

중복시험을 배제할 수 있는 사이버대학의 온라인 실시간 시험 시간표 작성 방안

박 찬 권* · 김 형 도** · 엄 지 환**

A Heuristic Approach for Establishing On-line Real-time Exam Timetables in Cyber Universities

Chankwon Park* · Hyoung Do Kim** · Jihwan Yum**

Abstract

Students in cyber universities can take every course because there are no physical constraints such as class rooms. On the other hand, cyber universities should take a heavy burden to schedule real-time exams in a designated time period. Any two courses a student takes must not be simultaneously allocated to a time slot in such a schedule. It is well known that the complexity of this kind of scheduling problem increases exponentially as the number of courses does. This research provides a heuristic method to make on-line real-time exam timetables with minimal exam time span. It considers the constraint of not allocating courses a student takes to simultaneous exam time and prioritizes courses by the number of students.

Keywords : On-line Real Time Exam Timetable, NP Hard, Heuristic Algorithm

1. 서 론

2001년부터 기존의 오프라인 대학과 달리 인터넷을 이용해서 강의를 수강하는 형태의 사이버 원격교육 체제가 도입된 이래, 현재 국내에는 17개의 사이버대학이 설립되어 활발하게 교육이 이루어지고 있다. 온라인 원격 강의의 유용성에 대한 인식 확산과 함께 평생교육에 대한 관심이 고조되면서 학생 수는 꾸준히 증가하고 있으며, 이와 함께 시험 평가를 포함한 학사운영의 강화에 대한 요구도 함께 증가하고 있다[4].

온라인 수업을 기반으로 한 사이버대학에 있어서 시험 및 평가 체제는 오프라인 수업과 비교했을 때 여러 가지 측면에서 제약을 받을 수밖에 없는 것이 현실이다. 이러한 제한된 평가 환경에 있어 그 운영상의 한계에도 불구하고 비교적 객관적이고 공정한 평가 방법이라는 점에서 온라인 실시간 시험은 현재로서는 가장 큰 비중을 차지하고 있다.

하지만, 사이버대학에서는 일반 오프라인 대학과 달리 물리적으로 강의실을 배정하는 수업 시간표가 따로 정해지지 않기 때문에 수업시간표 제약 없이 모든 과목에 대해 무작위로 수강하는 것이 가능한데, 이와 같은 환경에서 복수 과목을 수강하는 학생들의 수강 현황을 고려해서 시험시간이 중복되지 않도록 시험시간표를 편성하는 일은 시험 운영에 있어서 대단히 중요한 문제가 되고 있다.

학교의 규모가 작은 대학의 초기 학사 운영 체제에서는 실시간 시험 대상 과목 수가 한정되기 때문에, 이러한 실시간 시험 운영을 위한 시험계획의 작성은 단순 수작업에 의존해서 해결이 어느 정도는 가능하다고 할 수 있다. 그러나 사이버대학의 경우 기본적인 학교 규모가 있기 때문에 학생 수와 개설 강좌 수가 증가함에 따라 실시간 시험 대상 과목도 비약적으로 증가함

으로써 시험 운영 시간 계획을 작성하는 작업 자체의 복잡성뿐만 아니라, 수작업에 따른 오류 발생 가능성도 증가하게 된다. 따라서, 시험기간이라는 제한된 시험 일정을 탄력적으로 운영할 수 있는 체계적 시험 운영 계획 수립 방안이 요구된다.

본 연구에서는 과목별 수강생의 동시 수강 제약조건과 과목별 수강생 규모를 고려해서 동시에 온라인 시험을 치를 수 있는 시험 운영계획을 수립할 수 있는 현실적인 방안을 제시하고자 한다. 이 논문의 구성은 다음과 같다. 제 2장에서는 관련된 연구를 정리하고, 제 3장에서는 그래프를 이용해서 문제를 모델링하는 방법을 설명한다. 그리고 제 4장에서는 발견적 기법을 제시하고, 제 5장에서는 실제 문제에 적용한 결과를 가지고 분석을 수행한다. 마지막으로 제 6장에서는 결론과 향후 연구 방향을 제시한다.

