

平成 28 年度卒業研究論文

# 空き時限の回避を考慮した履修登録表生成

法政大学 理工学部 経営システム工学科

経営数理工学研究室

13x4012 植竹 智弘

13x4033 笹川 翔太

指導教員 五島 洋行 教授

学科名	経営システム工	学籍番号	13x4012 13x4033
申請者氏名		植竹 智弘 笹川 翔太	
指導教員氏名		五島 洋行	

## 論文要旨

論文題名	空き時限の回避を考慮した履修登録表生成
------	---------------------

本研究では、整数計画問題として空き時限の回避を考慮した履修登録表を生成する。重みを用いて、重みの合計が最大になるように目的関数を設定して、空き時限の回避を考える。履修登録表生成は時間割編成問題に類似しており、手作業で行われる事が多く手間がかかる業務である。また各大学独自の情報をもとに作成するため、同一なシステムはない。

履修登録表の生成にあたり問題点が三つある。一つ目は手作業で行うため時間がかかる。二つ目は空き時限が発生する場合がある。三つ目は学期毎に作成することが多く、授業に一貫性がない場合がある。したがって本研究では、これらを考慮した履修登録表生成を試みる。

上記の問題点を考慮することにより、4年間を通じて一貫性のある空き時限の少ない履修登録表が生成できる。空き時限が減ることにより学習効率の向上が期待できる。

# 目次

1.はじめに .....	1
1.1 研究背景.....	1
1.2 研究目的.....	2
2.類似研究 .....	3
3.関連知識 .....	4
3.1 履修登録表生成.....	4
3.2 列挙法 .....	5
3.3 SCIP ソルバーとは.....	5
3.4 使用データ .....	6
4.提案手法 .....	7
4.1 前提条件.....	7
4.1.1 対象科目.....	9
4.1.2 ユーザーの入力.....	9
4.2 問題の定式化 .....	10
4.2.1 目的関数.....	10
4.2.2 登録表生成の制約条件.....	13
5.実行結果と考察 .....	16
5.1 実験環境.....	16
5.2 実行結果.....	16
5.3 考察 .....	20
6.おわりに .....	21
参考文献 .....	22
謝辞.....	23

## 1.はじめに

時間割編成問題とは、スケジューリング問題である。主に高校の時間割、大学の講義の時間割、試験の時間割のようなスケジューリングである。その大学の時間割をもとに学生は必修科目や教養科目、複数の専門分野の科目を選択する。また各分野別に卒業に必要な単位数が設けられている等の複雑な制約の中、履修登録表を作成する。それにより学生自身が手作業で履修登録表を作成するため、その際時間がかかり、空き時限ができることが考えられる。また、学期毎に履修登録表を作成するため、授業に一貫性がない場合もある。

本論文では、空き時限の回避を考慮した履修登録表の作成を提案する。従来の手動による履修登録表生成方法では、選択する際に空き時限の存在する時間割表となっていた。そこで、整数計画法を用いて、目的関数に空き時限の大小で変化する重みを導入し定式化を行う。その重みの合計が最大になるよう選択し、その問題を解決する。また、どの分野の科目をより多く選択したいかについても考慮し履修登録表の作成を行う。この章では、時間割作成問題の背景を述べた後、研究目的を紹介する。

### 1.1 研究背景

一般の大学では単位制度が設けられている。大学を卒業し学位を得るには、大学の定める必要単位数を修得する必要がある。単位を修得するには、科目を履修登録する必要がある。履修登録とは、各年度の初めに一括して行う科目登録のことであり、未登録の科目は受講し試験を受けても単位修得はできない。一般的に大学における履修登録表は、学生が自分自身で作成することになっている。科目履修には満たすべき条件があり、必修科目の取得や必要単位数等により、時間割の作成には多くの時間を費やすことが予想される。そして作成すると空き時限が多くなる事がある。また、各大学におけるカリキュラムは社会のニーズ、技術の進歩等と共に変化していき、実際に各年度を比較しても年々異なってきた。大学におけるカリキュラムが変化しているということは、1年後において制約条件が異なることが考えられる。同じ学科名の学科だとしても異なる大学であれば、それぞれの大学のカリキュラムは多少近いものもあるが同一のものは存在しない。そのため履修登録表を作成するための有効な手法が見つけられていない。その時々に合わせて改良した時間割の作成手法の開発が必要である。問題点として、手作業のため時間がかかる。空き時限ができてしまう。学期毎に作成するため授業に一貫性がない。システム化するとユーザーの意思が反映されない履修登録表が作成される。

そこで、本研究では、空き時限の回避を考慮した履修登録表を作成する。またその際にユーザーの意思が反映されるように、制約の段階で選択できるようにした。そして学期毎に作成すると授業に一貫性がなくなる可能性があるため、1年生春学期4年生秋学期までの履修登録表を同時に作成し、多く学習したい専門分野に偏らせて、履修登録する。

