

大規模時間割作成システムの開発

宮原豊, 松本俊二

富士通株式会社) 計算科学研究センター

5 J-4

1 はじめに

時間割作成は一般に解の探索空間が広く、また制約条件も複雑なため、計算機による自動化が困難な問題である。[1]

時間割作成支援パッケージ等も商品化されているが、多くは割当率が高々90%である。仮に解修正用のエディタがあっても1000駒規模の時間割で生ずる未割当100駒を手修正するのは現実的だろうか。また、学校向け等ドメイン固有の制約条件が埋め込みのためユーザカスタマイズできなかつたり、あるいは新規制約が定義できても制約記述が複雑でエンドユーザにメンテナンスが困難であるなどの問題点も指摘されている。

本稿では、特殊法人自動車安全運転センター安全運転中央研修所殿向けに開発した時間割編成支援システムの事例を報告する。実データを用いて97%前後という高い自動割当率を達成するとともに、エンドユーザでも制約条件を容易に定義できるという特徴をもつ。

なお、本システムはUNIX WS上にES構築支援ツールKwESHELLを用いて実現された。

2 問題の特性とシステム要件

2.1 問題の特性

- ・大規模 … 1000駒以上の割当対象に、日付・時限・施設・教官を割り付ける。組合せ総数は 10^{2200} 以上。
- ・複雑な条件 … 実施順序や同一性等の制約が複雑。
- ・高い変更頻度 … 毎月変更、新設課程の制約追加等。

2.2 システム要件

- ・自動割当で高品質な解を導く。
- ・現実的な時間で求解(現状の人手1週間より短く)。
- ・エンドユーザにもメンテナンスできる制約表現。

A System Development of a Large-scale Timetable Scheduling

Yutaka Miyahara, Shunji Matsumoto

FUJITSU LIMITED

Research Center for Computational Science

1-9-3 Nakase, Mihama-ku, Chiba, Chiba 261, Japan

3 問題の定式化

3.1 駒の定義

時間割を表現する基本的なデータ構造を、組合せ要素のリストとして定式化し、これを駒と名付ける。駒は変数部と選択肢部の2つの部分から構成される。

課程	実理	段階	研修項目	日付	時限	施設	教官
変数部 (一部抜粋)				選択肢部			

変数部は時間割作成に先だつて決まっている要素の組合せであり、割当対象に相当する。

課程	実理	段階	研修項目	…
青少年四輪1日	実技	基本	運行前点検 基本走行	…

選択肢部は時間割作成中に決まる要素の組合せであり、変数部に割り当てられる資源に相当する(日付は課程の開始日と実施日数に基づき自動的に展開される)。

日付	時限	施設	教官
-	1~8	高速周回路…	実技教官

このように定式化すると、時間割作成とは、各駒に一意に与えられた変数部に対して制約条件を満足する選択肢を一つ決めることになる。

3.2 制約条件の定義

変数部と選択肢部の組合せの制約条件を列挙する。

- 1) 種別 … 個数、位置、隣接等9種の制約種。各種特殊の方式で2,3を解釈する制約判定プログラムに対応。
- 2) パターン部 … 制約判定したい駒と照合する要素列。
- 3) 比較演算子と制約値 … 制約種により解釈が異なる。

次の制約定義例は、「課程Aの高速周回は3日目以降、なるべく3日目に近く実施する」ことを意味する。

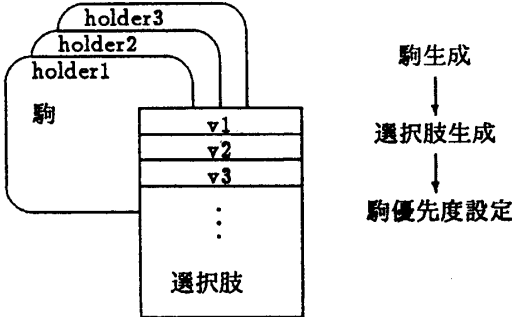
種別	パターン部				比較	制約値
	課程	研修項目	日付	…		
位置	A	高速周回	?	…	≥	3日目

制約判定は、上の形式の制約条件を各制約判定プログラムが解釈・実行する方式をとっている。このように制約記述が単純で形式的であるため、エンドユーザでもテーブル形式で容易に制約条件を定義できる。