2. 관련 연구

일반적으로 대학에서 강의시간표를 작성하는 문제는 개설되는 모든 교과목에 대해 강의실과 강의시간대(타임 슬롯)라는 제한된 자원을 배정하는 일정계획 문제로 볼 수 있다. 하지만, 실제 강의시간표를 작성하는 과정에서는 교과목에 대해 단순히 강의실과 강의시간을 배정하는 데 그치는 것이 아니라 강의담당 교수와 학생의 수강 내용 등의 추가적인 제약을 고려해야 하기 때문에 훨씬 복잡한 문제가 되고 있다[1, 2, 3, 5, 6, 7]. 이들 제약조건들은 시간표를 작성하기 위해서 반드시 지켜져야 하는 필수적 제약조건과 교수 개인의 선호도 등과 같은 부수적인 제약조건들로 구분될 수 있는데, 필수적인 제약조건에는 다음과 같은 사항들이 포함된다.

- 한 교수는 같은 요일, 같은 시간(동일한 타임

슬릿)에 두 개 이상의 강의에 배정될 수 없다.

- 한 학년의 한 분반은 동일한 타임 슬릿에 두 강의를 받을 수 없다.
- 한 학생은 동일한 타임 슬릿에 수업이 중복 되면 안 된다.
- 어떠한 과목도 겹쳐서 강의될 수 없다.

대학의 강의시간표는 거의 모든 대학에서 매 학기 겪게 되는 어려운 문제로 자동화된 방법을 통해 시간표를 작성하고자 하는 많은 노력들이 진행되어 왔다[5]. 이 같은 시도들은 크게 수리적인 방법이나 발견적(heuristic) 기법들로 구분될 수 있는데, 필수적 제약조건 하의 강의시간표 작성 문제 자체가 NP-Hard 문제로 모델링되기 때문에 대부분 발견적 기법을 통해 접근하고 있다[1, 2, 3, 5, 6, 7]. 특히, 그래프를 이용해서 시간표 문제를 접근하는 경우 이들 문제가 그래프 착색 문제로 모형화될 수 있음을 보이고 있다[2, 3].

한편, 대학의 시험시간표 작성 문제도 강의시간표 작성 문제와 비슷하게 생각할 수 있는데, 각 시험과목에 대해 감독자와 시험장소가 미리 결정되는 것이 일반적이므로 문제의 복잡도는 훨씬 줄어든다고 할 수 있다[1]. 더욱이 전공 시험의 경우는 학기 초 결정된 강의시간표를 이용해서 치러질 경우라면 별도의 시간표 작성 문제 자체가 무의미할 수 있다.

반면에 사이버대학의 강의는 일반 대학의 강의와 달리 인터넷을 이용해서 원하는 교과목을 원하는 시간에 수강할 수 있는 비대면, 비동기식 강의를 특징으로 하고 있기 때문에 앞서 제시한 필수적 제약조건의 제한을 받지 않는다. 즉, 일반대학과 같은 강의시간표 작성 문제가 발생하지 않으며, 학생입장에서는 교과과정이 허용하는 한 대학 내에 개설된 모든 과목을 수강할 수 있게 된다. 대신 시험시간표에 따라 실

시간으로 치러야 하는 온라인 시험은 “한 학생은 동일한 타임 슬릿에 시험이 중복되면 안 된다”는 필수적 제약을 가지는 시간표를 작성하는 문제가 발생한다. 하지만 이 경우도 일반 대학 강의시간표와는 달리 이미 수강신청이 이루어진 이후에 중복 시험 배정을 막아야 한다는 점에서 차이가 있다. 또한 많은 제약조건외의 완화에도 불구하고 문제의 복잡성은 여전히 줄어들지 않는다고 하겠다.

강의시간표 작성에 대한 그동안의 많은 연구에도 불구하고, 대부분 연구자가 소속된 대학의 문제를 중심으로 연구가 진행되어 온 만큼 원격 대학의 강의 특성을 반영한 시험시간표 작성에 대한 연구는 아직 찾아보기 어려운 실정이다.

3. 모델링

3.1 문제 개요

사이버대학의 경우 물리적인 강의 공간이 필요치 않기 때문에 학생들은 별도의 제약 없이 교과과정이 허용하는 한 모든 교과목을 수강할 수 있다. 대신 실시간 시험시간표를 작성하는 과정에서, 학생의 중복수강 문제를 고려하지 않는다면 동일한 시간대에 시험을 동시에 치러야 하는 문제가 발생할 수 있다. 이 같은 제약조건 하에서 n 개의 과목이 개설되는 경우 극단적으로는 n 교시의 시험시간표 편성이 요구될 수도 있지만, 교양과 전공, 학년별 구분, 학과별 구분 등 중복 수강을 줄이는 요인들이 있기 때문에 현실적으로는 시험기간 내에 실시간 시험을 치를 수 있도록 시간표를 편성할 수 있게 된다. 하지만, 학생 수가 증가하고, 개설 강좌 수가 확대될수록 정해진 기간 내에 시험을 마칠 수 있도록 시험시간표를 편성하는 문제의 복잡도는 비약적으로 증가하게 된다.

