形状モデル CBR を用いたプロジェクト・スケジューリング解法*

大久保 雅史^{*1}, 山地 泰久^{*2}
井上 勝敬^{*1}, 福田 収一^{*3}

Project Scheduling Using Shape Model CBR

Masashi OKUBO, Yasuhisa YAMAJI,
Katsunori INOUE and Shuichi FUKUDA

The importance of project scheduling problems is growing rapidly with the increasing diversity of products. The most important requirements are flexibility and simplicity. However, most of the methods proposed up to now have difficulty in coping with such problems involving varying workload activities. Therefore, a new technique based on figure manipulation is proposed to meet these requirements. Our new technique utilizes a shape model called CBR (Coded Boundary Representation) proposed by S. Fukuda, which expresses a shape in the form of lists using directional codes. Figures can be easily composed or discomposed and their features can be extracted without any difficulty. Scheduling is solved as a packing problem of the rectangles representing activities. Our approach greatly simplifies the overall procedures, since they can be reduced to simple list operations, and can cope with the varying work load problem with considerable ease.

Key Words: Project Scheduling Problem, Packing Problem, Figure Manipulation, New Shape Model, Coded Boundary Representation

1. はじめに

最近の製品の急激な多様化によりプロジェクト・スケジューリングがこれまでにまして重要な問題となってきた。プロジェクト・スケジューリング問題の解決が容易でない理由は、解析的に最適解を求める一般的な手法が確立されていないこと、また制約条件を満たすすべての解を調べたのでは、その数が非常に多く実質的に不可能であることなどが挙げられる。このため適用される手法の多くは近似算法で、実時間内にある程度満足できる解を求めることを目標としている。PERT はこの近似算法の代表的な手法である。

ほとんどの近似算法は、プロジェクト全体で作業員や使用可能な機械の数が変化しない。一つの作業の中で必要とする作業員や機械の数は一定である等の前提を置いている。しかし、実際のスケジューリングにおいてはプロジェクトが大きくなれば使用可能な機械の数が作業員が変化したり、一つの作業の中で必要とされる機械の数が作業員が変化する、あるいはある作業

が終了すると同時に別の作業を開始しなければならないことは十分に考えられる。

そこで本論文では、著者の一人福田が提案した新しい形状モデル CBR を利用して図形処理的に解を求める方法を提案する。形状モデル CBR は、形状をリストで表現し、図形の合成・分解や特徴抽出操作を容易に行える。この形状モデルの特徴を利用し、プロジェクトを構成する作業を箱で表現し、スケジューリング問題を箱詰め問題に帰着させ解を求める。プロジェクト・スケジューリング問題を箱詰め問題に帰着させたため作業人数変更問題に対しても柔軟に対処できる。

2. 従来のプロジェクト・スケジューリング
問題の解法

これまでさまざまなプロジェクト・スケジューリング問題の解法が提案されてきた^{(1)~(5)}。ガント・チャートは比較的簡単な仕事のスケジュールに適し、非常に見やすい。しかし、作業間の詳細な相互関係が表現できない。コストなどの制約条件が表現されない。仕事の進行にともない現在までの進行が残された作業の日程にどの程度影響を与えるのかを直感的につかめないなどの欠点がある。そこで、それらの欠点を補うために有向グラフを用いて各作業の相互関係を明らかにす

* 原稿受付 平成6年3月22日。

^{*1} 正員、大阪大学溶接工学研究所 (〒567 茨木市美穂が丘11-1)。^{*2} シャープ (〒632 天理市樺本町2613-1)。^{*3} 正員、東京都立科学技術大学 (〒191 日野市旭ヶ丘6-6)。

る PERT, CPM が提案された。これらの手法は有向グラフを用いるネットワーク手法であるため、(1) 作業の分割の程度によってスケジュールをミクロにもマクロにも扱える。(2) 途中で進行が遅れても有効に対応できる。(3) 仕事の管理者にも担当者にも全体の進行が容易に理解できる。(4) 非常に大きな仕事も扱うことができる。(5) 部分と全体との関係が明りょうになる。などの利点をもつ。

また最近では、例えば、ATMS⁽⁶⁾ と呼ばれる手法を適用して探索空間を小さくして解を求める試みもなされている⁽⁷⁾。この手法は探索によって一度矛盾が発見されると、それまでの推論と発見された矛盾を記録しておき、以降の推論で随時それらを参照することによって効率よく矛盾回避を行う。しかし、この方法にも、推論や矛盾を記録するデータベース維持の時間、空間的なオーバーヘッドが大きいなどの欠点があり、問題空間の大きさにより、求解が実際上不可能な場合がある。