## 1.2 研究目的

大学では、科目を履修しその単位を修得するには、履修科目登録をする必要がある。また、手動による学生の科目選択は自由度が大きく、複数の制約を同時に満たすように選択すると、履修登録表の作成は困難である。それは特に大学1年生の履修登録表の作成において言える。履修登録や各科目に対し知識の無い1年次の学生は、入学当初に配られる理工学部のための履修の手引きやweb上から参照できる授業概要をまとめたシラバスから各科目の詳細を知る必要がある。一方、大抵の学生は時間割を作成する際、現在の学年とその学期のみを考慮した時間割の作成にあたる。しかしその時間割の作成方法では、いくつかの問題に悩まされる。仮に1限目と3限目に選択したい科目があったとしても、履修単位数の上限によって2限目を入れられなくなることが考えられる。このようにして選択した科目同士の間空き時限ができてしまうことが考えられる。また、学科毎にある学習分野を広く取ることになり、結果それぞれの分野に対しての知識が浅いものになることも考えられる。

そこで本研究では、上記の問題を解決する履修登録の作成手法を提案する。空き時限の回避を考慮した履修登録をすることにより、空き時限を減らし、より充実した大学生活を送れる。空き時限に明確な予定がある場合は、時間を無駄にしないが、そうでない人にとっては空き時限は好ましくないため、無駄な時間を減らすことになる。また、ユーザーの意思を反映できる部分に関しては、複数のパターンに対応できるため、一人の学生のための履修登録表作成とはならず、汎用性がある。そして、多く学習したい専門分野に偏らせるため、4年間を通じての学習効果が上がる。受けた授業の基礎的な授業を前年に受け忘れたなどのことがなくなる。

## 2. 類似研究

西森ら[2]の研究では、より複雑な大学の時間割を制約充足問題により編成する方法を提案している。大学の時間割編成問題は、課せられる制約条件が多種多様であり、また探索空間が膨大である、という2つの問題点がある。制約を、個別の科目への制約、2科目間の関係に関する制約、時間割表全体の整合性に関する制約に分類整理することにより、前者の問題点を解決する。また、後者に対しては、制約に違反点数を与え、その違反点数の最小化を規範とする山登り法を導入し、さらに対話によってユーザーと協調して探索を進めている。実行し生成する時間割表には、表のすべての科目に対し総合違反点数を表示する。その総合違反点数を減少させるようにユーザーへ提案する。その結果、違反点数を減少させた時間割表を生成する。

柿本ら[3]の研究では、時間割編成問題を制約充足問題として定式化し、単体法を用いて厳密解を求めている。制約条件の中に問題を難しくする制約がある。その問題を困難にする制約に対し違反を許しその違反の合計を最小化する方法を提案する。

本研究では、西森ら[2]が述べた時間割編成問題での2つの問題点の解決を提案する。これに対し西森ら[2]、柿本ら[3]を参考にする。前者の問題点に対しては、個別の科目への制約を細かくすることを参考に解決する。後者の問題点に対しては、点数を与え違反点数を最小にする手法を参考にし、本研究では重みの合計を最大にする手法を提案する。また対話によってユーザーの意思を強調する手法を参考に解決する。

### 3. 関連知識

本章では、本研究で用いる手法に関する知識について紹介する。本研究で扱う履修登録表生成問題について記述し、その後、すべての科目の組み合わせに対して空き時限を考慮するために使用する列挙法、実例として扱った使用データについて紹介する。

#### 3.1 履修登録表生成

オペレーションズ・リサーチや数理計画法において整数計画問題とは、線形計画問題の変数の一部または全部に整数条件が付加されたものである。線形計画問題とはいくつかの1次不等式および1次等式を満たす変数の値の中で、ある1次式を最大化または最小化する値を求める問題である。本研究の整数計画問題は、扱う式はすべて1次式からなっているので、厳密には整数線形計画法問題というべきものである。整数計画法問題は、一般に次のように定義される[1]。

[整数計画問題]

整数である $x_1, x_2, \dots, x_n$ の変数があり、その変数の組み合わせにより最小化されるべき関数となる目的関数があるとする。また、その変数に満たすべき条件である制約条件があるとする。そのとき、制約条件を満たす $x_1, x_2, \dots, x_n$ の変数の組み合わせのうち、目的関数を最大化または最小化させるなど最適にする変数の組み合わせを求めよ。

このような形で定式化される組み合わせ最適化問題はよく知られており、巡回セールスマン問題、ナップサック問題などが代表的な例として知られている。これらの整数計画問題の多くは、変数が0または1の値しかとらない $\{0, 1\}$ 計画問題である。また、 $\{0, 1\}$ 計画問題以外の整数計画問題であっても、適当な変数をほどこすことにより、 $\{0, 1\}$ 計画問題に帰着できる問題がかなり多く存在する。本研究における、変数も $\{0, 1\}$ 変数を使い、科目の表現をしている。以上の理由により、本研究では $\{0, 1\}$ 変数で表された、0-1計画問題として扱うことにする。

時間割作成問題とは、基本的に科目が実施される学年、および時間帯をまとめた時間割を編成する問題のことである。各科目には担当する教官がおり、その教官が担当可能な時間を設定する必要がある。また教室も被ることのないように振り分ける必要がある。