4 自動編成

自動編成は、初期化→自動割当の順に行なう。

4.1 初期化

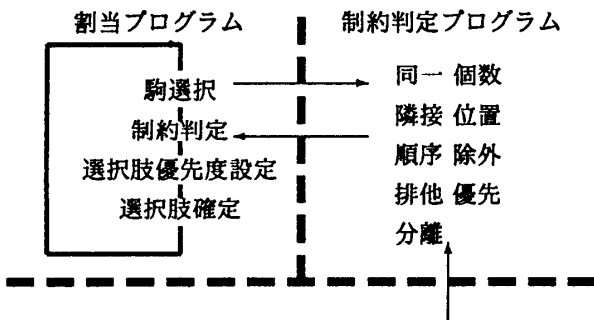


- 1) 駒生成 … 業務定義に基づき割当対象の駒を生成。
- 2) 選択肢生成 … 駒の変数部が取り得る選択肢を生成。
- 3) 駒優先度設定 … 駒割当順序を決める優先度を設定。
駒の優先度は次の基準から計算している。

- ・ 選択肢の個数の少ない駒を優先。
- ・ パターン照合する制約条件の多い駒を優先。
- ・ ユーザにより優先指定された駒を優先。

これは、選択肢が少なく制約の厳しい駒は割当の難しい駒と判定して序盤で割り当てることを意味する。

4.2 自動割当



制約条件

種別	パターン部						比較	制約値
個数	*	-	-	*	*	...	≤	1
隣接	-	-	実技	*	●	...	-	-
...

次のサイクルを割当対象駒がなくなるまで繰り返す。

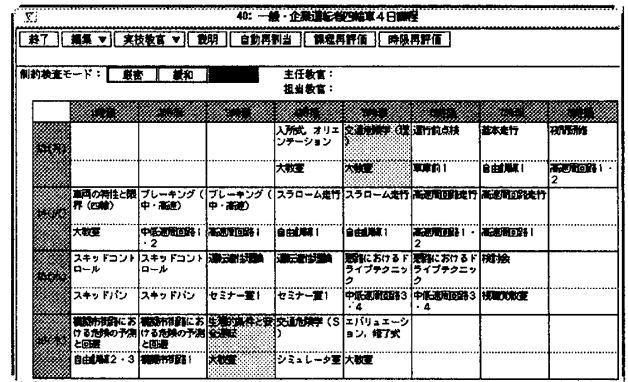
- 1) 駒選択 … 優先度の最も高い駒を取り出す。
- 2) 制約判定 … 制約判定プログラムを順次呼び出して制約判定をする。制約違反の選択肢は棄却する。
- 3) 選択肢優先度設定 … 棄却されずに残った各選択肢に、制約充足度に応じて優先度をつける。
- 4) 選択肢確定 … 優先度の最も高い選択肢を確定する。

ところで、同一や隣接等、複数駒の相互作用で制約違反が発生するような条件については、単一駒の制約判定だけで割当を決めてしまうと、後から割り当てる駒の制約違反を回避できなくなる可能性が高い。そこで、相互作用に与る複数駒をまとめて制約判定して可能な配置パターンを先読みすることで、割当率向上を図っている。

また、自動割当部は独立した三つのモジュールに分かれているため、少ない修正量でプログラムの機能を変更できる。業務変更で制約種が増えた場合でも、新規制約判定プログラムと制約条件定義の追加だけで済む。

5 解修正

自動編成で割当に失敗した駒を手修正するためのインタフェースとして解修正エディタを用意した。マウス操作による駒の移動、変更や制約違反状況の色分け表示ができる。



6 実施例

97%前後という高い割当率を達成した。自動編成には60MHz SuperSparc+ 搭載 WSで約1日間要する。

対象年月	クラス数	コマ数	割当数	割当率%
1995年1月	44	976	955	97.8
1995年2月	53	1121	1083	96.6

7 むすび

高い求解品質、ユーザメンテナンス性、モジュラー性を備えた時間割作成システムの開発について報告した。

今後は、他事例への適用を通じて、本手法の一般性や機能変更の容易さについて検証、改良してゆきたい。

参考文献

- [1] 渡辺正信: "制約ベース推論とその応用", 第1回 OR セミナーテキスト, pp. 36-48, 1995.