プロジェクト・スケジューリング問題に対して、このようにさまざまな解法が提案されているが、どの解法も一長一短があり新しい解法の開発が各分野において要望されている。

3. PERT 解法の問題点

PERT は山くずし法を用いれば労働力の平準化を計り、制約条件を満足できる利点がある。しかし、山くずし法は作業所要人数がその作業を通じて一定という制限がある。実際の作業では一連の作業内で使用できる機械の台数の制限の変化、あるいは使用できる所要人数の制限の変化などによって、その一つの作業に必要とされる人数が変化する可能性がある。

例えば、PERT で所要人数が変化する問題を解く場合、ある作業 A を各作業の人数がすべて一定になるように作業 A1, 作業 A2... に分割して別の作業であると考え、作業 A1 が終了すると同時に作業 A2, 作業 A3 と開始する方法を用いる。しかし、後続する作業 A2, 作業 A3 などが制約条件を満たさなければ、はじめの作業 A1 の開始時刻にさかのぼって見直す必要があり、山くずし法は非常に煩雑となる。また、PERT による山くずし法によって得られた解は必ずしも最適解とは一致しない。

4. 箱詰め問題としてのアプローチ

そこで、本研究では各作業を形状モデル CBR を用いて矩形としてリスト表現し、スケジューリング問題を箱詰め問題に帰着させ、図形処理による解法を採用して解法の簡便化と柔軟性の増大を図った。

CBR は Freeman の方向コード⁽⁸⁾ の応用であり、その方向コードは図 1 で、また基本形状プリミティブは図 2 で定義される。本研究では、横方向に所要日数、縦方向に所要人数を取る。図形のトポロジカルな特徴は方向コードにより容易に把握できる。形状をリスト表現しているため、合成、分解、鏡像、回転、縮退等さまざまな図形操作が簡単なりスト操作として処理可能である。スケジューリングでは主に図形の合成と分解および辺の縮退操作を取り扱う。形状モデル CBR とその操作法の詳細については文献(9), (10)を参照されたい。

スケジューリングを図形处理的に解くには、まず基準となるベースと呼ぶ辺 2 と辺 5 の幅が 0 の箱を設定する(図 3)。辺 1 と辺 3, 辺 4 と辺 6 (有向辺の右側の数字が辺の名前を、左側の数字が辺の長さを表わす)は横軸、縦軸に対応し、適当な長さとするばよい。横

図 1 形状モデル CBR の方向コード

図 2 基本形状プリミティブ

図 3 基準ベース

図 4 箱詰め操作

軸、縦軸が箱として表現されるので、各作業(箱)を箱詰めの形で図形の合成として処理できる。

すなわち軸に箱(作業)が接する場合も、各箱(作業)同士が接する場合も同様に図形の合成として処理できることになる。

箱詰めの際には、箱詰めした作業の下に余剰空間を作らない。箱詰めした作業の下に余剰空間が存在する場合[図 4 (a)], その作業を一度分解し[図 4 (b)], そのすぐ後に接するように合成して余剰空間を埋める[図 4 (c)]。箱詰め場所によっては、さらに分解の必要がある。

5. 形状モデル CBR を用いた 箱詰めアルゴリズム

以下に述べるアルゴリズムは最早計画および制約条件のある場合のスケジューリングに適用可能である(図 5)。

[処理 1] 作業リストから優先順位を決定し、処理 2 へ進む。優先順位は、現在注目している作業から最終作業に至るまでの各作業の所要日数の総和を計算して最早終了日数とし、最早終了日数の大きいものを優先する方法を取る。ただし、後続する作業が複数である場合にはその工程すべてを探索し、最大値をその作業の最早終了日数とする。また、先行作業をもつ作業の工程は先行作業をもたない作業の工程の中に含まれることから、それを参照することによって求められ最早終了日数を計算する必要はない。また、優先順位を決定する際に最早終了日数が同じ場合には所要日数の短いものを優先し、所要日数も同じであれば、所要人数の少ないものを優先する。

[処理 2] ベースおよび各作業のリストを作成し、処理 3 へ進む。ベースの辺長リストの各要素は仕事の大きさに応じて設定する。

[処理 3] 優先順位が 1 の作業をベースに箱詰めし、その作業の終了時刻を保存しておき、処理 4 へ進む。優先順位が 1 の作業をベースに箱詰めする場合、ベースの方向リストは必ず $[e, n, w, n, w, s]$ であるから、3 番目の要素 w に該当するベースの終点リス

