[정의 1] *2-유형 에지 그래프 착색의 정의*

- *그래프 $G=(V, E^R, E^r)$가 있고, 양의 정수 k와 k'가 있다. 이때, V를 k개의 E_R로 분할하고 분할된 노드의 집합을 V^k_r을 구성한다. 다음으로 각각의 집합 V^k_r을 다시 E_r의 에지가 없도록 k'개로 분할한다.*

2.1 2-유형 에지 그래프 착색의 예

(그림 1)은 일반 그래프에서 확장된 두 가지 유형을 갖는 그래프의 예이다. 여기서 R로 표시된 에지는 요일제약 조건을 표현하는 것이고 r은 시간 충돌의 제약 조건을 갖는 것이다.



(그림 1) 2-유형 에지 그래프의 예

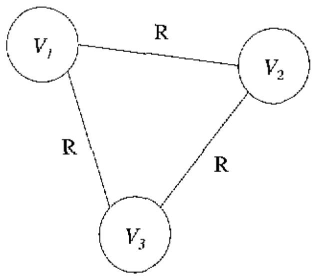
(그림 1)의 착색의 다양한 결과를 직관적으로 상상해 보자. 동일 색으로 착색된 노드의 집합이 의미하는 것은 두 가지 유형의 제약 조건을 모두 만족하는 집합을 의미한다. 색 $_1$로 착색된 노드의 집합을 V_1라고 하자. 예제 1에서 해결한 그래프 착색의 해 S_1과 S_2는 (그림 2)와 같은 형태로 그래프의 동일한 색을 배정받은 노드가 병합된다.

(그림 2)의 결과는 착색의 방식에 따라 필요 자원의 배정 형태가 달라질 수 있음을 보이고 있다. S_1의 경

우에 필요한 자원의 양은 3개의 R 자원과 각 R 자원마다 1개의 r 자원이 필요하다. 그러나 S_2의 경우는 이와는 다르게 R 자원의 입장에서는 $\{V_1\}$, $\{V_2, V_3\}$로 분리될 수 있다. 이는 2개의 R과 각 1개와 2개의 r이 필요하게 된다는 의미이다.

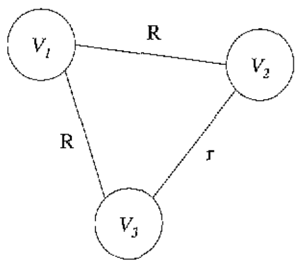
(a) S_1

$V_1 = \{ v_2, v_6 \}$
$V_2 = \{ v_1, v_3, v_5 \}$
$V_3 = \{ v_7, v_4 \}$



(b) S_2

$V_1 = \{ v_1, v_4 \}$
$V_2 = \{ v_2, v_6 \}$
$V_3 = \{ v_7, v_5, v_3 \}$



(그림 2) 2-유형 에지 그래프의 착색의 예

2차원적 구조에 이를 표시하면 <표 1>과 같다.

〈표 1〉 (그림 1)의 그래프의 착색의 결과

	r \ R	1	2	3		r \ R	1	2	3
S_1	1	V_1	V_2	V_3	S_2	1	V_1	V_2	
	2					2		V_3	

결과를 $R : r$ 착색되었다고 하자 위의 <표 1>에서 S_1은 3 : 1 착색되었고, S_2는 2 : 2 착색되었다. 따라서 두 가지 유형의 제약 조건을 고려하지 않은 착색의 결과는 다양할 수 있다. 위에서 결론적으로 그래프 착색이 최소 착색의 수를 찾는 문제라고 한다면, *2-유형 에지 그래프 착색*은 최소 착색의 수를 찾는 문제와 더불어 착색이 제약 조건에 따라 어떻게 2차원적 평면상에 배치 될 수 있는지에 대한 문제를 포함한 것이라 할 수 있다

3. 유전자 알고리즘을 이용한 문제의 해결

이번 절에서는, 2-유형 에지 그래프의 R 유형(요일)이 k-착색인 문제를 유전자 알고리즘을 이용하여 해결하고자 한다. 시간표 문제의 예를 들면 주 k=5일로 요일에 대한 길이 제약이 있고 매일의 시간을 최소로 작성하는 문제가 된다.

3.1 염색체 정의

염색체는 그래프의 노드가 임의의 정렬 상태로 놓여 있을 때, 각 노드가 어떤 색을 갖게 될 것인지를 정의한다. *2-유형 에지 그래프 착색*의 경우에 염색체에 부여된 색은 요일을 의미한다. 염색체는 그래프 G가 있을 때, 그래프의 노드의 집합 $V=\{v_1, v_2, \cdots, v_n\}$가 주어지면 각 노드에 하나의 요일이 할당될 수 있도록 한다.

