

動的再構成可能なシステムのための 計算資源割り当てスケジューリング手法

小平 行秀[†] 児玉 親亮^{††} 藤吉 邦洋^{††} 高橋 篤司[†]

[†] 東京工業大学大学院 理工学研究科 集積システム専攻 〒152-8552 東京都目黒区大岡山 2-12-1-S3-58

^{††} 東京農工大学大学院 工学教育部 電子情報工学専攻 〒184-8588 東京都小金井市中町 2-24-16

E-mail: †{kohira.atsushi}@lab.ss.titech.ac.jp, ††{kodamada,kfujiyos}@fjlab.ei.tuat.ac.jp

あらまし 2次元平面上に計算資源が配置された動的に再構成可能なシステムにおいて、全体タスクが少ない計算資源で短時間に完了するよう部分タスクを矩形形状の計算資源にある期間割り当てる問題は、計算資源を2次元平面とし時間軸を第3軸とする3次元空間に、部分タスクに対応する直方体を互いに重なることなく埋め込む直方体パッキング問題と見立てることができる。しかし、部分タスクは独立した処理ではなく処理順序に一定の制約が課されるため、その制約を満足するようなパッキングを求めなければならない。処理順序の制約を考慮した直方体パッキング問題を3次元スケジューリング問題と呼ぶ。様々な直方体パッキングの表現方法が提案されているが、どの表現が3次元スケジューリング問題に有効であるかはこれまで検討されてこなかった。そこで本稿では、容易に処理順序制約を考慮できる3次元パッキング表現手法 Sequence-Quadruple を提案するとともに、どの表現が3次元スケジューリング問題に対して有効であるかを実験的に示す。

キーワード 動的再構成可能なシステム, 3次元スケジューリング, 直方体パッキング, Sequence-Quadruple

3D-Floorplanning for Scheduling of Dynamically Reconfigurable Systems

Yukihide KOHIRA[†], Chikaaki KODAMA^{††}, Kunihiro FUJIYOSHI^{††}, and Atsushi TAKAHASHI[†]

[†] Department of Communications and Integrated Systems, Tokyo Institute of Technology
2-12-1-S3-58 Ookayama, Meguro-ku, Tokyo, 152-8552 Japan

^{††} Department of Electrical and Information Engineering, Tokyo University of Agriculture and Technology
2-24-16 Nakacho, Koganei-shi, Tokyo 184-8588 Japan

E-mail: †{kohira.atsushi}@lab.ss.titech.ac.jp, ††{kodamada,kfujiyos}@fjlab.ei.tuat.ac.jp

Abstract In a dynamically reconfigurable system, computation resources are arranged in 2D-plane and each partial task is assigned to computation resources of rectangle-shape for a certain time period. The problem can be regarded as the rectangular box packing problem in 3D-space of 2D-plane and time axis. However, since partial tasks have order constraints, a packing should satisfy the given order constraints. We call this problem 3D-scheduling problem. Although there have been proposed various types of rectangular box packing representation, it was not examined until now which representation is fit to a 3D-scheduling problem. In this paper, we propose a new representation of rectangular box packing called Sequence-Quadruple which can handle order constraints easily, and show experimentally that which representation is fit to a 3D-scheduling problem.

Key words reconfigurable system, 3D-scheduling, rectangular box packing, Sequence-Quadruple

1. はじめに

2次元平面上に計算資源が配置された動的に再構成可能なシステムにおいて、全体タスクが少ない計算資源で短時間に完了するよう部分タスクを矩形形状の計算資源にある期間割り当てる問題は、計算資源を2次元平面とし時間軸を第3軸とする3次

元空間に、部分タスクに対応する直方体を互いに重なることなく埋め込む直方体パッキング問題と見立てることができる。しかし、部分タスクは独立した処理ではなく処理順序に一定の制約が課されるため、その制約を満足するようなパッキングを求めなければならない。我々は、処理順序の制約を考慮した直方体パッキング問題を3次元スケジューリング問題と呼ぶことと

する。

直方体パッキング問題は、2次元パッキング問題がそうであることから分かるように、NP 困難である。従って、3次元スケジューリング問題に対しては、処理順序制約を考慮しながら、直方体パッキング表現手法と確率的探索手法とを組み合わせることで準最適解を求めるアプローチが考えられる。

