5M-2

数理計画法を用いた時間割作成アルゴリズムの考察

植村 隆[†]　　神田 毅[‡]
近畿大学工業高等専門学校 専攻科 生産システム工学専攻 電気電子工学[†]
近畿大学工業高等専門学校 総合システム工学科 情報コミュニケーションコース[‡]

1． 背景

学校の時間割の自動作成には問題点が多い。そこで、できる限り複雑な場合に対応しながら、時間割作成を支援する方法を考察したい。本研究では数理計画法を用いることにした。

2． 時間割作成問題の定式化

時間割作成問題を数理計画問題として定式化する時に用いる変数を、次のように定義する。

$$X_{c,s,d,p} = \begin{cases} 1 & (\text{クラス } c \text{ が科目 } s \text{ を曜日 } d \text{ の時限 } p \text{ に受ける}) \\ 0 & (\text{クラス } c \text{ が科目 } s \text{ を曜日 } d \text{ の時限 } p \text{ に受けない}) \end{cases}$$

さらに、様々な制約条件を数式で表すことが重要となる。簡単な時間割で扱う制約条件を以下に示す。

単位数制約	：	クラス c が科目 s を週に何限受けるかを制限するための制約条件
教員制約	：	同じ教員が、同じ時間帯に別のクラスで授業を持たないようにするための制約条件
学生制約	：	1 限目に必ず授業を入れる制約と、授業の無い時限の次の時限には授業を入れないようにする制約条件

この他にも、複雑な時間割に対応するための制約条件をいくつか定式化した。表 1 は連続する時限に関する制約条件の定式化の可否をまとめたものである。

表 1. 時限の連続性の定式化の可否

<table>
<tr><th colspan="2">時限の連続性＼週あたりの時限数</th><th>2 限</th><th>3 限</th><th>4 限以上</th></tr>
<tr><td rowspan="3">連続しない</td><td>曜日が分散</td><td colspan="3">○</td></tr>
<tr><td>中間</td><td>／</td><td colspan="2">×</td></tr>
<tr><td>1 つの曜日に集中</td><td colspan="3">○</td></tr>
<tr><td colspan="2">部分的に連続</td><td>／</td><td>×</td><td>×</td></tr>
<tr><td colspan="2">完全に連続</td><td>○</td><td>○</td><td>×</td></tr>
</table>

定式化できたものは○
できていないものは×

3． 時間割作成問題の定式化の検証

2 章で定式化した制約条件が正しいか、数理計画ソフトを用いて検証した。処理の流れは図 1 である。

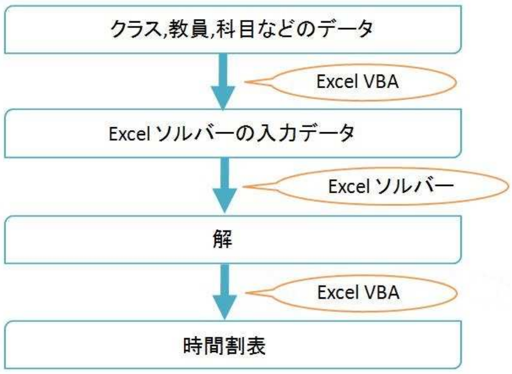

図 1. 処理の流れ

実際に数理計画問題を解くのは Excel ソルバーで行い、その Excel ソルバーの入力データを自動生成するプログラムを Excel VBA で作成した。得られた解から、表形式に成形された時間割表を作成するためにも Excel VBA を用いた。

その結果、規模の小さい時間割作成問題については、図 2 のように制約条件通りの時間割を出力することができた。

曜日1

時限＼クラス	クラス1	クラス2
1時限	科目1	科目3
2時限	科目2	科目3
3時限	科目2	科目3
4時限	科目3	科目1
5時限	科目3	科目2
6時限	科目3	科目2

図 2. 出力結果 (科目が連続する場合)

4． 分枝限定法の適用

3 章では、定式化した制約条件の正しさを検証したが、実際に数理計画問題を解く部分については考えていない。そこで、組合せ最適化問題の代表的な解き方である分枝限定法[1][2]を用いる方法について考察する。

4-1　分枝限定法とは

分枝限定法とは、問題を場合分けで分割していき、最適解を見つける方法である。

例えば、以下の問題を考える。

$$\begin{aligned} \text{max.}\quad & 23x_1+19x_2+16x_3 \\ \text{s.t.}\quad & 5x_1+3x_2+2x_3 \leqq 6 \\ & x_1, x_2, x_3 \in \{0, 1\} \end{aligned}$$

