

CESPによる大学講義時間割作成問題の解決

6T-1

鈴木陽史, 高橋文博, 渋谷昇, 野田健一

拓殖大学工学部

1. はじめに

大学での講義時間割表の作成問題は、様々な制約条件を持つ講義を、その条件に従いながら時間割表に配置していく、組み合わせ問題である。従来の研究では、グラフ理論の一種である局所最良優先探索法を用いたもの[1][2]や、複数の例題から獲得した知識を実戦で活用するもの[3]などが見られた。両者とも、確実に正解が出ると仮定した上で配置作業を行い、重複配置発生時の回避対策に重点が置かれた。

本研究では、配置作業部分に論理型言語CESPを活用し、初期配置における「優先順位決定機能」「重複検索機能」「制約違反検索機能」を設け、「評価条件」による解答選択と、未解決時の条件変更提案機能を有する問題解決法を提案をする。

2. 制約条件の扱い

時間割表に配置する講義には、様々な制約条件が付加している。本研究では、時間割を構成する上で最低限必要な、時間割構成条件「講師は同一時間帯に複数の講義が出来ない。」と、得られた解答の評価に重点を置くことにより着目した、考慮することが望ましい条件「2時限以上の講義は連続して設置するのが望ましい。など。」を挙げ、これらを配置時点に盛り込む「補助的条件」と、解答の評価に使用する「評価条件」とに分類して扱う。

3. 問題解決手法

Solving Timetable Scheduling Problem for University Using Common ESP

Yoji Suzuki, Takehiro Takahashi, Noboru Schibuya, Kenichi Noda Faculty of Engineering, Takushoku University, 815-1 Tate-machi, Hachioji-shi, Tokyo 193, Japan

3. 1 配置優先順位の決定

最初に配置優先順位を決める。講義配置は、一講義ずつその配置可能範囲の一つを選んで行なう為、配置可能範囲の限定されている講義を優先する。

3. 2 制約違反検索を含む配置操作

優先順位に従って初期データを参照した配置を行う。初期データとは各講義の配置可能位置を示したものである。配置には「補助的条件」を考慮する。配置時に考慮する作業を「制約違反検索」と呼ぶ。

3. 3 重複検索

配置が完了したら、重複箇所が存在しないかどうかを学年内、学年間について検索する。検索の際に重複点数を数え、ある一定の値（重複条件）以内の解答であれば、評価対象として一時的に記録する。

3. 4 解答評価

重複点が存在しない解答が複数得られた場合には、「評価条件」を用いて解答の採点を行う。そしてその得点によって最適解を選択する。

3. 5 条件変更提案

重複点=0の解答が無い場合は、初期データ変更を促す条件変更提案を行う。これについては、(1)出講可能時限の拡張、(2)出講可能時限のスライド、(3)出講可能曜日の移動、の3種類を提案事項とした。

4. 手動配置による例題の解決

先に述べた処理手順を、曜日数3×時限数3、2学年分の時間割問題に適用し、手動配置による解決を行った。作成対象となる時間割の規模と、そこに配置する講義の初期データを表1にまとめて記す。

解答は、1講義の配置や置き換えを1手として72手で全ての有効解を検索できた。「補助的条件」には「2時限連続講義は連続配置する。」、「評価条件」には「講義間空時間」「講師の講義連続」「1

| 学年1 | | 出講可能曜日(時限) | | | | | | | | | | |
|---------|----|------------|----|----|----|----|----|----|----|----|----|--|
| 科目名 | 講師 | コマ | A1 | A2 | A3 | B1 | B2 | B3 | C1 | C2 | C3 | |
| 回路理論 | 伊藤 | 2 | ○ | ○ | ○ | | | | ○ | ○ | ○ | |
| 電子回路 | 野田 | 2 | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | |
| プログラム序論 | 植村 | 1 | | | | | ○ | ○ | | | | |
| 情報工学 | 幹 | 1 | | | | | | | | | ○ | |
| 医用情報 | 伊藤 | 1 | ○ | ○ | ○ | | | | ○ | ○ | ○ | |
| 論理回路 | 渋谷 | 1 | | | | | | | | ○ | ○ | |