本論文における履修登録表生成問題とは、大学から提供される各科目が実施される時間帯の表から、履修登録をする表を作成する問題のことである。1年次から4年次の各学年の春、秋学期の8つの時間割を作成することを目的とする。時間割作成問題は、制約充足問題、最適化問題として研究なされている。しかし時間割作成問題を解くには、「卒業要件を満たすように科目を選択し単位数を確保しなければならない」等の制約が存在しており、それらを満たすように時間割を作成しなければならない。本論文では実例として、法政大

学理工学部経営システム工学科の時間割作成問題について取り上げる。時間割編成問題を解くということは、すべての制約を満たすように科目を選択し、月から金、1限目から5限目までの表に科目を割り当てることである。本問題では、空き時限に対し重みを与え、空き時限が少なくなるにつれ重みが大きくなるように設定した。そこで本研究では重みの価値の合計点数を最大化する整数計画問題としてとらえることにする。

## 3.2 列挙法

多くの組み合わせ最適化問題の許容領域は、有限個の要素を持つ。この場合すべての許容解 $x \in S$ を列挙し、その中で目的関数 $f(x)$ を最大にする解を選べばそれが最適解である。このような解法を列挙法と呼ぶ。このように列挙法とは、与えられた条件を満たすものを漏れなく、重複なく出力する手法である。列挙アルゴリズムとは列挙問題を解くためのアルゴリズムである。列挙アルゴリズムとして、部分集合列挙問題などが挙げられる。部分集合列挙問題とは、自然数  $n$  が与えられたとき、 $\{1, 2, \dots, n\}$ の部分集合をすべて出力する問題である。

本論文では、ある1日の1時限目から5時限目に、選択可能な科目がそれぞれ複数あるとき、1限から5限に存在する選択可能な科目の個数を、かけ合わせた組合せを考えて問題に取り組んでいる。組合せは積で表現しているので、膨大な数の組合せが存在する。そこで本研究では、列挙法により、すべての科目の選択の組み合わせの中から最適な解を出力する。

## 3.3 SCIP ソルバーとは

SCIP ソルバーとは、ドイツの Zuse Institute Berlin で開発されている非商用ソルバーである。数ある非商用ソルバーの中で SCIP を利用する理由は以下のとおりである。

- ・非商用ソルバーとしては、高速なものの1つである。
- ・バイナリ版が準備されており、コンパイルせずにすぐに使える。
- ・コマンドラインインターフェースがシンプルで使いやすい。
- ・LP ファイルに対応しているため、商用ソルバーへの切り替えが容易である。
- ・アカデミックユースは無償である。

LP ファイルとは、多くのソルバーが扱うことのできるファイル形式の一種で、数理計画問題を記述した、拡張子が.lp のテキストファイルである。



### 3.4 使用データ

本研究で時間割作成する対象は、法政大学理工学部経営システム工学科の時間割とする。2016年度理工学部経営システム工学科時間割から、科目名、科目が行われる時間帯をデータとして使用する。法政大学における教育課程は、すべて単位制度が採用されており、試験、レポート等による成績評価の結果合格することによって、単位を修得することが可能となっている。単位は春学期・秋学期、各年度や卒業所要単位等にそれぞれに対し制限が存在している。単位に関する情報や進級条件・卒業要件、履修単位数の上限等に関して、「2016年度理工学部生のための履修の手引き」に収録されている内容を使用する。

## 4.提案手法

履修登録表の生成を整数計画問題とみなす。そのため、目的関数、制約条件について定義する。まず、目的関数と制約条件で使用される記号や定数、決定変数を定義する。目的関数は、空き時限を回避するために、重みを用いて、その合計を最大化するように設定する。目的関数で使用される記号について、目的関数を設定した後、詳細を記述する。制約式では、大学によって定められている制約や、履修をする際の前提となる制約について提案する。

大学によってかなり事情は異なると思われるので、ここでは、本大学の経営システム工学科の2013年度入学者用の時間割表を例に取り検討してみる。時間割作成を困難にしている原因は以下のものがある。

- ・同一時間に選択できる科目が複数存在する
- ・複数の同名の科目が存在する
- ・各科目には異なる数値の単位、必修科目や専門科目等の情報が含まれている

このような原因を、整数計画法を用いて定式化する。

### 4.1 前提条件

この節では、本モデルを定式化するにあたって使用する記号の定義を述べた後、本論文で使用した対象科目について述べる。

次に記号の定義を行う。

$S_{abcde}$  : 本研究で扱う全ての科目

科目を表す変数 $S_{abcde}$ により、全ての科目名と変数が対応する。以下表1は1年生春学期の月曜と火曜日の科目と変数の対応表である。表1の場合、数理技法入門は学年 $a = 1$ 、春学期 $b = 1$ 、月曜日 $c = 1$ 、4時限目 $b = 4$ 、一つ目 $e = 1$ の科目であるため、 $S_{11141}$ と表現する。