3.2 중복수강과 Adjacency Matrix

학생들의 중복수강을 고려해서 실시간 시험 시간표를 작성하기 위해서는 <그림 1>과 같이 전체 교과목에 대해서 각 과목별로 중복수강 여부를 파악하는 것이 필요하다. <그림 1>은 예로 보여주기 위해 그 중 일부만을 표현한 것으로, 이렇게 파악된 중복수강인원은 시험시간표 편성 문제의 기본입력 값이 된다.

강좌명	수강 인원	1	2	3	4	5	6	7	8	9	10	
자바프로그램	273	1	273	3	13	1	0	0	0	1	3	1
e-비즈니스시스템	89	2	3	89	1	0	0	0	0	0	0	1
웹프로그래밍	241	3	13	1	241	13	4	8	3	1	1	1
평생교육개론	105	4	1	0	13	105	6	16	13	0	0	0
웹기반시스템	33	5	0	0	4	6	33	5	3	1	0	0
교육용컨텐츠개발	103	6	0	0	8	16	5	103	17	0	0	0
이러닝세미나	68	7	0	0	3	13	3	17	68	0	0	0
광고매체론	110	8	1	0	1	0	1	0	0	110	0	13
브랜드관리론	131	9	3	0	1	0	0	0	0	8	131	101
옥외광고론	165	10	1	1	1	0	0	0	0	13	101	165

<그림 1> 교과목별 중복 수강인원의 파악 예

두 과목 사이에 단 한 사람의 중복수강 인원이 있더라도 해당 과목들은 동일한 타임슬롯에 시험을 편성하면 안되기 때문에, 시험시간표 작성에는 중복수강 인원보다는 중복수강 여부가 기준이 된다. 따라서 식 (1)과 같이 중복수강 여부를 정의해서 이를 <그림 2>와 같은 Adjacency Matrix로 표현할 수 있다.

$$a_{ij} = \begin{cases} 1, & \text{중복 수강이 없는 경우} \\ 0, & \text{중복 수강이 있는 경우} \end{cases} \quad (1)$$

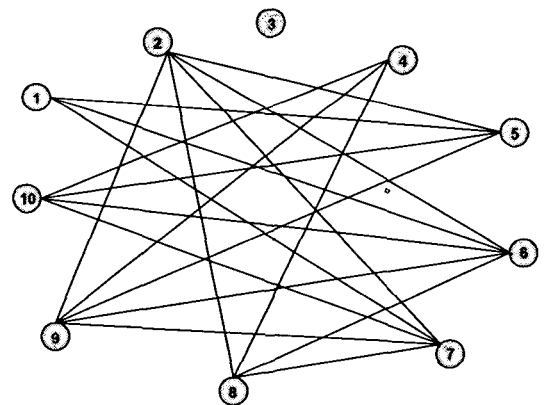
3.3 Adjacency Matrix의 그래프 표현

일반적으로 Adjacency Matrix는 아크와 노드로 이루어진 그래프를 표현하기 위한 것으로, <그림 2>와 같이 정의된 Adjacency Matrix는

그래프를 이용해서 나타낼 수 있다. 즉, 각 교과목을 노드로 정의하고, $a_{ij}=1$ 인 경우에 대해서 과목 i와 과목 j 사이에 아크를 정의하면 <그림 3>과 같은 무방향 그래프로 표현될 수 있다[9].

