

マルチ辞書探索による RCPSP/ τ の新しい解法について01507203 名古屋学芸大学
01204223 南山大学*堀尾 正典 HORIO Masanori
鈴木 敦夫 SUZUKI Atsuo

1. はじめに

一般に、一つのプロジェクトにおいて、各作業間の実施順序に先行関係があり、作業の実施に有限のリソースが必要になる時に、各作業の実施時刻を決定する問題がリソース制約付きスケジューリング問題 (Resource Constrained Project Scheduling Problems, 以降 RCPSP と称する) となる [1, 2, 4]. 多くのスケジューリング問題は、各種制約をリソースとみなすことにより RCPSP の枠組みの中でとらえることができるため、いくつかの問題が RCPSP の枠組みで解かれたり、汎用的なソルバーのフレームワークとして採用されたりしている [1, 2].

我々は、作業の処理時間に応じて必要となる資源が変化する RCPSP/ τ モデル [1] を用いた汎用スケジューラ SEES (Scheduling Expert Engine System) を開発し [2], 能力の検証 (PSPLIB, 自作ベンチマーク問題による評価 [3]) と適用性の検証 (大学の時間割作成問題, 看護師の勤務スケジュール問題) を実施してきた [2, 3].

今回の発表では、これら検証の中で判明した RCPSP/ τ モデルの困難さに対応するため、性質の異なる二つの探索辞書を用いたヒューリスティクス探索を提案する。

2. RCPSP/ τ における SSGS

2.1 SSGS とは

広く利用されるスケジューリングスキームに Serial Scheduling Generation Scheme [5] (以降 SSGS と称する) がある。SSGS とは、未スケジューリングの中から一つずつ割付作業を選択し、制約が充足する最早時刻に作業をスケジューリングしていく方法である。リソースの使用可能量、消費が一定となる標準的な RCPSP では、SSGS を用いて有効スケジュールが得られることが Kolisch により証明されている [5]。さらに総所要時間最小化など正規完了時刻基準を有するスケジューリング問題においては、有効スケジュールで生成される解集合の中に最適解が存在することも知られている。このような SSGS であるが、RCPSP/ τ モデルでは、本スキームだけでは最適解を得ることができないことがわかった [2].

2.2 SSGS の誤差評価例

(1) 誤差の評価方法

我々は、RCPSP/ τ 用のベンチマーク問題を自作し、SEES を使って求めた SSGS の解と、汎用数理計画ソフト XpressMP を使用して求めた最適解の比較を行った。これにより、最適解に対して SSGS がどの程度の誤差を与えているかを評価した [3].

作成した問題は、すべて総所要時間最小を目的としたもので、スケジューリング作業数が 6 個, 12 個, 24 個の 3 タイプ, それぞれ 20 問ずつの計 60 問である。そのうち XpressMP が 24 時間以内に解を求めることができた 6 問題, 12 問題の 40 問に対して実験を実施した。

(2) 誤差の評価結果

評価結果は次の通りである (表 1 参照)。ただし使用した PC 環境は AMD Athron (1.1GHz), メモリ 384MB, OS は Windows XP pro. である。

表 1 SSGS が与える影響

	最適解の 合計	SSGS による結 果の合計	合計の相対誤 差 (%)
6 問題	645	664	2.9
12 問題	1180	1182	0.17
TOTAL	1825	1846	1.15

ただし最適解の合計とは、すべての問題の最適解の合計であり、合計の相対誤差 (%) とは、(SSGS による結果の合計-最適解の合計)/最適解の合計*100, を表している。この結果、今回作成した問題においては、全体の誤差に対して SSGS が原因となって生ずるものが、トータルで 1.15% 程度発生していると判断した。

3. 複数辞書を用いた新探索手法

我々の提案する探索手法ではメインとサブの二つの辞書が用いられる。

メイン辞書とは SSGS で用いられるもので、次の割付け作業の候補集合 (適合度作業集合) をリストとしている。

サブ辞書は、一つの作業を割付ける時点で、その作業のすべての割付け可能な時間帯の中で最小のもから順に最大 n 個をリストとしたものである (図 1 参照)。この二つの辞書を次のようなアルゴリズムで使い分ける。

Step1 : (原因作業の推定)