| 学年2 | | 出講可能曜日(時限) | | | | | | | | | | |
|---------|----|------------|----|----|----|----|----|----|----|----|----|--|
| 科目名 | 講師 | コマ | A1 | A2 | A3 | B1 | B2 | B3 | C1 | C2 | C3 | |
| 画像工学 | 野田 | 1 | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | |
| 制御工学 | 小畑 | 1 | | | ○ | | | | | | | |
| 論理設計 | 渋谷 | 1 | | | | | | | ○ | ○ | ○ | |
| 周辺機器 | 大岡 | 1 | ○ | ○ | ○ | | | | | | | |
| プログラム演習 | 鎌角 | 2 | ○ | ○ | ○ | ○ | ○ | ○ | | | | |
| 情報実験 | 小泉 | 2 | | | | ○ | ○ | ○ | ○ | ○ | ○ | |

表1 初期データ

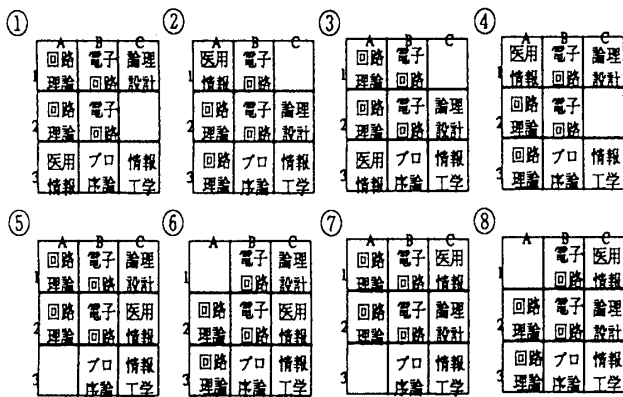


図1 手動配置による学年1の結果

時限目の講義数」の3条件を使用した。図1に手動配置による学年1の結果を示す。

5. GESpによる例題の解決

CommonESPは、ICOTで開発されたオブジェクト指向環境を持つPrologである。CESPは、Prologには無いbind_hook述語や他言語インターフェースなどの機能を持つことから、本問題解決に適用した。

配置操作は、初期データと配置場所を示す未定義変数の書式を「講義名(出講可能位置)」と決め、出講可能位置を数値コード化し、ユニフィケーションとバックトラックを活用して行なう。制約違反検索と学年間重複検索はbind_hook述語を用い、全解検索を行なうバックトラックによる無駄を削減する。

得られた解答は、講義順配置位置リストとして表現されている為、これを時間割順にソートしてから解答評価を施す。解答には3種類の評価条件による採点を行なう。評価条件は、(5)講義間空き時間が全

く無ければ表ごとに+1。(6)講師の講義時間に空きが全く無ければ1人につき+2。(7)1時限目の講義が無ければ表ごとに+1。という評点にし、得点の大きいものを最適解と判断し、画面に時間割表として出力する。これを先に挙げた例題に適用し、初期データによる解を60組、その中から最適解を4組(各学年2通り)を選択した。その結果を図2に示す。



図2 実行結果

6. まとめ・今後の課題

本研究は、時間割作成問題をCESPの持つ特徴とともに、配置優先順位決定、違反検索、重複検索の機能を持ち、なおかつ解が存在しない場合の条件変更提案機能を持つシステム構築を試み、これを条件を絞った試験的な課題に適用してその有効性を確認した。今後の課題として、①各種条件の増設、②解答評価における評価値の扱い、③より大規模な問題への活用、④条件変更提案のプログラム化と再配置操作導入、⑤エキスパートシステム活用、がある。

謝辞

本研究を進めるに当たり、CESPについて御指導下さいましたAI言語研究所松浦氏に感謝致します。

参考文献

[1]J.A.Ferland, et al.:comput.& ops.res., Vol.12, No.2, pp-207-218(1985)
 [2]藤原祥隆 他:信学論D-II, Vol.J76, No.3, pp.605-614(1993)
 [3]筒井茂義 他:,日本経営工学誌, Vol.41, No.5, (1990)