これまで提案された直方体パッキング表現手法は、順列による表現、グラフによる表現、木構造による表現に大別される。順列による表現には、2次元パッキングを表現する Sequence-Pair [1] を3次元に拡張した、山崎らによる直方体名の順列を3つ用いることで直方体の相対位置関係を表現する Sequence-Triple [2] や、同じく順列を5つ用いて表現する Sequence-Quintuple [2] がある。グラフによる表現には2次元パッキング手法の TCG (Transitive Closure Graph) を3次元に拡張した 3D-subTCG [3] が提案されている。木構造による表現には、2次元パッキング手法の O-Tree [4] を直方体パッキングへ拡張した川井らの手法 [5] がある。また、2次元パッキング表現として有名なスライス木を3次元に拡張した手法 [6] が最近提案されている。

これらの手法は、どんな3次元パッキングでも表現可能であるが良いパッキング結果を得にくかったり、良いパッキング結果を得やすいが表現できないパッキングが存在したりと、一長一短である。また、どの表現が3次元スケジューリング問題に有効であるかはこれまで検討されてこなかった。

そこで本稿では、容易に処理順序制約を考慮できる3次元パッキング表現手法 Sequence-Quadruple を提案するとともに、どの表現が3次元スケジューリング問題に対して有効であるかを実験的に示す。

2. 直方体パッキング問題と既存手法

直方体パッキング問題とは、直方体の集合 M が与えられたとき、各直方体を互いに重なることなく最も小さい体積の直方体内に配置せよ、という問題である。説明の都合上、 x 軸の増加 (減少) 方向を右 (左)、 y 軸の増加 (減少) 方向を奥または後 (手前または前)、 z 軸の増加 (減少) 方向を上 (下) と呼ぶ。また、直方体 a の右端の x 座標が直方体 b の左端の x 座標と同じか小さいとき、直方体 a は直方体 b の左にあると呼ぶ。右、前後、上下の位置関係に関しても同様に定義する。

2.1 Sequence-Pair

Sequence-Pair (SP) [1] は、矩形集合 M を入力とする2次元パッキング問題において、長さ $|M|$ の矩形名の順列 Γ_+, Γ_- の対 (Γ_+, Γ_-) により2次元パッキングを表現する。

$\Gamma_+^{-1}(a)$ は Γ_+ の中で矩形 a が何番目かを指す関数とする。 Γ_- についても同様に定義する。任意の矩形対 a, b の相対位置関係の制約は以下のように与えられる。

左右関係: $\Gamma_+^{-1}(a) < \Gamma_+^{-1}(b)$ かつ $\Gamma_-^{-1}(a) < \Gamma_-^{-1}(b)$ のとき、 a は b の左。

上下関係: $\Gamma_+^{-1}(a) < \Gamma_+^{-1}(b)$ かつ $\Gamma_-^{-1}(a) > \Gamma_-^{-1}(b)$ のとき、 a は b の上。

(Γ_+, Γ_-) により、矩形対には上下左右のいずれか1つの相対位置関係が与えられる。順列対から2次元パッキングを得るために、矩形対に与えられた相対位置関係の制約から、それぞれ矩形の幅、長さを点重みとした左右制約グラフ G_x 、上下制約

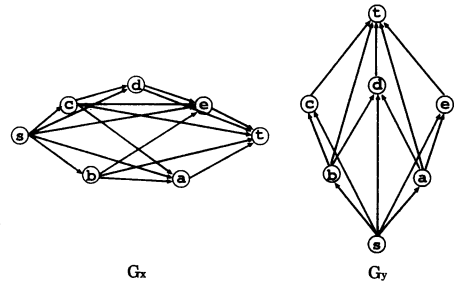


図1 $(\Gamma_+, \Gamma_-) = (cdbea, bcade)$ の制約グラフ。

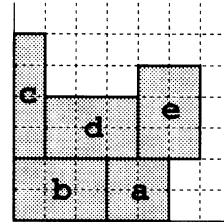


図2 $(\Gamma_+, \Gamma_-) = (cdbea, bcade)$ に対応するパッキング。

グラフ G_y を構築する (図1)。この G_x, G_y 上でソース点からの最長パス長を計算し、各矩形における最長パス長をそれぞれ x 座標、 y 座標とすることにより、相対位置関係を制約とする面積最小解を求めることができる (図2)。

与えられた SP に対応する2次元パッキングが必ず存在し、与えられた2次元パッキングに対応する SP も必ず存在することが知られている [1]。

2.2 Sequence-Triple

Sequence-Triple (ST) [2] は、2次元パッキング表現手法 SP を3次元へ拡張した手法で、3つの順列の組 $(\Gamma_1, \Gamma_2, \Gamma_3)$ により直方体対の相対位置関係の制約を与え、3次元パッキングを表現する手法である。任意の直方体対 a, b の相対位置関係の制約は以下のように与えられる。