“失敗作業”を割付けた状態から、割付け作業を一つずつさかのぼって調査し、以下のルールに従う作業で最初に発見されたものを“原因作業”とする。

- ・“失敗作業”の直接の先行作業

もしくは、

- ・一つ以上のリソース制約を共有し、“失敗作業”の実施時間帯に実施時刻が含まれる作業

Step2 : (メイン辞書を用いた従来のバックトラック)

“原因作業”の割付け時点より以前に“失敗作業”の割付けが可能かどうか、メイン辞書を用いて調査する。可能ならば、割付け状態を“原因作業”割付ける以前の状態まで戻り、“失敗作業”を“原因作業”の先に割付け Step6 :へ。そうでなければ Step3 :へ

Step3 : (新バックトラックへの分岐判断)

次の二つのルールを同時に満たす場合、Step5 :へ。

そうでなければ Step4 :へ

- ・“原因作業”は“失敗作業”の直接の先行作業ではない
- ・“原因作業”の別の割付け候補時刻 T_k がサブ辞書内に存在する

Step4 : (新しい失敗作業の推定)

新しい“失敗作業”を推定して、Step1 :へ

Step5 : (サブ辞書を用いた新バックトラックの実行)

状態を“原因作業”の割付け直前の状態まで戻し、“原因作業”を“新時刻 T_k に割付け。

Step6 : (バックトラック終了)

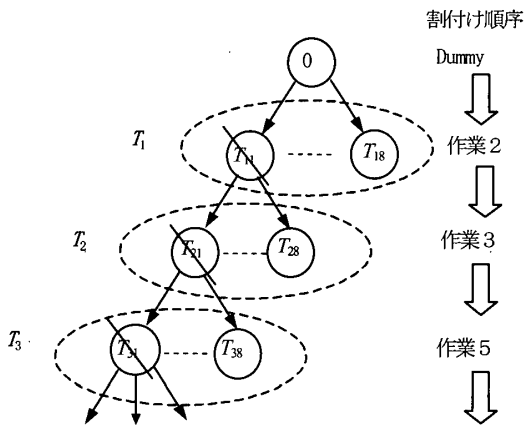


図1 サブ辞書の探索例

4. 新手法に対する実験結果

作成したベンチマーク問題の中の55問(6作業:20問, 12作業:20問, 24作業:15問)を用いて実験を行った。ただし、24作業は、XpressMPで24時間以内に解を得ることができた問題のみを用いている。この問題に対して、

SEESのメイン辞書だけを用いたものと、新アルゴリズム(メインとサブのマルチ辞書)を採用したもので探索を行った。その結果を表2に示す。

表2 新アルゴリズムを用いた結果と最適解との比較

	XpressMP	SEES (メイン辞書のみ)	SEES (マルチ辞書)
6 作業問題	645	672	645
12 作業問題	1180	1203	1180
24 作業問題	1400	1413	1400
TOTAL 誤差	-	1.95%	0%

このように、発見したすべての解が最適解に一致した。次に計算時間について述べる(表3参照)。

表3 合計実行時間の比較

	XpressMP	SEES (メイン辞書のみ)	SEES (マルチ辞書)
計算時間計	147956.8	161.65	1677.39

上記の表で示されるように、計算時間はメイン辞書だけを用いる従来型に比較して10.37倍必要となった。これは、サブ辞書を用いたことによるトレードオフと考えられる。しかし、最も計算時間を要する24問題でも平均108.7sec/1問で解を得ることができた点を考慮すれば、新手法の実用性は高いと判断できる。

参考文献

- [1] S. Harthman: *Project Scheduling under Limited Resources Models, Methods, and Applications* (Springer-Verlag Berlin, Heidelberg, 1999)
- [2] 堀尾正典, 鈴木敦夫: 時間制約のある RCPSP/ τ を用いた汎用スケジューラの開発. 日本経営工学会誌, 54-3 (2003), 203-213.
- [3] 堀尾正典, 鈴木敦夫: RCPSP/ τ モデルを用いた汎用スケジューラの定量的性能評価. 日本経営工学会平成15年度春季大会予稿集, D13 (2003), 166-167.
- [4] R. Klein: *Scheduling of Resource-Constrained Projects* (Kluwer Academic Publishers, Boston, 2000).
- [5] R. Kolish: Serial and parallel resource-constrained project scheduling methods revisited: Theory and computation. *European Journal of Operational Research*, 90 (1996), 320-333.