図 3 のように、まず $x_1=1$, $x_1=0$ の場合の二つの問題（小問題）に分ける。次に、それぞれの場合を $x_2=1$, $x_2=0$ に分ける。最後に、それぞれの場合を $x_3=1$, $x_3=0$ に分ける。そのとき、制約条件に反することが無く目的関数が最大のものを選択することで最適解が得られる。ただし、最適解が得られないと分かった時点で下側の探索をやめる(図 3 黒丸)。最大の目的関数値は 35 と分かる。

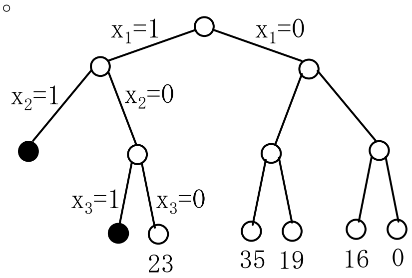

図 3. 小問題の探索

4-2　時間割問題への適用

例えば 2 クラス、2 科目、2 時限の場合で考える。まず図 4 のように、クラス 1 の 1 限目に科目 1 が入る場合と、入らない場合の二つの問題（小問題）に分ける。

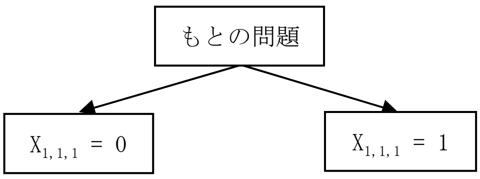

図 4. 時間割問題の小問題への分割

さらに図 5 のように、各小問題について、クラス 1 の 2 限目に科目 1 が入る場合と、入らない場合の二つの問題に分ける。このように全変数が埋まるまで繰り返す。その過程で、制約条件に反する小問題ができた場合、それ以降の探索は必要がないため探索を止める。この例では、クラス 1 の 1 限目に科目 1 が入っているとき、クラス 1 の 2 限目に科目 1 は入れることができない。つまり、$x_{1,1,1} = 1$ のときの $x_{1,1,2} = 1$ は制約条件に反するため、これ以降は探索しない。この方法を繰り返し、最適解を求める。

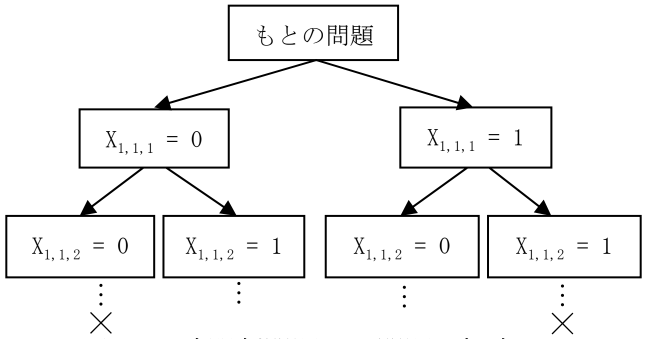

図 5. 時間割問題の小問題の探索

4-3　本解法の検証

4-2 で示した解法について、以下の点の検証を行う。

1. <u>この方法でどの規模まで解くことが可能か</u>
2. <u>どんな探索順序が有効か</u>
3. <u>時間割問題に特化した有効な探索方法の工夫</u>

例 1:　作成途中の時間割において、この先、制約条件に反しそうな状況を感知する指標を定義し、その指標の低いものから探索する。

例 2:　図 5 の例では時限を変化させつつ、探索しているが、実際はどの変数から値を決めていくのが効率が良いかを、実験的に調べる。

5．まとめ

簡略化した時間割を例として、様々な制約条件を定式化し、時間割を自動生成することに成功した。これによって、数理計画法によって時間割を自動生成することが可能なことが分かった。さらに、組合せ最適化問題の代表的な解き方である分枝限定法の適用を試みた。課題は以下の通りである。

・時間割作成に必要となる制約式の種類を増やす
・制約条件をより効率良く定式化する。
・時間割作成問題に特化した解法を探る。

参考文献

[1] 松井泰子・根本俊男・宇野毅明『入門オペレーションズ・リサーチ』，東海大学出版会，2008 年.

[2] 久保幹雄『組み合わせ最適化とアルゴリズム』，共立出版株式会社，2000 年.

Algorithm of Class-scheduling Using Mathematical Programming

†Ryu Uemura・Kinki University Technical College, Advanced Course, Department of Manufacturing System Engineering, Electric and Electronic Engineering Course

‡Takeshi Kanda ・ Kinki University Technical College, Department of Total System Engineering, Information and Communication Course