左右関係: $\Gamma_2^{-1}(a) > \Gamma_2^{-1}(b)$ かつ $\Gamma_3^{-1}(a) < \Gamma_3^{-1}(b)$ のとき、 a は b の左。

前後関係: $\Gamma_1^{-1}(a) < \Gamma_1^{-1}(b)$ かつ $\Gamma_2^{-1}(a) < \Gamma_2^{-1}(b)$ かつ $\Gamma_3^{-1}(a) < \Gamma_3^{-1}(b)$ のとき、 a は b の手前。

上下関係: $\Gamma_1^{-1}(a) > \Gamma_1^{-1}(b)$ かつ $\Gamma_2^{-1}(a) < \Gamma_2^{-1}(b)$ かつ $\Gamma_3^{-1}(a) < \Gamma_3^{-1}(b)$ のとき、 a は b の下。

SP と同様、各相対位置関係の制約に基づいて x, y, z 軸それぞれの制約グラフを作成し、最長パス長を計算することにより各直方体の座標値を求めると、3次元パッキングが得られる。

与えられた ST に対応する3次元パッキングは必ず存在するが、与えられた3次元パッキングに対応する ST が必ず存在するとは限らず、ST で表現できない3次元パッキングが存在することが知られている (図3) [2]。

2.3 Sequence-Quintuple

Sequence-Quintuple (SQ) [2] は、5つの順列の組 $(\Gamma_1, \Gamma_2, \Gamma_3, \Gamma_4, \Gamma_5)$ により、直方体対の相対位置関係の制約を与え、3次元パッキングを表現する。任意の直方体対 a, b の相対位置関係の制約は以下のように与えられる。

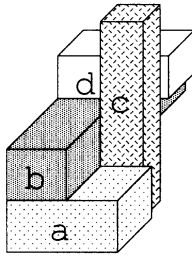


図3 STでは表現できない3次元パッキングの例.

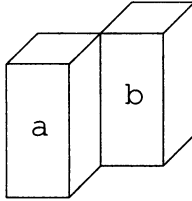


図4 $(\Gamma_1, \Gamma_2, \Gamma_3, \Gamma_4, \Gamma_5) = (ab, ab, ab, ab, ab)$ に対応するパッキング.

左右関係: $\Gamma_1^{-1}(a) < \Gamma_1^{-1}(b)$ かつ $\Gamma_2^{-1}(a) < \Gamma_2^{-1}(b)$ のとき, a は b の左.

前後関係: $\Gamma_3^{-1}(a) < \Gamma_3^{-1}(b)$ かつ $\Gamma_4^{-1}(a) < \Gamma_4^{-1}(b)$ のとき, a は b の手前.

上下関係: a と b が x - y 平面上で重なりがあり, $\Gamma_5^{-1}(a) < \Gamma_5^{-1}(b)$ のとき, a は b の下.

SQ は ST と同様, 各相対位置関係の制約に基づいて x, y, z 軸それぞれの制約グラフを作成し, 最長パス長を計算することにより直方体の座標値を求めると, 3次元パッキングが得られる.

与えられた SQ には対応する 3次元パッキングが必ず存在し, 与えられた 3次元パッキングに対応する SQ が必ず存在することが知られている [2].

しかし, 明らかに悪いパッキングを生成してしまう順列の組もある. 図4に, $(\Gamma_1, \Gamma_2, \Gamma_3, \Gamma_4, \Gamma_5) = (ab, ab, ab, ab, ab)$ という順列の組に対応するパッキングを示す. この順列の組が表す制約は Γ_1, Γ_2 より a は b の左, Γ_3, Γ_4 より a は b の手前となり, a と b は順列の組から2つの相対位置関係が直接与えられる. これにより, 直方体が配置されない空間ができてしまうことがある.

2.4 木構造データを用いた手法

O-Tree [4] を基にして考え出された木構造データを用いた直方体パッキング表現手法 [5] は, 兄弟関係を区別しない根つき多分木である x 制約, y 制約を表す2つの制約木と, z 制約を表す1つの順列からなる. SQ と同様, x, y の制約木により x, y 座標を定め, x - y 平面に関して重なっている直方体に対し, 1つの順列により上下方向の相対位置関係を定め, z 座標を決定する.

この表現手法は与えられた左下手前詰め3次元パッキングに対応する表現が必ず存在する [5]. また, SQ の解の総数が $(|M|!)^5$ に対して, この表現手法の解の総数は $(|M|+1)^{2(|M|-1)} \cdot |M|! < (|M|!)^5$ となり解空間が小さい. しかし, SQ と同様に, 直方体対に2つの相対位置関係が直接与えられる場合がある. また,

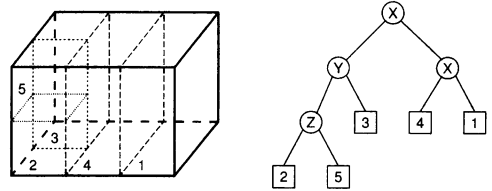


図5 スライス構造とスライス木.