#### 添字

$a$  : 学年 = {1:1 学年, 2:2 学年, 3:3 学年, 4:4 学年}

$b$  : 学期 = {1:春学期, 2:秋学期}

$c$  : 曜日 = {1:月曜日, 2:火曜日, 3:水曜日, 4:木曜日, 5:金曜日}

$d$  : 時限 = {1:1 限目, 2: 2 限目, 3: 3 限目, 4: 4 限目, 5: 5 限目}

$e$  : 同一時間に存在する科目を並べたときの順序

$f$  : ある1日における科目選択の組み合わせ番号

表 1. 科目の対応例

	月	火		
1	環境と資源	s(1,1,1,1,1)	パーソナリティー	s(1,1,2,1,1)
	日本文化論	s(1,1,1,1,2)		
2	化学基礎 I	s(1,1,1,2,1)	自然科学の方法	s(1,1,2,2,1)
	日本文化論	s(1,1,1,2,2)	パーソナリティー	s(1,1,2,2,2)
	社会科学の方法論	s(1,1,1,2,3)		
	言語学概論	s(1,1,1,2,4)		
3	日本文化論	s(1,1,1,3,1)	英語中級リーディング	s(1,1,2,3,1)
	生物の多様性	s(1,1,1,3,2)		
4	数理技法入門	s(1,1,1,4,1)	コミュニケーションストラテジー	s(1,1,2,4,1)
5			英語中級コミュニケーション	s(1,1,2,4,2)
			科学技術史	s(1,1,2,5,1)

### 決定変数

本研究で扱う全ての科目を以下の変数 $S_{abcde}$ で表す. 科目選択の有無を $\{0, 1\}$ 変数で表す.

$$S_{abcde} = \begin{cases} 1 & : \text{科目を選択する} \\ 0 & : \text{科目を選択しない} \end{cases}$$

以下の変数 $Y_{abcf}$ により, 想定する科目選択の組み合わせの有無を $\{0, 1\}$ 変数で表す.

$$Y_{abcf} = \begin{cases} 1 & : \text{想定する科目選択の組み合わせがある} \\ 0 & : \text{想定する科目選択の組み合わせがない} \end{cases}$$

### 定数

本研究で扱う大学の科目の単位は全て $\{1, 2\}$ である. 各科目に存在する単位数を以下の変数 $T_{abcde}$ で表す. すべての科目に含まれる単位を, 科目に応じて与える.

卒業要件, 進級条件等の条件は, 科目の合計単位数を考慮した条件である. ある科目の単位を表す場合, その科目が選択されるとき単位を付加させることにより, 合計単位数による条件を考慮する.

$$T_{abcde} : \text{単位} = \{1, 2\}$$

本研究では空き時限を回避するために目的関数で重みを用いる. 重み $w$ の設定は目的関数の章で提案する.

$$w_{abcf} : \text{重み} = \{94, 96, 98, 100\}$$

$t$ : ある 1 日の選択可能な科目が存在する時限数

$m$ : 1 年次から 4 年次の春, 秋学期における 1 日の組み合わせの総数

$L$ : 各合計単位数の下限の集合

各合計単位数の集合として, 考慮すべき項目を下に記す. これらはすべて大学の設定する単位に関する制約であり, 理工学部生のための履修の手引きを参照している.

- ・前期単位数（後期単位数） $\leq 30$
- ・年間単位数 $\leq 49$
- ・4年間の総単位数 $\geq 124$
- ・3年次終了時の単位数 $\geq 90$
- ・英語 $\geq 8$
- ・教養 $\geq 12$
- ・理系教養 $\geq 12$
- ・専門科目 $\geq 80$

### 4.1.2 対象科目

本研究は、実例として2013年度入学者用の履修登録表の作成を対象としている。対象科目は、すべて時間割に記載されている教養科目と専門科目である。教職課程の科目は、選択しないものとした。体育Ⅰから体育Ⅳ、基礎外国語、に関して、選択から排除した。これらは、様々な時間帯に繰り返し出る科目である。また連続で行われる実験授業も選択しないものとした。空き時限を避けるため本研究では、取りうるすべての組み合わせを考慮している。そして、その組み合わせは積で表されることにより、下に記述する目的関数における文字 $Y$ が膨大な数になり、計算するときメモリが不足してしまうからである。

### 4.1.3 ユーザーの入力

理系教養系科目は、物理、化学、生物の科目がある。3つの科目にはそれぞれ基礎Ⅰ、基礎Ⅱがあり、両方の授業をとることにより理解を深められる構成になっている。つまり、物理基礎Ⅰと化学基礎Ⅰという組み合わせで履修を行うより、物理基礎Ⅰ、物理基礎Ⅱという組み合わせで履修を行うのが、理系教養科目の理解を深められると考える。また大学受験のために高校の理科科目を一つに絞って学習していた人もいる。そのためユーザーに合わせられるよう、物理、化学、生物の3科目から1つを入力するように要求し、要求された科目の基礎Ⅰ、基礎Ⅱの授業を履修するようにする。

## 4.2 問題の定式化

問題の定式化について紹介する。この節では、目的関数の設定を述べた後、目的関数に使用される記号の詳細を紹介する。その後、本研究で使用する制約式を紹介する。

### 4.2.1 目的関数

この節では、目的関数の設定について記述する。本研究でのテーマは、空き時限を回避することである。空き時限を回避することとは、例えば、ある1日の科目の選択を、2時限目と4時限目とするのではなく、2時限目と3時限目のように選択する科目同士の間を生じ得る余分な時間を減少させることである。そこで、想定する科目の選択の組み合わせに重みを導入する。そしてそれらの合計を最大化する最適化問題とする。