강좌명	수강 인원	1	2	3	4	5	6	7	8	9	10
자바프로그램	273	1	0	0	0	0	1	1	1	0	0
e-비즈니스시스템	89	2	0	0	0	1	1	1	1	1	0
웹프로그래밍	241	3	0	0	0	0	0	0	0	0	0
평생교육개론	105	4	0	1	0	0	0	0	0	1	1
웹기반시스템	33	5	1	1	0	0	0	0	0	0	1
교육용컨텐츠개발	103	6	1	1	0	0	0	0	0	1	1
이러닝세미나	68	7	1	1	0	0	0	0	0	1	1
광고매체론	110	8	0	0	0	1	0	1	1	0	1
브랜드관리론	131	9	0	0	0	1	1	1	1	0	0
옥외광고론	165	10	0	0	0	1	1	1	1	0	0

<그림 2> 예제에 대한 Adjacency Matrix



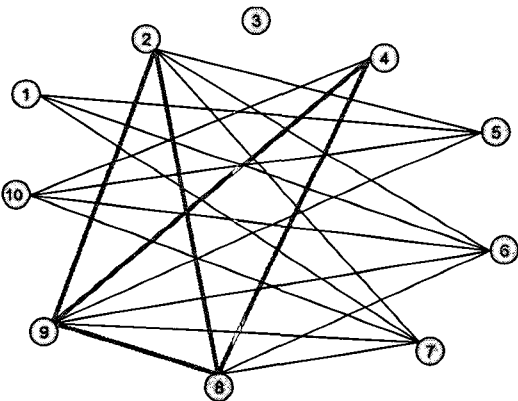
<그림 3> 예제에 대한 그래프 표현

3.4 동일 타임슬롯에 배정 가능한 과목 집합

노드 집합 $V = \{v_1, v_2, \dots, v_n\}$ 에 대해 $G = (V, E)$ 가 무방향 그래프라고 할 때, V 의 부분 집합으로서 부분집합 내 각 노드가 부분집합 안의 다른 모든 노드와 연결된 그래프(즉, Complete Graph)는 Clique으로 정의된다. 이 때 부분집합을 구성하는 노드의 수를 Clique Size라고 하며, 주어진 그래프에서 찾을 수 있는 가장

큰 크기의 Clique을 Max Clique이라고 한다[8].

주어진 Adjacency Matrix를 이용해서 표현된 그래프에서 Clique을 구성하는 노드 집합은 동일 시간대에 시험 편성이 가능한 과목들의 집합을 의미하게 된다. 예컨대 <그림 4>에서 보는 것처럼 노드(2, 4, 8, 9)는 하나의 Clique을 구성하고 있고, 이들 과목은 모두 같은 시간대에 시험을 치를 수 있는 과목 집합 즉, 동일한 타임슬롯에 배정될 수 있는 과목들을 의미한다. 반면에 노드(3)은 함께 치를 수 있는 과목이 없고, 단독으로만 시험이 가능한 과목을 나타낸다.



<그림 4> Clique과 동일 시간대 시험편성 가능 과목

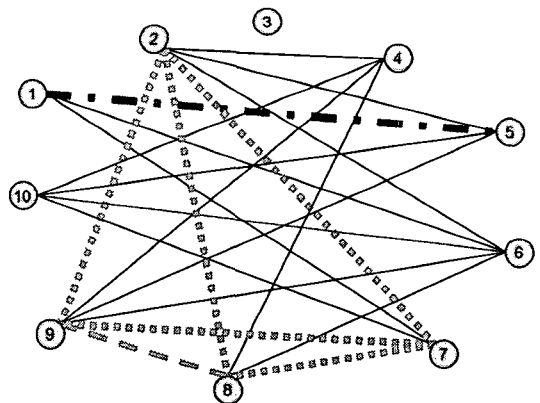
하지만, 주어진 그래프로부터 Max Clique을 찾는 문제는 NP-Hard 문제로 알려져 있다[8]. 이는, 문제의 크기가 커질수록 그 해를 찾는 데 소요되는 시간이 급격하게 증가함으로써 실제 문제에서 최적해를 찾아내기가 쉽지 않다는 것을 의미한다.

3.5 전체 타임슬롯의 최소화를 위한 시간 배정

노드 집합 $V = \{v_1, v_2, \dots, v_n\}$ 에 대해 $G = (V, E)$ 가 무방향 그래프라고 할 때, V 를 서로 중복되지 않는 V_1, V_2, \dots, V_k ($1 \leq i \leq k$ 인 i 에 대해 V_i 는 Clique 즉, Complete Graph)로 분할하는

문제는 Clique Partition 문제로 정의된다[9]. 이때 V_i 의 수 즉, k 를 최소화하는 문제를 Minimum Clique Partition(MCP) 문제라고 한다. 따라서 주어진 시험 기간 내에 시험을 모두 마칠 수 있도록 최단 기간 내에 모든 과목의 시험 시간을 배정하는 문제는 MCP 문제로 간주할 수 있다.