隣接解の生成手法に問題があり, パッキング結果があまり良くない.

2.5 3D-subTCG

3D-subTCG [3] は, 順列を使わず, 推移律が成り立つ3つの DAG から一部の推移的な枝を省略した制約グラフにより3次元パッキングを表現する手法である. しかし, 与えられた3つの DAG が 3D-subTCG であると判定することが難しく, 解空間を探索する際の隣接解の生成も困難である. [3] では, 提案されている隣接解の生成手法では非許容解を生成してしまう場合がある.

2.6 スライス木を用いた手法

スライス木を用いた手法 [6] では, X, Y, Z のラベルによりスライス面の向きを表す $|M| - 1$ 個の内部点と, 各直方体に対応する $|M|$ 個の葉を持つ2分木により, 3次元パッキングを表現する (図5). 例えば, 内部点のラベルが Y ならば, その内部点の左側の子孫は右側の子孫より手前, という位置関係を持つ.

順列を用いた手法が3次元パッキングを生成するのに $O(|M|^2)$ の計算時間を要するのに対し, スライス木を用いた手法は $O(|M|)$ となり, 高速に3次元パッキングを得ることが可能である. しかし, スライス構造は一般構造に対し限定された構造であり, 表現できないパッキングがかなり存在する. 例えば, 図3に示したパッキングは表現できず, 図3から直方体 d を取り除いたパッキングも表現することができない.

3. Sequence-Quadruple

本稿では, 4つの順列の組 $(\Gamma_1, \Gamma_2, \Gamma_3, \Gamma_4)$ により, 3次元パッキングを表現する Sequence-Quadruple を提案する. Sequence-Quadruple は, 2つの順列で上下関係を表現し, 残りの2つの順列で左右, 前後関係を表すことで, ST やスライス木より表現できるパッキングの数が多くなり, また, 3次元スケジューリング問題における処理順序制約を考慮しやすくなると考えられる. 任意の直方体対 a, b の相対位置関係の制約は以下のように与えられる.

上下関係: $\Gamma_1^{-1}(a) < \Gamma_1^{-1}(b)$ かつ $\Gamma_2^{-1}(a) < \Gamma_2^{-1}(b)$ のとき, a は b の下.

左右関係: 上下関係がなく, $\Gamma_3^{-1}(a) < \Gamma_3^{-1}(b)$ かつ $\Gamma_4^{-1}(a) < \Gamma_4^{-1}(b)$ のとき, a は b の左.

前後関係: 上下関係がなく, $\Gamma_3^{-1}(a) < \Gamma_3^{-1}(b)$ かつ $\Gamma_4^{-1}(a) > \Gamma_4^{-1}(b)$ のとき, a は b の手前.

Sequence-Quadruple は ST, SQ, と同様に, 相対位置関係の制約に基づいて x, y, z 軸それぞれの制約グラフを作成し, 最長パス長を計算することにより各直方体の座標値を求めると, 3次元パッキングが得られる.

例として, $(\Gamma_1, \Gamma_2, \Gamma_3, \Gamma_4) = (abdc, cabd, bacd, dbca)$ につい

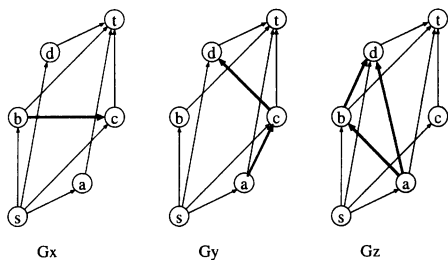


図6 $(\Gamma_1, \Gamma_2, \Gamma_3, \Gamma_4) = (abcd, cabd, bacd, dbca)$ の制約グラフ。

て考える。直方体 a と直方体 b は、 $\Gamma_1^{-1}(a) < \Gamma_1^{-1}(b)$ かつ $\Gamma_2^{-1}(a) < \Gamma_2^{-1}(b)$ なので、 a は b の下となる。直方体 a と直方体 c については、 $\Gamma_1^{-1}(a) < \Gamma_1^{-1}(c)$ かつ $\Gamma_2^{-1}(a) > \Gamma_2^{-1}(c)$ であるので、上下関係を持たない。そこで Γ_3, Γ_4 について調べると、 $\Gamma_3^{-1}(a) < \Gamma_3^{-1}(c)$ かつ $\Gamma_4^{-1}(a) > \Gamma_4^{-1}(c)$ であるので、 a は b の手前となる。このように、全ての直方体に対し相対位置関係を定めると、図6の制約グラフを得る。これより、図3のパッキングを得ることができる。