本研究では1日を最大5時限考えており、空き時限は、0, 1, 2, 3時限の最大4つが考えられる。4つの空き時限ごとに重み $w$ を与え、科目の選択状況の組み合わせが、空き時限を避けるよう目的関数を設定する。1年次の春学期から4年次の秋学期までの、1日の科目の組み合わせの総和 $m$ 個の重みの合計を最大化する。その目的関数は下記の式(1)である。上記の目的関数で扱う重み $w$ 、変数 $Y$ については、この節の下記で記す。

$$\text{目的関数} \quad \sum_{f=1}^m w_{abcf} Y_{abcf} \rightarrow \max \quad (1)$$

ある1日の科目の選択状況に対応する変数 $Y$ の作成手順について記述する。 $Y$ は、ある学年、学期、曜日の1日における科目の選択状況を表す $\{0,1\}$ 変数である。ここでの選択状況とは、ある1日の1限から最大5限まで、選択するか、選択しないかを考慮した組み合わせである。1日の科目の選択は、1日は最大5時限であることから、最大 $2^5 = 32$ 通り存在する。ある1日が3限、4限の場合も考慮し、その場合の組み合わせは、8通り、16通り存在する。

本研究における科目選択の有無は、 $\{0,1\}$ 変数で表し、1のとき科目選択を意味する。ある1日の1時限から5時限までの選択状況は $\{0,1\}$ 変数の組み合わせで表される。その組み合わせに対応する変数 $Y$ を作成する。32通りある組み合わせに存在する空き時限を考慮し目的関数では重みと対応させ、合計の最大を選択させる。

ここで科目 $S_d$ は、時限 $d$ における $\{0,1\}$ 変数である。ある1日の選択可能な科目が時限数 $t = 5$ 時限分存在し、科目が、 $d = 1, 2, \dots, 5$ 時限に存在する場合を考える。このときの科目選択状況に変数 $Y$ を対応させる場合を記述する。32通りの組み合わせの中から、1時限から5時限までの選択状況をそれぞれ、 $S_1 = 1, S_2 = 1, S_3 = 0, S_4 = 0, S_5 = 0$ とし、この場合にのみ $Y_{abcf} = 1$ とする変数 $Y_{abcf}$ を作成することを考える。そのとき $d = 1, 2, \dots, 5$ 時限の科目の中に最低一つでも異なる科目 $S_{abcde}$ の選択が存在するならば、

$Y_{abcf} = 0$ と対応させる．これを満たす変数 $Y_{abcf}$ を以下の式(2)で表現する．式(2)で記す式は、積で表現された数式である．さらにこれを SCIP ソルバーで扱える形へと以下の式(3)で表現する．

$$Y_{abcf} = S_1 * S_2 * (1 - S_3) * (1 - S_4) * (1 - S_5) \quad (2)$$

$$\begin{aligned} Y_{abcf} &= S_1 + S_2 + (1 - S_3) + (1 - S_4) + (1 - S_5) - 4 \\ Y_{abcf} &\leq S_1 \\ Y_{abcf} &\leq S_2 \\ Y_{abcf} &\leq (1 - S_3) \\ Y_{abcf} &\leq (1 - S_4) \\ Y_{abcf} &\leq (1 - S_5) \\ Y_{abcf} &\geq 0 \end{aligned} \quad (3)$$

上記で表した式(2)は、掛け合わせる成分すべてが1となることで、対応する $\{0,1\}$ 変数 $Y_{abcf}$ が1となる式である． $Y_{abcf}$ は1時限から5時限までの選択状況の組み合わせである．式(2)、式(3)では、 $S_1 = 1, S_2 = 1, S_3 = 0, S_4 = 0, S_5 = 0$ を考えたが、そのような組み合わせは合計32通り存在し、それらに対しても同様に表現する．式(2)、式(3)を以下の式(4)、式(5)で定式化する．

$$Y_{abcf} = \prod_{d=1}^t S_d \quad (4)$$

$$Y_{abcf} \geq \sum_{d=1}^t S_d - t + 1 \quad (5)$$

$$Y_{abcf} \leq S_d, Y_{abcf} \geq 0$$

また、1日の選択可能な時限数が $t=1$ 時限分のとき、その一つの科目の選択が、そのまま1日の選択状況になる．よって式(6)で表現する．

$$Y_{abcf} = S_d \quad (6)$$

次に、空き時限を避けるために使用する重み $w$ の設定を記述する．重み $w$ は、4.2 目的関数の式(1)で使用する．重みを付加させ、空き時限数の少ない1日の科目選択から優先して選択することにより空き時限の生成を回避する．重み $w$ は空き時限が小さくなるにつれて、大きな重みを与える．本研究で使用する重み $w$ について、生じうる空き時限と重みを対応

表 2 重み $w$ の設定

空き時限数	重み $w$
0	100
1	98
2	96
3	94

させる. 表 2 で表現する. 空き時限数とは, ある一日における選択した科目の間に生じる空き時間のことである. ある 1 日の科目選択が, 1 時限と 2 時限のように連続して科目選択している場合, 空き時限数は 0 とする. 一方, 1 時限と 3 時限を選択し, 2 時限を選択しない場合, 空き時限数は 1 とする.