예컨대, <그림 5>와 같이 앞서 작성한 예제 그래프를 $\{1, 5\}$, $\{6, 10\}$, $\{2, 7, 8, 9\}$, $\{3\}$, $\{4\}$ 의 5개 부분집합으로 분할하게 되면, 이들 각각은 Clique을 구성하게 되므로 5교시로 시험 시간을 편성할 수 있게 된다. 하지만, MCP 문제도 Max Clique을 찾는 문제와 마찬가지로 NP-Hard 문제로 알려져 있다[2].



<그림 5> Clique Partition 문제

4. 발견적 기법(Heuristic Approach)

모든 과목에 대한 실시간 시험 시간을 배정하기 위한 발견적 기법(Heuristic Approach)은 전체 과목을 대상으로 동시에 시험을 치를 수 있는 과목 집합을 편성(타임슬롯의 배정)하고, 남은 과목 집합을 대상으로 다시 동시에 시험을 치를 수 있는 과목집합을 차례로 반복해서 편성(타임슬롯의 배정)해가는 방법을 사용한다.

4.1 타임슬롯의 배정

각 교과목(노드)과 동시에 시험이 가능한 과목은 그래프에서 해당 노드로부터 아크가 연결된 노드들이다. 해당 노드에 연결된 아크가 많을수록 동시 시험편성이 가능한 (즉, 동일한 타임슬롯에 배정될 수 있는) 과목 수가 많아질 수 있으며, 반대로 아크의 수가 적을수록 동일한 타임슬롯에 배정될 수 있는 과목의 수(즉, Clique Size)는 적어진다고 할 수 있다. 타임슬롯의 배정은 아크의 수가 많은 과목부터 배정하는 방법과 아크의 수가 적은 과목부터 배정하는 방법이 있다.

연결된 아크의 수가 많은 과목부터 타임슬롯을 배정해가는 경우, 차시 배정이 반복될수록 타임슬롯이 배정된 과목을 제외한 나머지 과목들 사이의 전체 아크 수는 급격히 줄어들면서 남은 과목들은 단독으로 차시가 배정될 가능성이 많아진다. 더욱이 차시 배정 초기에는 동시 편성 가능한 과목 수가 많기 때문에 Max Clique을 찾는 데 더 많은 시간이 소요될 수 있다. 반면에, 연결된 아크의 수가 적은 과목부터 차시 배정을 해가는 경우는 최소한으로 아크 수를 줄여갈 수 있기 때문에 배정되고 남은 과목들 사이의 동시 편성 가능한 과목 수(즉, Clique Size)를 크게 하면서 차시 배정을 진행할 수 있다.

노드 집합 $V = \{v \mid v = 1, 2, \dots, n\}$ 와 아크 집합 $E = \{e_{ij} \mid e_{ij} = 1 \text{ where } a_{ij} = 1 \text{ and } 1 \leq i \leq n, 1 \leq j \leq n\}$ 에 대해 $G = (V, E)$ 라고 할 때, 노드 i 에서의 아크의 수 N_i 는 식 (2)와 같이 정의될 수 있다.

$$N_i = \sum_{j=1}^n a_{ij} \quad (\text{where } i \in V, j \in V) \quad (2)$$

따라서 차시 배정 절차를 정의하면 다음과 같다.

MCP에 대한 발견적 기법(MCPme)

- 단계 1: 노드 집합 V 의 각 노드에 대해 N_i 를 계산
- 단계 2: 최소 N_i 를 가지는 노드 인덱스 i 를 탐색
- 단계 3: $V' = \{i\} \cup \{c_1, c_2, \dots, c_s\}$ 과 $E' = \{e_{ij} \mid a_{ij} = 1 \text{ for } j \in \{c_1, c_2, \dots, c_s\}\}$ 를 만족시키는 서브 그래프 $G_i = (V', E')$ 를 구성
- 단계 4: G_i 로부터 최대 클릭을 가지는 노드 집합 $\{1, 2, \dots, m\}$ 를 찾아서 k 번째 타임슬롯 T_k 로 배정
- 단계 5: $V = V - \{1, 2, \dots, m\}$ 와 $E = E - \{e_{ij} \mid (1 \leq i \leq n, 1 \leq j \leq m)\}$ 로 구성되는 새로운 그래프 $G(V, E)$ 를 가지고 단계 1부터 반복해서 적용