与えられた Sequence-Quadruple に対応する3次元パッキングが必ず存在するが、Sequence-Quadruple で表現できない3次元パッキングが存在する(図7)。

直方体パッキングは、直方体対が推移的に2つや3つの相対位置関係を持つことがある。例えば図3に示したパッキングは、 a は b の下にあり、 b は d の下にあるので、推移的に a は d の下にあるという相対位置関係を持つ。また、 a は c の手前にあり、 c は d の手前にあるので、推移的に a は d の手前にあるという相対位置関係を持つ。つまり、推移的に a は d の下かつ手前という2つの相対位置関係を持つことになる。このような推移的な2つの相対位置関係の組み合わせは、上下関係かつ左右関係で4通り、上下関係かつ前後関係で4通り、左右関係かつ前後関係で4通りあり、推移的な3つの相対位置関係の組み合わせは、上下関係かつ左右関係かつ前後関係で8通りある。

STは、直方体対が推移的な2つ以上の相対位置関係を持つパッキングを全く表すことができない。全てのパッキングを表現できるSQ、木構造データを用いた手法は、推移的な2つ以上の相対位置関係を持つパッキングを表すことができる。Sequence-Quadruple は、 Γ_1, Γ_2 の2つの順序列で上下関係を表し、 Γ_3, Γ_4 の2つの順序列で左右、前後関係を表すので、上下関係かつ左右関係の4通り、上下関係かつ前後関係の4通り、計8通りの推移的な2つの相対位置関係を持つパッキングを表現することが可能である。しかし、左右関係かつ前後関係の4通りと、推移的な3つの相対位置関係を持つパッキングを表すことができない。例えば、図7では a は c の左かつ奥、 b は d の左かつ手前という推移的な2つの相対位置関係を持つので Sequence-Quadruple では表現できない。しかし、Sequence-Quadruple では、上下関係かつ左右関係と上下関係かつ前後関係の推移的な2つの相対位置関係を表すことができるが、各直方体対には1つの相対位置関係しか直接与えられないので、図4のような効率の悪いパッキングを避けることができると思われる。

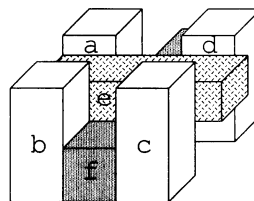


図7 Sequence-Quadruple では表現できない3次元パッキングの例。

4. 3次元スケジューリング手法

3次元スケジューリング問題は、幅(x 軸方向の長さ)、長さ(y 軸方向の長さ)、処理時間(時間軸方向の長さ)を持つタスク集合 M と、あるタスク b の処理が開始される前には、タスク a の処理が終了していなければならないというタスクの処理順序の制約を表す順序対 (a, b) の集合 C が与えられる。ここで、タスクを点に、処理順序制約を有向枝に対応させたグラフについて考える。もし、このグラフにサイクルが存在した場合、全ての処理順序制約を満たすスケジューリングを行なうことは不可能である。したがって、上記のようにグラフ表現をすると、与えられる処理順序制約はDAGとなる。3次元スケジューリング問題は、このような入力が与えられた時に、全ての処理順序制約を満たし、各タスクが3次元的に重なりがないという制約の下で、全タスクが出来るだけ少ない計算資源で短時間に完了するような各タスクの $x-y$ 平面への配置と処理の開始時間を求める問題である。3次元スケジューリング問題は、直方体パッキング問題を解く手法に処理順序制約を考慮することで解くことができる。

4.1 Sequence-Quadruple

Sequence-Quadruple は、 Γ_1, Γ_2 によって上下関係を表す。処理順序制約を表す順序対 (a, b) が与えられているとき、 $\Gamma_1^{-1}(a) < \Gamma_1^{-1}(b)$ かつ $\Gamma_2^{-1}(a) < \Gamma_2^{-1}(b)$ であれば、直方体パッキング問題で相対位置関係は、 a は b の下となり、3次元スケジューリング問題での処理順序制約を満たすことが保証される。我々は、全ての処理順序制約の順序対が Γ_1, Γ_2 2つの順序列の中で上記の順序関係にある順序列の組を処理順序制約許容解と呼び、この処理順序制約許容解空間のみで確率的探索手法を用いて準最適解を探索することにする。