トの要素に、詰める作業の方向リストの要素 e に該当する始点リストの要素を合致するように箱詰めして新たにベースとする。そのとき、新たなベースの 3 番目の辺長リストの要素をその作業の終了時刻として保存する。

[処理 4] 優先順位にしたがって作業を取り出し、先行作業を調べて存在しなければ処理 5 へ、存在すれば処理 6 へ進む。

[処理 5] 頂点候補のうち候補時間が最大値のものを選択して箱詰めし、終了時刻を保存しておく。作成した頂点候補をすべて消去し、処理 4 へ進む。頂点候補とは作業を詰めることができるベースの頂点リストの要素であり、次のようなリストの操作によって作成する。まず、箱詰め前のベースの方向リストの最後の要素 w を除いた要素 w を順に取り出し、それに該当する要素を箱詰め前のベースの終点リストと辺長リストから取り出す。次に、辺長リストから取り出した要素は、それ以前に取り出した辺長リストの要素の総和(候補時間)を求め、先の終点リストの要素と一組にして頂点候補として保存しておく。箱詰めは、詰める作業の方向リストの要素 e に該当する始点リストの要素が頂点候補に合致するように行い、箱詰め後を新たにベースとする。

[処理 6] 先行作業が一つであれば処理 7 へ、複数であれば処理 8 へ進む。

[処理 7] 頂点候補のうち先行作業の終了時刻に合致する候補時間を選択してその作業を箱詰めし、その作業の終了時刻を保存しておく。作成した頂点候補をすべて消去し、処理 8 へ進む。処理 5 で用いた方法によって頂点候補を作成し、先行作業の終了時刻に合致する候補時間をもつ候補を取り出し、詰める作業の方向リストの要素 e に該当する始点リストの要素をその候補に箱詰めして新たにベースとする。

[処理 8] 各先行作業の終了時刻のうち最小値を選択してそれと合致する頂点候補に箱詰めし、終了時刻を保存しておく。作成した頂点候補をすべて消去し、処理 9 へ進む。複数ある先行作業の終了時刻の中から最小のものを選び、その終了時刻と処理 5 にある方法

図 5 形状モデル CBR を用いた箱詰アルゴリズム

で作成した頂点候補の候補時間が合致するものを見つけだし、その点に詰める作業の方向リストの要素 e に該当する始点リストの要素を箱詰めし、新たにベースとする。その作業の終了時刻は、処理 5 にある方法と同様に箱詰めされた候補時間からその作業の所要日数の差を取る。

〔処理 9〕残っている作業があれば、処理 4 に戻り、残っていない場合は一連の処理を終了する。

作業人数変更問題に対するアルゴリズムは前項と基本的には同じであるが、制約条件を満足するように箱

詰めする処理操作は処理 5 以降の操作において若干異なる。

所要人数が変化しない場合、方向リストの要素 e は一つしかないので上述のアルゴリズムの処理 5 では、それに該当する始点リストの要素をベースの候補に合致するように箱詰めを行った。しかし、所要人数が変化する作業は、方向リストの中に要素 e が複数存在する。この場合、それぞれの要素 e に該当する始点リストを順に取り出し、制約条件を満たしてできるだけ前に作業を詰めるように、その作業の始点リストの要素

と頂点候補を選択し、その要素と頂点候補が合致するように箱詰めを行う。ただし、先行作業を超えないように箱詰めをする必要がある。このとき、その作業の終了時刻は選択された始点リストの要素が第1番めの要素から数えて何番めにあるかを確認し、それに該当する方向リストの要素を選ぶ。もし、その要素の後ろに要素 e があるなら、その選択した要素を含めて後ろのすべての要素 e に該当する辺長リストの和を計算

し、箱詰めした候補とその値の差を取ってその作業の終了時刻とする。

6. 方法論の検討

方法論の有効性をいくつかの例題により検討した。

表 2 作業リスト

Activity name	Predecessor name	Necessary days	Necessary workers	Priority
A	—	5	2	2
B	—	3	3	1
C	A	4	1	4
D	A	3	2	6
E	B	4	1	3
F	C	3	5	7
G	D	2	3	8
H	E	5	3	5
I	F, G	3	2	10
J	H	4	2	9
K	I, J	4	3	11

表 1 作業リスト

Activity name	Predecessor name	Necessary days	Necessary workers
A	—	3	4
B	—	5	3
C	B	4	1
D	A	4	2
E	A	7	3
F	C, D	5	3
G	E, F	4	5