表 1 の重みの設定に関して, 今回のテーマは空き時限を減少させることであり, 必ずしも空き時限を消滅させることではないので, 重みには大きな差を持たせていない. 仮に重み $w$ を $w = 94, 98, 96, 100$ の代わりに $w = 0, 1, 2, 3$ とする. その場合, 空き時限数 1 での重みは空き時限数 2 での重みの 2 倍である. このように重み $w$ を小さな数字にすることで重みに数倍もの差が生じる可能性がある. よって, 重み $w$ の数を大きくすることによって重みに何倍もの差が生じることがなくなる.

ある 1 日の選択可能な時間帯が 1 時限から 5 時限まで  $t = 5$  つあるときを考える. 科目  $S_d$  は  $\{0, 1\}$  での変数であり, 1 日の選択  $\{\text{しない}, \text{する}\}$  の組み合わせの総数は  $2^5 = 32$  通り存在する. 選択の場合分けとその重みについて, 下の表 2 を参照する. 例えば,  $S_1 = 1, S_2 = 1, S_3 = 1, S_4 = 1, S_5 = 1$  ならば, 空き時限数は 0 である. また, 選択状況  $Y$  が,  $S_1 = 1, S_2 = 1, S_3 = 0, S_4 = 1, S_5 = 0$  であるならば, 空き時限は 2 時限と 4 時限との間に生じる時限分の 1 である. 空き時限が生じるのは科目が二つ以上のときからである. よって, 選択できる科目が一つだけのとき, 空き時限は存在しない. その場合, 空き時限は空き時限を 0 とみなし重みを 100 とする. また, 時間割表では選択可能な時限数が  $t=4$  つである場合も存在する.  $t=4$  の場合は, 1 日の科目の選択の組み合わせの総数は,  $2^4 = 16$  通りである. この場合の空き時限は, 0, 1, 2 が存在する. この場合も同様に空き時限に対して重みを設定する. 選択可能な時限が  $t = 4$  のとき, 選択できる科目の時限が 1 時限, 2 時限, 4 時限, 5 時限である場合が存在する. このときも同様に 16 通りの選択状況を考えるが, 1 日の選択状況が  $(S_1, S_2, S_4, S_5) = (1, 1, 1, 1)$  であるとしても,  $S_3$  には選択できる科目がないため, 空き時間は 1 であることに注意する. これは,  $t = 2, 3, 4$  において考える.

表3 32通りの選択状況とその重み

	重み $w$	(科目 $S_1$ ,科目 $S_2$ ,科目 $S_3$ ,科目 $S_4$ ,科目 $S_5$ )
1	100	(1,1,1,1,1)
2	100	(0,0,0,0,0)
3	100	(0,1,1,1,1)
4	100	(1,1,1,1,0)
5	100	(0,0,1,1,1)
6	100	(0,1,1,1,0)
7	100	(1,1,1,0,0)
8	100	(1,0,0,0,0)
9	100	(0,1,0,0,0)
10	100	(0,0,1,0,0)
11	100	(0,0,0,1,0)
12	100	(0,0,0,0,1)
13	100	(1,1,0,0,0)
14	100	(0,1,1,0,0)
15	100	(0,0,1,1,0)
16	100	(0,0,0,1,1)
17	98	(1,0,1,1,1)
18	98	(1,1,0,1,1)
19	98	(1,1,1,0,1)
20	98	(1,0,1,0,0)
21	98	(0,1,1,0,1)
22	98	(1,0,1,1,0)
23	98	(1,1,0,1,0)
24	98	(1,0,1,0,0)
25	98	(0,1,0,1,0)
26	98	(0,0,1,0,1)
27	96	(1,0,0,1,1)
28	96	(1,1,0,0,1)
29	96	(1,0,1,0,1)
30	96	(1,0,0,1,0)
31	96	(0,1,0,0,1)
32	94	(1,0,0,0,1)

## 4.2.2 登録表生成の制約条件

空き時限を回避する提案手法の中で、目的関数で扱う変数を用意するために制約式を設けたが、その他に科目を履修する際に単位に関する満たすべき条件が存在する。そのような科目履修の際に考慮する点として、記号の定義で紹介している一つの学期、年間に対して、選択できる合計単位数の上限が存在すること等が挙げられる。それは、大学が履修の際に単位に対して設定する制約である。上に挙げた制約の他にも、履修登録表を作成する際に前提となる制約が存在し、本研究ではこれらの制約について考慮する。本研究で使用する制約式について記述する。各大学においてその満たすべき条件は定まっていない。対象にした大学では、以下の条件を考慮し、履修登録表生成を行う。また、以下に示す $CR_i$ とは、 $i$ 番目の制約条件を表している。