処理順序制約を表す集合 C において、処理順序制約数を $m = |C|$ 個、処理順序制約が付いているモジュール数を $n \leq |M|$ とする。ある順序列の組が、処理順序制約許容解であるかを判断するための計算時間は $O(m)$ である。また、実際に処理順序制約を満たす順序列を得るには、全ての処理順序制約を満たすようにタスクに全順序をつけ、その順序で Γ_1, Γ_2 をソートすれば良いので、計算時間は $O(n \log n)$ である。

次に、与えられた処理順序制約が全て満たされている状況で、解空間を探索するときの隣接解の生成方法について考える。ここで、 $F_{in}(a)$ を処理順序制約の順序対 $(x, a) \in C$ であるタスク x の集合、 $F_{out}(a)$ を処理順序制約の順序対 $(a, x) \in C$ であるタスク x の集合とする。モジュール a が Γ_1, Γ_2 で $F_{in}(a)$ のどれか1つよりも前になると、処理順序制約許容解ではなくなる。同様に、 $F_{out}(a)$ のどれか1つよりも後になると、処理

順序制約許容解ではなくなる。したがって、 Γ_1 での a の移動可能範囲は、 $(\max_{i \in F_{in}(a)} \Gamma_1^{-1}(i), \min_{i \in F_{out}(a)} \Gamma_1^{-1}(i))$ 、 Γ_2 での a の移動可能範囲は、 $(\max_{i \in F_{in}(a)} \Gamma_2^{-1}(i), \min_{i \in F_{out}(a)} \Gamma_2^{-1}(i))$ となる。

全てのモジュールの移動可能範囲が分かっているとき、隣接解の処理順序制約が処理順序制約許容解であるかどうかは、 $O(1)$ で判断できる。例えば、 Γ_1 であるタスク a とあるタスク b を交換する場合、 $\max_{i \in F_{in}(a)} \Gamma_1^{-1}(i) < \Gamma_1^{-1}(b) < \min_{i \in F_{out}(a)} \Gamma_1^{-1}(i)$ 、かつ、 $\max_{i \in F_{in}(b)} \Gamma_1^{-1}(i) < \Gamma_1^{-1}(a) < \min_{i \in F_{out}(b)} \Gamma_1^{-1}(i)$ であれば良いからである。交換した後、交換したタスクに制約を与えている処理順序制約の交換していないタスクに対して移動可能範囲の更新する。

隣接解の生成手法を下記に示す。

(1) タスク 1 つを選び、幅と長さを交換する。(3次元直方体パッキング問題における z 軸を中心に 90° 回転させることに対応する。)

(2) タスク 2 つを選び、全ての順列において、2 つのタスクの順番を交換したものが処理順序制約許容解ならば交換する。

(3) タスク 2 つと順列 1 つを選び、1 つの順列において、2 つのタスクの順番を交換したものが処理順序制約許容解ならば交換する。

タスクを z 軸のみを中心に 90° 回転させるのは、 $x-y$ 平面でのタスクの向きは変えることができるが、処理時間は変えられないと仮定したからである。

【定理 1】 複数回の隣接解の生成により、処理順序制約許容解空間の任意の解から他の任意の解を生成することが可能である。(証明) Γ_3, Γ_4 は処理順序制約に関係なく交換できるので、それぞれ高々 $|M| - 1$ 回の交換で他の任意の解を生成することは可能である。また、タスクの向きを揃えるのも、高々 $|M|$ 回の回転で可能である。 Γ_1, Γ_2 は、バブルソートをする手順で順列での位置を交換すれば、処理順序制約に違反せず他の処理順序制約許容解を生成することができる。それぞれの順列で、高々 $\frac{1}{2} \cdot |M| \cdot (|M| - 1)$ 回の交換で可能である。 ■

3次元スケジューリング問題において、STはSequence-Quadrupleと同様に各タスクの $\Gamma_1, \Gamma_2, \Gamma_3$ の移動可能範囲を計算し、Sequence-Quadrupleと同様、3種の隣接解の生成手法を用いる。しかし、STは左右関係や前後関係の制約を変える際にも、移動可能範囲により制限してしまう。SQは、上下関係を左右関係の決め方に、左右関係を上下関係の決め方変えるだけで、Sequence-Quadrupleと同様に3次元スケジューリング問題に応用できる。Sequence-Quadrupleと同様、3種の隣接解の生成手法を用いる。