図 6 PERT による山くずし法

(a) 本手法

(b) PERT

図 7 表 2 に対する本手法と PERT の比較

表 3 作業数, 最早計画による総所要日数の仕事例

Project	1	2	3	4	5	6
Activity number	9	12	12	13	15	19
Earliest scheduling time	30	19	25	23	23	28

表 4 表 3 に対する本手法と PERT の比較

Project		1	2	3	4	5	6
Constraint on 5workers	New method	39	30	29	33	32	40
	PERT	39	30	29	33	33	44
Constraint on 6workers	New method	39	25	29	30	26	35
	PERT	39	25	29	30	28	33

使用計算機は Sun Sparc II, プログラミング言語は Prolog (IF/Prolog)である。

表 1 の作業リストについて図形処理的アプローチと PERT による山くずし法(図 6)を比較すると, 総所要日数に変わりがない。また表 2 の作業リストから人数制限が 5 人の制約条件のもとで本手法および PERT による山くずし法を用いて得られた結果を図 7 に示す。さらに表 3 の例の, 人数制限 5 人, 6 人の制約条件のもとでの本手法と PERT による結果を対比して示す(表 4)。

これらの結果より, 本論文で提案した新手法と PERT では総所要日数に大きな違いが認められず, 本手法はスケジューリング問題に対して有効であると考えられる。

ところで, 本手法による結果が PERT と同じく最適解であるという保証はない。本手法は前述のようにまず優先順位を決定し, それにしたがって箱詰めを行っているため, その結果は優先順位に大きく依存する。したがって, 本手法の優先順位の決定法が必ずしも最優とはいえず, この点に関しては PERT と同じである。

しかし, 作業の途中で制約条件の人数に変化があるスケジューリングの場合, PERT では再び山くずし計算を行う必要がある。これに対して, 本手法では図形処理であるため, 箱(作業)を詰め直すだけでよく, 簡単なリスト操作によって解決できる。また, 一連の作業が中断される場合も, 図形の分解操作によって作業を分解して詰めることで簡単に解を求められる。

7. ま と め

本研究では, プロジェクト・スケジューリング問題が箱詰め問題に帰着させられる点に注目し, 福田が提

案した形状モデル CBR を用いて図形処理により解を求める方法を提案し, その有効性の検討を行った。本研究から得られた結論は,

(1) 形状モデル CBR はリスト表現であるため, 各種の図形処理を簡単なリスト操作により実行できる。したがって, プロジェクト・スケジューリング問題を箱詰め問題として容易に処理できる。

(2) PERT では求解が困難な作業人数変更問題に対しても図形処理により容易に解を求めることができる。

(3) 制約条件である人数制限が途中で変更される場合でも, 本研究で提案したアルゴリズムでは制約条件のデータのみを変更するだけで, PERT のように計算をはじめからやり直す必要はない。

(4) 途中で中断する一連の作業がある場合, その作業を表現した図形をリスト操作によって作業を二つの図形に分ければ, 本アルゴリズムにしたがって箱詰めを行って解を求めることが可能となる。

しかし, 本解法には以下に述べるような問題点がある。

(1) 本手法では優先順位を決定してそれにしたがって箱詰めを行っている。したがって, 解が優先順位に大きく依存する。そのため, 解が最適解である保障はない。

(2) 本手法は有向グラフを用いる方法と比較して, 各作業間の関係把握が容易ではない。

これら問題点に対しては, リスト操作の特徴を活かしたより柔軟な解法とする必要があるが, 今後本アルゴリズムの改良により十分に対処可能と考える。

文 献

- (1) 関根, PERT・CPM 入門, (1989), 日科技連。
- (2) 寺野, システム工学入門, (1985), 230, 共立出版。
- (3) 通商産業省産業構造審議会管理部会, 工程管理, (1989), 190, 日刊工業新聞社。
- (4) 工藤, 矩形表現された生産計画における対話的変更機構, 情報処理学会第 41 回全国大会, (1989), 243。
- (5) 工藤, 負荷平準化のための制約指向インターフェイス, 人工知能学会全国大会 (第 4 回), (1990), 445。
- (6) de Kleer, J., An Assumption-based TMS, *Artificial Intelligence*, 28 (1986), 127。
- (7) 湯上伸弘, 原裕貴, スケジューリング問題の解法の検討, 人工知能学会全国大会 (第 4 回), (1990), 527。
- (8) Freeman, H., *Computer Processing of Line-drawing Images, Computer Surveys*, 6-1 (1974), 54。
- (9) 福田, 信頼性設計エキスパートシステム-形態の処理とその応用-, (1990), 丸善。
- (10) 福田, 新しい形状モデル CBR 法の開発, 機論, 57-541, A (1991), 31。