$CR_1$  ある学年学期曜日時限で，選択できる科目数は1以下にならなければならない：

対象とする時間割表には，一つの時間帯に複数の選択可能な科目が存在することがある．一つの時間帯に，選択する科目が一つあるとき，その時間帯に存在する選択可能な科目はすべて選択することができない．また，科目の選択は $\{0,1\}$ 変数を使用しており，目的関数を最適化するように科目の選択を行うため，必ずしもある時間帯の科目選択があるわけではない．よって以下の式(7)で表現する．

$$\sum_e S_{abcde} \leq 1, \forall a, b, c, d \quad (7)$$

$CR_2$  同一名の科目は2つ以上選択することができない：

本研究で対象にする時間割表の科目には，同一名の科目が異なる時期，時間帯に存在する科目がある．大学が科目に設定する制約として，同名科目の選択は異なる時間帯で履修することができない．また，これも目的関数を最適化する等を優先して科目選択を行うため，必ず同名科目のうち一つを選択するとは限らない．よって同名科目の合計科目数を1以下にする必要があり，以下の式(8)で表現する．

$$\sum_{a,b,c,d,e} S_{abcde} \leq 1 \quad (8)$$

$CR_3$  専門科目や卒業要件等に関して，定められた合計単位数を満たさなければならない：

全ての科目 $S$ には， $\{1, 2\}$ の単位が存在する．卒業要件や進級条件等では，専門科目等の合計単位数によって，進級や卒業を判定している．科目の選択を表す $\{0,1\}$ 変数と，その科目の単位を掛け合わせることにより，合計単位数を表現することができる．また，合計単位数に関する制約を表現するには，上限と下限の条件がある．よって，合計単位数を判定する制約を式(9)で表現する．

$$\sum_{a,b,c,d,e} T_{abcde} S_{abcde} \leq L, \sum_{a,b,c,d,e} T_{abcde} S_{abcde} \geq L \quad (9)$$

$CR_4$  必修科目等の選択 :

対象科目には, 必修科目が存在する. 必修科目は進級条件や卒業要件に関わる科目である. 他にも英語科目や理系教養科目のような卒業要件に関わる科目が存在する. 本研究では, 卒業や進級に関わる科目を必ず選択するものとして考慮する. よって, これらの科目の選択を式(10)で表現する.

$$S_{abcde} = 1 \tag{10}$$

## 5. 実行結果と考察

本研究で提案する履修登録表の生成について、法政大学理工学部経営システム工学科を例とし検証を行う。また整数計画問題を解くために非商用ソフトSCIPソルバーを用いる。

### 5.1 実験環境

実験環境は以下のとおりである。

CPU: Intel(R) Core(TM) i5-4590 CPU @ 3.30GHZ 3.30GHZ

OS: Windows 7 professional

メモリ: 4GB

言語: Microsoft Visual C++ 2010 Express Edition

### 5.2 実行結果

ユーザーの意思により理科系科目である生物学，物理学または化学から，今回は例として物理学を選択する場合の結果を出力する。本研究では，履修登録表の生成を整数計画法問題とみなし，SCIPソルバーを用いて計算する。実験に必要なとなった変数は2,657個存在し，そのうち156個を $\{0,1\}$ 変数であることの情報を与えるために用いた変数である。残り2,501個は，すべて $S_{abcde}$ と $Y_{abcf}$ による決定変数である。また，考慮する制約数は16,354個である。 $S_{abcde}$ の選択状況から $Y_{abcf}$ を決定する制約をはじめ，進級条件や卒業所要単位，一つの時間帯に二つ以上の科目が入れないなどの制約が含まれている。実験で掛かる計算時間は，27.46secである。

1年次春学期の履修登録表は図1の結果となった。空き時限を回避するために，科目選択において必ず選択することが考えられる英語科目のコンプリヘンシブ・イングリッシュIや，理系教養科目の物理学基礎Iなどの前後の時間帯に授業が履修されることが見受けられた。図1は月曜日，火曜日に科目が一つのみという結果であるが，これは，水曜日に，進級条件に関わる数学系・理科系である理系教養科目や必修科目の科目選択があり，春学期の学期制約を満たすように選択されるためである。1年次春学期は合計21単位を履修する結果となった。また，科目間には空き時間が生じることは無かった。よって本研究のテーマである空き時限の回避を満たす結果が表示された。

	月	火	水	木	金
1			物理学基礎 I		プログラミング言語C
2	社会科学の 方法論		計算機実習基 礎	アフリカ文化 論	プログラミング 言語C演習
3			線形代数学及 び演習	コンプリヘン シブ・イング リッシュ I	
4		コミュニケー ション・スト ラテジー	微分積分学及 び演習	文章作法	
5					

図 1. 1 年次春学期履修登録表

1 年次秋学期の履修登録表は以下の図 2 の結果となった。基礎数学，計算機実習応用や確率統計は，必修科目である。また，水曜日には卒業所要単位に含まれる理系教養科目である物理学基礎 II や線形代数学及び演習 II 等の科目が存在する。それらを選択し，空き時限ができないように水曜の 2 時限目に，先端技術社会論の科目を選択する結果となった。このように 1 年次春学期と同様に，選択しなければならない科目の間に生じる空き時限を優先して減少させることが分かる。1 年次秋学期は合計 21 単位を履修する結果となった。また，1 年次秋学期においても科目間に空き時限が生じることは無く，空き時限の回避を満たす結果となった。

	月	火	水	木	金	土
1		基礎数学	物理学基礎 II		計算機実 習応用M	
2			先端技術社会論	確率統計		音楽芸術
3			線形代数学及び 演習 II	コンプリヘンシ ブイングリッ シュ II		
4			微分積分学及び 演習 II	経済学		
5			哲学入門			