3.2 평가 함수의 정의

*2-유형 에지 그래프*의 착색은 요일을 어떻게 결정하느냐에 따라 시간 배정의 결과가 좌우 된다는 특징을 갖는다. *2-유형 에지 그래프*의 k최소 착색의 목표는 요일을 k개로 착색을 완료하는 것과 더불어 최소의 동일한 요일을 매칭 받은 노드가 최소의 시간을 갖게 될 것인가를 찾아야 한다는 것이다.

$$Eval(G) = \sum_{i=0}^{i \le k} (f_R(V_i) + f_i(V_i)) \qquad (1)$$

여기서, $f_R(V_i)$은 요일이 배정된 결과를 말한다. 여기서 V_i는 동일한 요일에 배정된 과목의 집합을 의미하고 평가 함수 f_R은 요일의 제약 조건을 만족하지 못한 노드의 개수이다.

$f_r(V_i)$는 요일 i로 배정 받은 과목의 시간 배정 결과를 말한다. 함수 $f_r(V_i)$는 각각의 집합 V_i의 착색의 최소수를 결정하는 문제이다. 임의의 그래프가 최소한 몇 개의 색의 수를 필요로 하는지에 대한 정리는 Brooks에 의해 증명되었다[29].

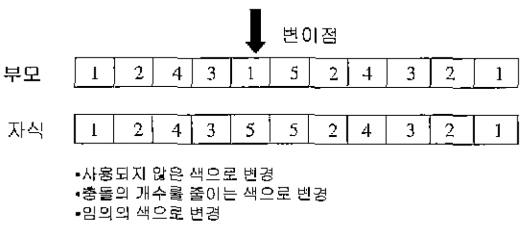
[Brooks의 정리] *그래프 G는 $\chi(G) \leq \Delta G+1$이다.*

여기서 $\chi(G)$는 그래프 G에 대한 최소 색의 수이며, ΔG는 그래프 G의 최대 에지 연결도(maximum degree)를 말한다. 따라서 함수 f_r은 집합 V_1에서 r 유형의 에지의 연결도 값이 된다.

3.3 운영자의 정의

전통적인 돌연변이 연산자는 임의로 유전자 군으로부터 부모를 선택하고 부모의 유전 형질을 무작위로 바꾸는 방법을 사용한다. *2-유형 에지 그래프*의 착색 문제에서는 임의로 선택한 유전자의 형질 변경에서 탐색 속도의 개선을 위해 국부적인 탐색을 수행하는 방법을 제안한다. 임의로 선택된 노드를 어떤 색으로 변경할 것인지를 판단하기 위해 k 착색의 과정에서 사용하지 않은 색이 있는지를 검색한다. 만약, 이러한 색이 존재하지 않는다면 현재 노드가 배정된 색으로부터 다른 색으로 이동할 경우 충돌을 줄일 수 있는 색으로 착색한다. 모든 방법에서 실패하면 임의의 값으로 노드의 색을 변경한다. 시간표의 경우 이것의 의미는 하나의 강좌에 요일을 결정하는데 있어서 우선 주어진 k 개의 요일에서 강좌가 배정되지 않은 요일이 있는지를 살펴보고 만약 존재한다면 우선하여 해당 요일에 배정하고 만약 이에 만족하지 못한다면 모든 요일에서 이미 배정되어 있는 강좌들과 요일 충돌을 최소로 야기하는 곳을 탐색하여 강좌를 배정을 한다는 것이다.

(그림 3)은 제안된 돌연 변이 연산자이다.



(그림 3) 제안된 돌연 변이 연산자

염색체를 *chromosome*[*i*][*j*]은 2차원 배열의 형태를 갖는다. 여기서 i는 전체 유전자 군의 각각의 염색체 색인 값이고, j는 염색체 내부 형질 인자에 대한 색인 값이다 또한 각 노드의 R 유형 착색 결과를 V_k이라고 하고 $|V_k|$는 이 원소의 개수이다. 각 착색의 결과로서, R 유형의 착색의 오류를 *nb_C*라고 하자. 다음은 (그림 3)에서 제안된 돌연 변이 연산자의 전체 프로시저이다.

```
Procedure Mutate;

Begin
   p := Randomly Select from Population
   h = Randomly Select a Mutate point
   k ' = length of Color
   For i = 1 to k do
         If ( |V_i| = 0 )
               chromosome [p][h] := i
               Return chromosome [p]
   End of i loop
   nb_C_old = Get number of R-Type Constraint Conflict
   For i := 1 to k do
         Temp = chromosome [p][h]
         chromosome [p][h] := i
         nb_C := Get number of R-Type Constraint Conflict
         If (nb_C < nb_C_old)
            Return chromosome [p]
         Otherwise
               chromosome [p][h] := Temp
   End of i loop
   i := Randomly Select from 1 to k
   chromosome [p][h] := i
   Return chromosome[p]
End
```

(그림 4) 변이 연산자 프로시저

4. 실 험

실험은 유전자 알고리즘의 변이 연산자의 무작위 방법과 국부적 탐색을 이용한 방법의 성능을 측정하여 본다. 실험을 위해 그래프의 생성에서 세 가지 인수(parameter)를 사용한다. 첫째, 노드의 개수 N, 둘째, R 유형의 에지의 연결 확률 p_1, r 유형 에지의 연결 확률 p_2, 에지의 개수는 평균 노드의 연결 개수를 사용한다.