4.2 スライス木を用いた手法

スライス木を用いた手法では、内部点のラベルによりその左側の子孫と右側の子孫の位置関係が定まる。したがって、処理順序制約が付いている2つのタスクに最も近い共通先祖のラベルが Z で、処理順序制約の順序対における前のタスクが左側の子孫に、処理順序制約の順序対における後のタスクが右側の子孫にあれば、その処理順序制約が満たされることが保証される。全ての処理順序制約において、上記のようになっているスライス木を処理順序制約許容解と呼ぶ。

隣接解の生成手法を下記に示す。

(1) スライス木の1点(内部点または葉)を選び、幅と長さを交換する。(3次元直方体パッキング問題における選んだ点とその子孫からなる直方体全てを z 軸を中心に 90° 回転させることに対応する。)

(2) スライス木の中から、子孫と先祖の関係にない2点(内部点または葉)を選び、処理順序制約許容解になるならば、交換をする。

(3) スライス木の内部点を1つ選び、処理順序制約許容解になるならば、ラベルをランダムに付け替える。

[6]では、 x, y, z 軸を中心にした 90° 回転と(2)の隣接解の生成手法のみが提案されている。しかし、3次元スケジューリング問題では x 軸と y 軸を中心にした 90° 回転させないので、(1)と(2)の隣接解の生成手法のみでは Z とラベル付けされている内部点の数が変わらない。つまり、初期解に依存した3次元スケジューリングしか得られなくなってしまうので、(3)の隣接解の生成手法を付加した。

5. 実験結果

3次元スケジューリング問題における各手法の有効性を調べるため、処理順序制約許容解空間のみを探索するST、Sequence-Quadruple、SQ、スライス木を用いた手法をSimulated Annealing(SA)に組み込んで実験した。

[5]で用いられた、幅、長さ、処理時間をそれぞれ5から50の整数値からランダムに定め作成した30個のタスクを入力として用いた。初期温度と終了温度を予備実験によって定め固定し、温度降下係数を変更することによって探索時間を変更してSAの探索を行なった。SAの評価関数は、パッキング率とした。パッキング率とは、得られた3次元パッキングの外周直方体体積を直方体の体積の総和で割った値である。

実験には、Pentium4 3.06GHz/512K、512MB RAMの計算機を使用した。

まず、処理順序制約の数とパッキング率の関係について考察する。処理順序制約を0個から40個まで5個ずつ変化させた時のパッキング率を図8に表す。処理順序制約は、非許容解ができないように配慮し、ランダムに生成したものをを用いた。4つの手法全て計算時間を55秒程度にし、それぞれ、SAを10回適用した時の平均である。

順列を用いた手法の中では、処理順序制約が0個の場合、STを用いた手法の方がパッキング率が良いが、処理順序制約が5個以上の場合には、Sequence-Quadrupleを用いた手法の方がパッキング率が良い。またSQを用いた手法は、常にSequence-Quadrupleを用いた手法よりもパッキング率が悪くなることも分かる。これは、SQはタスク対に2つの相対位置関係が直接与えられるからであると考えられる。しかし、表現できるパッキングが少ないスライス木を用いた手法でも、順列を用いた手法より常に良い結果を得た。これは、スライス木を用いた手法は短時間により多くの解を探索できるからであると考えられる。

次に、処理順序制約がある場合の計算時間とパッキング率の関係について考察する。図8で用いたタスクにおいて、処理順序制約を25個に固定し、計算時間を変更した結果を図9に示す。それぞれ、SAを10回適用した時の平均である。

計算時間が短い場合には、スライス木を用いた手法のパッキ

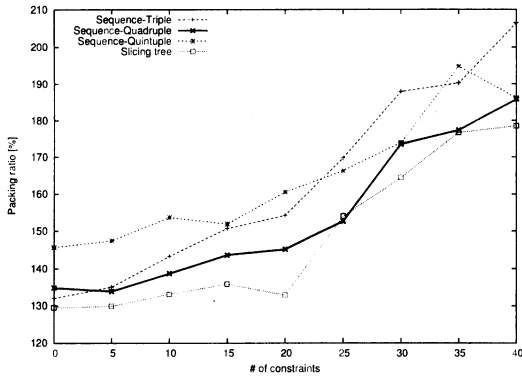


図 8 計算時間 55 秒, タスク数 30 における処理順序制約数とパッキング率の関係.

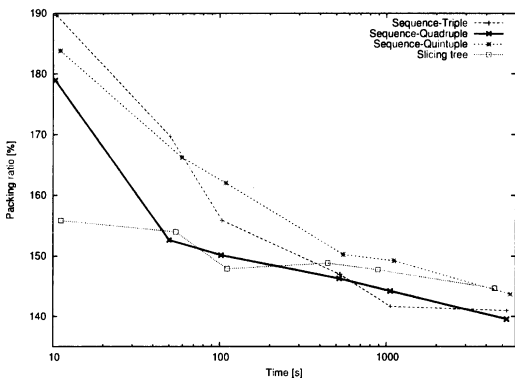


図 9 タスク数 30, 処理順序制約数 25 における計算時間とパッキング率の関係.