図 2. 1 年次秋学期履修登録表

2年次春学期の履修登録表は以下の図3の結果となった。数理統計学，経営工学計算演習基礎やオペレーションズ・リサーチⅠは，必修科目である。2年次春学期は合計19単位を履修する結果となった。また，2年次春学期においても科目間に空き時限が生じることは無く，空き時限の回避を満たす結果となった。

2年次秋学期の履修登録表は以下の図4の結果となった。オペレーションズ・リサーチⅡ，経営工学計算演習応用は，必修科目である。2年次秋学期は合計19単位を履修する結果となった。また，2年次秋学期においても科目間に空き時限が生じることは無く，空き時限の回避を満たす結果となった。

	月	火	水	木	金
1	数値解析		アカデミック・ リーディングⅠ		数論
2	数理統計学	経営工学計 算演習基礎			企業財務論
3	アカデミック・ ライティング	プログラミ ング言語 JAVA		オペレーション ズ・リサーチⅠ	
4	経済性工学				
5					

図3. 2年次春学期履修登録表

	月	火	水	木	金
1	オペレーショ ンズ・リサー チⅡ		アカデミック・ リーディングⅡ	応用解析	
2	応用代数学	経営工学計 算演習応用	リスク管理論		
3	産業経済論		応用確率論		
4	工業会計論		アクチュアリー 数理		
5					

図4. 2年次秋学期履修登録表

3年次春学期の履修登録表は以下の図5の結果となった。経営工学基礎演習は、必修科目である。3年次春学期は合計20単位を履修する結果となった。また、3年次春学期においても科目間に空き時限が生じることは無く、空き時限の回避を満たす結果となった。

3年次秋学期の履修登録表は以下の図6の結果となった。PBLは、必修科目である。3年次秋学期は合計15単位を履修する結果となった。また、2年次秋学期においても科目間に空き時限が生じることは無く、空き時限の回避を満たす結果となった。

	月	火	水	木	金
1					
2	生産情報工学			金融工学	
3	生産システム工学		ポートフォリオ理論	社会システム設計論	数理解析
4	組み合わせ最適化	経営工学基礎演習	保険数理論		経営システム特別講義
5					

図5. 3年次春学期履修登録表

	月	火	水	木	金
1					
2	非線形計画法			TQM	公経営論
3	符号暗号理論	国際経営分析		公共経済学	
4		PBL	管理会計論		
5					

図6. 3年次秋学期履修登録表

## 5.3 考察

上記のすべての実行結果では、科目間に生じうる空き時限を回避した履修登録表の生成となった。必修科目や卒業所要単位に関係する科目を基準として、その科目の前後に空き時間を埋めるように科目が選択されたことが実行結果に共通していえる。

一方、一般に学生が考慮するであろう条件を満たさないことが見られた。例えば、今回の実行結果では、ある1日の科目の選択が、一つとなる場合が存在した。その反対に1日の選択する科目が多すぎる場合が存在した。特に1年次秋学期の水曜日は、1限目から5限目まで科目がすべて選択される結果となった。

## 6.おわりに

本研究では、空き時限の回避を考慮した履修登録表の自動生成を行った。各大学を比較しても同じカリキュラムは無い。また技術の進歩により大学内のカリキュラムも年々変化している。そのため実例として法政大学経営システム工学科2013年度入学者を対象とした。生成の際に、本研究のテーマである空き時限の回避を考慮する。すべての選択状況を考えてその選択状況一つ一つに対しに重みを与え重みを最大にすることで、空き時限を回避する履修登録表の生成をすることができた。また従来の手作業から本システムでは自動による履修登録表の生成を提案する。科目に対する知識がほとんど無い状態でも作成できるため、結果として履修登録表生成の掛かる時間は減る。科目の分野に対しユーザーへ提案を設けることで、ユーザーの意思に沿った授業に一貫性のある履修登録表が完成した。

本研究で提案する履修登録表生成により、空き時限の回避を考慮できた結果が出力されたが、生成する表には、1日に一つのみの科目選択がされる場合が存在した。現実的には、1日の科目選択を複数になるように選択することが考えられる。その点でまだまだ実用的ではないが、本研究における空き時限とは、科目間に生じる空き時間のことであり、上記については考慮していないため、論文の趣旨を満たしている。今後、この論文を発展させるときに、科目選択がある日は、二つ以上の科目を選択することなどが考えられる。また、本システムでは実用的ではないものの、満たすべき制約は考慮することができており、様々な場合で活用できると考える。例えば、大学におけるカリキュラムは、年々変化しているが、その都度変化する科目などを変更することで今後の履修登録表生成を行うことが考えられる。



## 参考文献

- [1] 今野浩:「整数計画法」, 産業図書 (1981)
- [2] 西森雄一, 狩野均, 西原清:「制約に基づく対話型時間割編成システム」, 情報処理学会論文誌, Vol. 38, No. 6, pp.1094-1102 (1997)
- [3] 柿本陽平, 高橋弘毅, 島川陽一:「制約充足問題を線形計画法で解く場合の制約条件とその評価-時間割編成を例に」, 日本経営工学会論文誌, Vol. 66, No. 4, pp. 348-354 (1981)