유전자 알고리즘을 위한 환경 변수와 실험에 사용될 그래프의 생성을 정의한다.

4.1 유전자 알고리즘을 위한 환경 변수 정의

- 유전자 군의 크기 : 50
- 교배 연산자의 적용 확률 : 80%
- 돌연 변이 연산자의 적용 확률 : 55%
- 선택 방법 : 룰렛 휠 방법
- 최대 반복 회수 : 10000회

실험에 사용될 그래프는 세가지 인수의 변화에 따라 생성되고 이에 따라 각 실험의 성능을 측정하고자 한다.

4.2 그래프의 생성 방법

- 노드의 수 : N
- 에지 연결의 비율 : $0 < p_1 < 1.0$, $0 < p_2 < 1.0$

그래프의 생성에서 연결 비율 p_1과 p_2은 완전 그래프의 연결 확률을 1로 보았을 때, 이에 대한 각각의 비율 값을 의미한다. 즉, 노드의 개수가 $N = 10$개일 경우 $p_1 = 1$이면 총 $(N(N-1))/2 = 45$개의 링크가 생성된다 $p_1 = 0.5$일 경우에는 총 22개의 링크를 갖는 그래프가 생성된다. 즉 평균 연결 링크의 수는 다섯 개 정도가 된다.

그래프의 에지의 개수를 $|E|$라고 하고, 이미 두 노드가 연결되었는지를 검사하는 함수를 *IsAdjacent* (v_i, v_j)이고 이 함수는 참 혹은 거짓을 돌려 준다. 그리고 두 노드를 에지로 연결하는 함수를 *Connect* (v_i, v_j)라고 하자.

그래프 생성의 방법을 프로시저로 정의하면 (그림 5)와 같다.

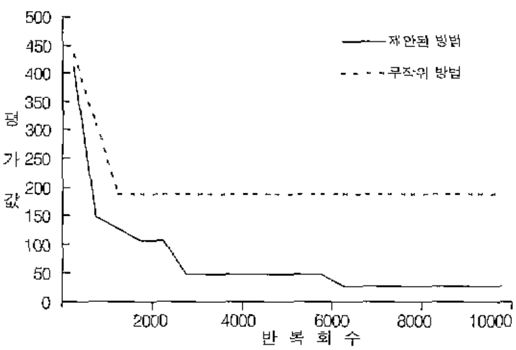
```
Procedure MakeRandomGraph;

Begin
   |E| := (N(N-1))/2 p1
   Until Count < |E| do
         vi := Randomly Select from V
         vj := Randomly Select from V
         if (vi ≠ vj AND IsAdjacent (vi, vj) = FALSE )
         Then Connect (vi, vj) Count ++
   End fo Until
End
```

(그림 5) 무작위 그래프 생성 프로시저

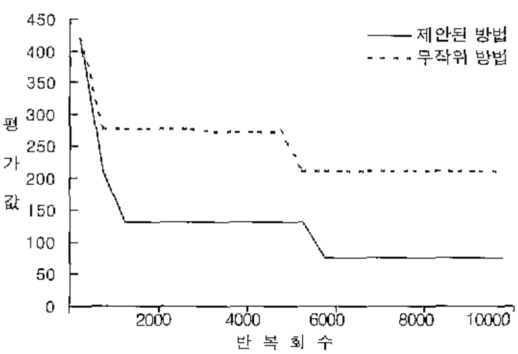
(그림 4)에서 제안한 돌연 변이 연산자를 무작위에 의한 방법과 국부적인 탐색을 사용한 방법을 비교한다. 유전자의 변수 환경은 위에서 정의한 방법을 사용하고 그래프는 노드를 50개로 설정하여 측정하였다. 각 실험에서 사용될 문제는 그래프의 연결도의 변화에 따른 결과의 변화를 살펴보았다. 각각의 그래프는 $p_1 = 0.1$, $p_2 = 0.1$과 $p_1 = 0.5$, $p_2 = 0.5$로 생성하였다