4.2 동시시험 가능 과목들의 동일 타임슬롯 배정

타임슬롯 배정을 위한 알고리즘 MCPme의 4번째 단계에서 Max Clique을 찾는 과정은 주어진 교과목 집합에 대해 동일한 타임슬롯의 배정이 가능한 과목들의 최대 집합을 찾는 것이다. Max Clique을 구하는 문제 자체는 NP-Hard 문제로서 많은 경우 [10]에서처럼 발견적 기법이나 근사접근법(Approximate Approach)에 의해 해를 찾고 있다. 여기서는 MCPme의 단계 2에서 최소 아크 수를 가지는 노드와 연결되어 있는 노드들의 집합으로부터 Max Clique을 구해가는 접근법을 채택하고 있기 때문에 문제 크기가 급격하게 커지는 경향을 피할 수 있다. 따라서 Branch and Bound 방법에 의해 최적해를 직접 찾아가도록 한다.

최대 클릭 탐색 알고리즘(MaxClq)

- 단계 1: $V' = \{1, 2, \dots, m\}$ 의 각 노드에 대해 N_i' 를 계산하고, 클릭 크기의 상한 u 를 $m-1$ 로 설정 And set clique size bound u as $m-1$.
- 단계 2: $N_i' \geq u$ 인 노드의 수 z 를 계산
- 단계 3: $z \leq u$ 이면 $u = u-1$ 로 설정하고 단계 2로 진행
- 단계 4: $N_i' \geq u$ 인 노드들로 선택된 노드 집합 S 를 구성
- 단계 5: 노드 집합 S 가 클릭을 형성하는 지 검사하고, 클릭을 형성하지 못하면, $u = u-1$ 로 설정해서 단계 2로 진행
- 단계 6: 노드 집합 S 를 최대 클릭으로 설정

4.3 노드 인덱스 결정 규칙

타임슬롯 배정을 위한 알고리즘 MCPme의 단계 2에서 최소 N_i 를 가지는 노드를 선택하는 경우 동일한 값을 갖는 노드들이 다수 존재할 수 있다. 이때 무작위로 노드 인덱스를 결정하는 방법도 있겠지만, 전체적으로 배정된 타임슬롯을 최소화하기 위해서는 일정한 규칙을 적용해서 노드 인덱스를 결정하게 되면 보다 좋은 결과를 얻을 수도 있다. 따라서, 식 (2)와 같이 단순히 단일 노드에 집중된 아크 수뿐만 아니라, 각 과목별 수강 학생 수 그리고 식 (3)처럼 해당 노드와 연결된 모든 노드들이 가지고 있는 아크의 합으로 아크 집중도를 계산해서 이 세 가지 기준을 적절히 선택함으로써 타임슬롯을 배정할 수 있다.

$$W_i = \sum_1^n N_j a_{ij} \quad (1 \leq j \leq n) \quad (3)$$

5. 실험과 결과

제안된 발견적 기법을 통해서 한양사이버대학교에서 실제 운용했던 실시간 시험에 적용해 보기 위하여 Visual Basic 6.0을 이용해서 코딩을 하였다. 2005년 1학기부터 2006년 1학기까지 시행되었던 6번의 실시간 시험시간표를 작성하는 실험을 실시했으며, 결과는 <표 1>과 같이 나타났다. 실제 편성대상이 되는 과목의 수는 적게는 65과목에서 118과목에 이른다.