パッキング率が良くなる. しかし, 計算時間を長くした場合, ST や Sequence-Quadruple を用いた手法のパッキング率が良くなる傾向にある. これは, ST や Sequence-Quadruple を用いた手法の方が, より多くの解を表現できるからであると考えられる. しかし, 計算時間が長い場合での ST と Sequence-Quadruple の大きな違いを得ることはできなかった.

実験で得られたパッキングの例を図 10 に示す.

6. まとめ

本稿では, 4 つの順列の組を用いた直方体パッキングの表現手法 Sequence-Quadruple を提案し, 既存手法と Sequence-Quadruple の 3 次元スケジューリング問題に対する有効性を比較した. 計算時間が短い場合には, 解空間が狭く, 短時間でより多くの解を探索できるスライス木を用いる手法が良い結果になる. しかし, 計算時間が長くなると, より多くの解が表現できる ST や Sequence-Quadruple を用いた手法が良い解を得る傾向にある. しかし, 計算時間が長い場合での ST と Sequence-Quadruple の大きな違いを得ることはできなかった.

今後の課題としては, 3 次元スケジューリング問題において, どのような隣接解の生成手法が有効であるか考察すること, 木構造を用いた手法が 3 次元スケジューリング問題に応用でき

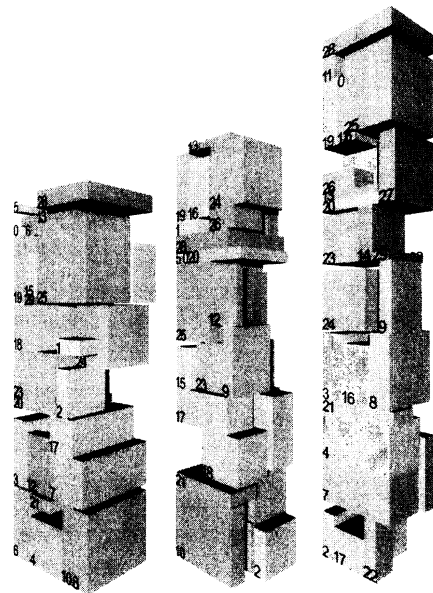


図 10 計算時間 55 秒, タスク数 30, 図 11 の処理順序制約を満たす 3 次元パッキング. それぞれのパッキング率は, 左の ST が 164.044%, 中央の Sequence-Quadruple が 149.829%, 右のスライス木を用いた手法が 150.237% である

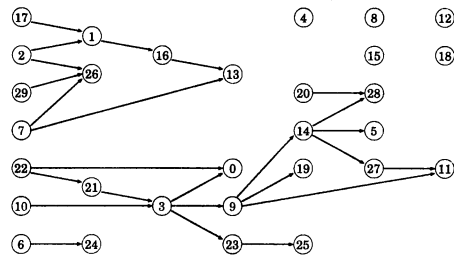


図 11 処理順序制約数 25 のグラフ表現.

ないか考察することである.

文 献

- [1] H. Murata, K. Fujiyoshi, S. Nakatake and Y. Kajitani: "VLSI Module Placement Based on Rectangle-Packing by the Sequence Pair", IEEE Trans. Computer-Aided Design, Vol.15, 12, pp. 1518-1524 (1996).
- [2] H. Yamazaki, K. Sakanushi, S. Nakatake and Y. Kajitani: "The 3D-Packing by Meta Data Structure and Packing Heuristics", IEICE Trans. Fundamentals, E82-A, 4 (2000).
- [3] P. Yuh, C. Yang, Y. Chang and H. Chen: "Temporal Floorplanning Using 3D-subTCG", Proc. ASP-DAC, pp. 725-730 (2004).
- [4] P. Guo, T. Takahashi, C. Cheng and T. Yoshimura: "Floorplanning Using a Tree Representation", IEEE Trans. Computer-Aided Design, Vol.20, pp. 639-645 (2001).
- [5] 川井英教, 藤吉邦洋: "木構造データを用いた直方体パッキング表現手法", 電子情報通信学会技術研究報告, VLD2004-29, pp. 49-54 (2004).
- [6] L. Cheng, L. Deng and D. Wong: "Floorplan Design for 3-D ICs", Proc. The 12th Workshop on SASIMI, pp. 395-401 (2004).