5. 결 론

지금까지 다중 제약 조건을 갖는 작업 스케줄링 문제의 해결 방법을 연구하였다. 제약 조건 만족의 문제에 대한 효과적인 모델링은 탐색의 효율을 높일 수 있는 핵심적인 정보를 제공해 준다. 기존의 최적의 시간표를 작성하는 문제를 모델링하기 위한 다양한 방법들이 제안되었다. 이 중, 그래프 착색으로의 문제 표현의 장점은 문제를 구조적으로 표현하여 문제의 복잡도와 탐색 될 작업의 순서, 제약 조건간의 탐색 과정에서 중요도의 결집 등 다양한 정보를 제공한다. 이러한 다양한 장점에도 불구하고 전통적인 그래프 착색의 문제는 제약 조건을 하나의 유형의 에지로 표현하므로 표현력이 약하다라는 단점을 갖는다. 따라서 본 연구에서는 이러한 전통적인 그래프 착색의 문제의 한계를 극복하기 위한 방법으로 에지의 유형이 두 가지인 2-유형 에지 그래프의 착색 문제를 정의하였다.



(그림 6) 돌연변이 연산자 수행 결과 그래프
($N = 50$, $p_1 = 0.1$, $p_2 = 0.1$)



(그림 7) 돌연변이 연산자 수행 결과 그래프
($N = 50$, $p_1 = 0.5$, $p_2 = 0.5$)

다음으로 유전자 알고리즘을 이용한 제안된 2-유형 에지 그래프의 착색의 방법을 모색하였다.

염색체는 R유형의 착색 결과를 스트링으로 표현하

였다. 염색체의 평가의 방법으로 동일한 요일로 배정된 노드의 시간 충돌 제약 조건의 에지 연결 관계를 분석하였다. 제안된 문제의 특성은 R유형의 착색의 결과에 따라 r유형의 착색에 영향을 미친다는 점이므로 본 연구에서는 두 가지 제약 조건을 동시에 만족하도록 평가함수를 설계하였다. 또한 탐색을 효율적으로 수행하기 위해 돌연변이 연산자에 국부적인 탐색 방법을 결합하여 탐색 효율을 높였다. 실험의 결과 무작위 탐색에 비해 국부적인 탐색을 수행한 결과 비용이 10000회의 반복 회수의 경우 평균 0.29 낮아졌다.

참 고 문 헌

[1] Ahn J.I., and Chung T.C., "Graph coloring algorithm to Make Timetable for Lessons Requiring Multiple Slots," *Proceedings of the 2nd international Conference on the Practice and Theory of Automated Timetabling*, pp.281-284, 1997.

[2] Burke E.K., Elliman D.G., "A University Timetabling System based on Graph coloring and Constraint Manipulation," *Journal of research on computing in Education*, Vol.27, No.1, pp.1-18, 1993.

[3] Cangalovic M. and Schreuder J.A.M., "Exact coloring algorithm for weighted graphs applied to timetabling problems with lectures of different lengths," *European Journal of Operational Research* 51, pp.248-258, 1991.

[4] Clementson A.T. and Elphick C.H., "App roximate Colouring Algorithms for Composite Graphs," *Operational Research* Vol 34, No.6, pp.503-509, 1983.

[5] Ferland J., "Generalized Assignment-Type Problems : A Powerful Modeling Scheme," *Proceedings of the 2nd international Conference on the Practice and Theory of Automated Timetabling*, pp.27-54, 1997.

[6] Jensen T.R and Toft B., "Graph Coloring Problems," Wiley-Interscience publication, pp.7-8, 1995.

[7] Krarup J. and D. de Werra, "Chromatic Optimization : Limitation, Objectives, Uses, References," *European Journal of Operational research* 11, (1982), pp. 1-19.

[8] Lawton G., "Genetic Algorithms for Scheduling Optimization," *AI Expert*, pp.23-29, 1992.

[9] Schaerf A., "A Survey of Automated Timetabling," *Center voor Wiskunde en Infomatica Report* CS-R9567, 1995

[10] Werra D. de, "An Introduction to timetabling," *European Journal of Operational Research* 19, pp.151-162, 1985

[11] Winston P.H., "Genetic Algorithms Artificial Intelligence 3rd," Addison-Wesley publishing Company, pp.505-528, 1992

[12] Taghi Ariani, "A Three Phased App.roach To Final Exam Scheduling," Vol.21, No.1, IIE Transaction, pp. 86-96, March 1989.

안 종 일

e-mail : ahn@nslab.kyunghee.ac.kr

1992년 국민대학교 기계설계학과 졸업(학사)

1994년 경희대학교 전자계산공학과 (공학석사)

1998년 경희대학교 전자계산공학과 (공학박사)

1999년 ~ 2000년 해전대학 컴퓨터서비스과 전임강사

2000년 ~ 현재 용인송담대학 컴퓨터소프트웨어과 전임강사

관심분야 : 인공지능, 최적화, 스케쥴링, 데이터마이닝