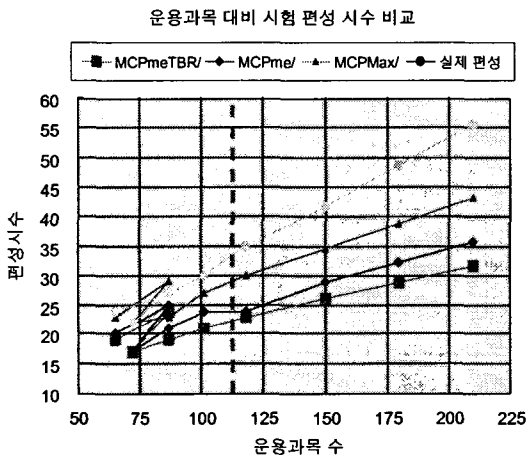
<표 1> 실험결과

		2005년 1학기		2005년 2학기		2006년 1학기	
		중간	기말	중간	기말	중간	기말
과목수		65	87	72	87	101	118
응시생수		13,124	15,963	10,603	12,476	18,891	21,847
MCPme TBR/	편성시간수	19	24	17	19	21	23
	평균	690.7	666.0	623.7	656.6	899.6	949.9
	표준편차	287.5	311.1	303.5	297.7	337.9	351.6
MCPme/	편성시간수	20	25	17	21	24	24
	평균	656.2	639.3	623.7	594.1	787.1	910.3
	표준편차	345.0	322.7	210.5	314.4	380.3	339.4
MCPMax/	편성시간수	23	29	22	23	27	30
	평균	570.6	515.6	482.0	699.7	699.7	728.2
	표준편차	287.2	347.4	235.3	398.1	398.1	488.1
실제 편성	편성시간수	24	27	22	27	30	35
	평균	546.8	529.0	482.0	462.1	629.7	624.2
	표준편차	300.8	250.7	217.0	239.7	279.5	305.9

MCPme는 최소 N_i 를 가지는 노드를 무작위로 선택하는 방법이고, MCPmeTBR은 4.3에서 제시한 노드 인덱스 결정 규칙을 조합해서 적용한 결과이다. MCPMax는 본 연구에서 제안된 방법과는 달리 연결된 아크 수가 가장 많은 노

드를 중심으로 시간표를 편성해가는 방법을 적용했을 때의 결과이다. 이 표에는 수작업에 의해 편성되어 실제로 운영된 결과도 함께 제시되어 있다. 실험결과에 따르면 노드 인덱스 결정 규칙이 적용된 MCPmeTBR이 가장 좋은 성능을 보여주고 있다.

<그림 6>은 실제 시험 편성 대상 과목이 증가할 경우 제안된 알고리즘에 따른 편성 결과를 예측한 내용을 보여주고 있다.



<그림 6> 운용과목 대비 시험 편성 시수 예측

실험에 따르면, 시험과목 수가 급격히 증가하더라도 제안된 알고리즘에 의해 시험 시간표를 편성할 경우 일정 수준에서 시험시간표를 편성할 수 있게 함으로써 정해진 시험기간 안에 실시간 시험을 적절히 운용할 수 있음을 확인할 수 있었다.

6. 결론 및 추후연구

본 연구에서는 사이버대학과 같이 대규모 강좌를 운영하는 기관에서 다수의 교과목에 대한 온라인 실시간 시험 운영을 위한 시험계획표 작

성 방안을 제시하였다. 실시간 시험계획표 작성 문제는 MCP 문제와 같은 NP-Hard 문제로 모델링이 될 수 있는데, 실제 적용을 보다 용이하게 하기 위하여 발견적 기법을 제안하였다.

제안된 기법은 전체 타임슬롯의 수를 최소화하는데 필요한 복잡도를 $O(n)$ 수준으로 줄이고 있다. 하지만 동일한 타임슬롯에 편성할 수 있는 최대 과목집합을 찾는 문제에 대한 복잡도는 낮추지 못하고 있는데, 대신 이전 단계에서 문제의 크기를 줄이는 방법으로 이를 보완하였다.

제안된 방법에 따른 실험 결과는 비교적 뛰어난 성과를 보여주고 있는데, 특히 최적 노드를 포함하는 서브 그래프로부터 Max Clique를 찾아가도록 노드 인덱스를 결정하는 규칙들을 조합해서 적용했을 때 가장 좋은 수행도를 보여주었다. 또한 실험 결과, 향후 운용 교과목 수가 현재의 2배 이상 증가하더라도 적절한 수준의 타임슬롯 내에서 시험시간표를 편성할 수 있도록 함으로써 실제 적용의 효용성을 보인다고 하겠다. 최근 일반 오프라인 대학에서도 점차 사이버 강의를 도입하고 있는 추세를 고려했을 때 본 연구 결과의 유용성은 더 커질 것으로 기대된다.

이후 연구에서는 편성된 시험시간표가 주어진 입력에 대해서 최적 시험시간표와 비교했을 때 얼마나 좋은 결과를 보이고 있는지를 확인하는 작업이 진행될 예정이고, 나아가 산출물로서 시험시간표뿐만 아니라 계산에 소요되는 시간적인 측면에서의 효용성도 밝힐 예정이다.

참고 문헌

- [1] 김대진, 김철현, "별칭 함수에 기반한 유전 알고리즘을 사용한 강의 시간표의 자동 작

성”, 정보과학회논문지(C), 제2권 제3호, 1996, pp. 317-325.

- [2] 김분홍, 이광호, “그래프 착색 방법에 기초한 대학교 시간표 작성 자동화”, 한국정보과학회 가을 학술발표논문집, 제26권 제2호, 1999, pp. 42-44.
- [3] 박상혁, 안종일, 김명재, 정태충, “시간 길이가 다른 과목들의 시간표 작성을 위한 그래프 칼라링 알고리즘 개발”, 한국정보과학회 가을 학술발표논문집, 제23권 제2호, 1996, pp. 345-348.
- [4] 산업자원부, 한국전자거래진흥원, 한국 U러닝연합회. 2005~2006 이러닝백서 서울, 한국 U러닝연합회. 2006.
- [5] 이호중, 전건욱, “발견적 알고리즘을 이용한 강의시간표 작성에 관한 연구”, 오늘의 국방경영분석, 제20권, 2004. pp. 104-137.
- [6] 하정탁, 김명재, “Genetic Algorithm을 이용한 시간표 작성에 관한 연구”, 한국정보과학회 봄 학술발표논문집, 제20권 제1호, 1993, pp. 229-232.
- [7] 황경순, 전중남, 이건명, “분산 제약조건 만족 특성을 이용한 다중 에이전트 기반 강의 시간표 자동화 시스템 설계”, 한국정보과학회 봄 학술발표논문집, 제29권 제1호, 2002, pp. 283-285.
- [8] Horowitz, E. and Sahni, S., *Fundamentals of Computer Algorithms*, Computer Science Press, 1978.
- [9] Swamy, M. N. S. and Thulasiraman, K., *Graphs, Networks, and Algorithms*, New-York, Hohn Wiley & Sons, 1981.
- [10] Battiti, R. and Protasi, M., “Reactive local search for the maximum clique problem”, *Algorithmica*, Vol. 29, No. 4, 2001, pp. 610-637.

□ 저자소개



박 찬 권

서울대학교에서 산업공학을 전공하였고, 동 대학원에서 석사 및 박사 학위를 취득하였다. 서울대학교 자동화시스템 공동연구소에서 Post Doc을 거친 후, 영산대학교 정보경영학부에 재직하였으며, 현재는 한양사이버대학교 경영학부에 재직 중이다. 현재 한국전자거래진흥원의 전자문서위킹그룹위원장을 맡고 있으며, 주요 관심분야는 ERP/SCM, 정보시스템 개발 방법론, XML, 전자문서 개발방법론 및 표준화 활동 등이다.



김 형 도

서울대학교 산업공학과에서 학사, 그리고 한국과학기술원 경영과학과에서 석사와 박사 학위를 취득하였다. 현재 한양사이버대학교 부교수로 재직하고 있으며, ECIF 전자문서기술위원회 부위원장, 전자거래진흥원 비즈니스 프로세스 위킹그룹 위원장 등을 맡고 있다. ebXML 전문위원회 위원장을 역임한 바 있다. 새로운 전자거래 기반을 중심으로 연구개발, 교육 및 표준화 활동을 활발히 전개하고 있으며, Decision Support Systems, ACM SIGMOD Record, IEICE Transactions on Communications, Expert Systems with Applications, Int'l Journal of Management Science 등에 다수의 논문을 게재하였다. 저서로는 B2B 전자상거래@XML, 전자상거래 원론, 전자문서 용어사전, 데이터마이닝의 이해 등이 있다. 주요 관심분야는 XML, 전자상거래, 비즈니스 프로세스, 디지털 워터마킹, IT 아키텍처, 데이터마이닝 응용 등이다.



염 지 환

한양대학교에서 무역학을 전공하였고, University of Nebraska에서 경영학 석사 (Master of Art in management) 를, US International Univer-

sity에서 전략경영 전공으로 경영학 박사(Doctor of Business Administration) 학위를 취득하였다. 현재 한양 사이버 대학교 경영학부 교수로 재직 중이다. 주요 관심 분야는 정보기술 전략과 기업성과, e-비즈니스 전략, e-learning, 전략 유형별 조직 성과 등이다.