

# 荷台への荷物配置を考慮した配送計画問題の解法に関する研究

市川侑輝<sup>†1</sup> 太田秀典<sup>†1</sup> 中森眞理雄<sup>†1</sup>

荷物を集積所から顧客や小売店舗などに運搬車両を用いて配送するとき、物流コストを削減するために短い配送経路に沿って配送することが求められる。既存の研究の多くは、コンテナ内の荷物の配置について考慮していない。しかし、コンテナ内の荷物の配置は積載量や積み下ろしの手間に大きな影響を及ぼす。本論文では、配送計画問題に対して、荷物配置を考慮した最短運搬経路を求めるアルゴリズムを提案する。

## A Vehicle Routing Algorithm with Special Emphasis on Item Packing in Container

YUKI ICHIKAWA<sup>†1</sup> HIDENORI OHTA<sup>†1</sup>  
MARIO NAKAMORI<sup>†1</sup>

When delivering items from depots to customers or stores by vehicles, it is important to find the shortest paths in order to make the delivering cost the minimum. Most of the existing studies on the vehicle routing problem have taken less care on the problem of packing items in the container. The location of items in the container, however, has a significant influence on the capacity of delivery or the unpacking cost. In the present paper we propose a vehicle routing algorithm that includes item packing method in the container.

### 1. はじめに

荷物の集積所から各顧客、あるいは小売店などに運搬車（トラックなど、以下、「車両」と略す）を使って物を配送するとき、車両がどのような順序で配送を行えばよいかを考えることは、物流コスト削減の観点から非常に重要である。配送経路が最短となる車両の訪問順を求める問題はVRP（Vehicle Routing Problem）と呼ばれ、古くから盛んに研究され、さまざまなバリエーションが存在する[6]。

これらの研究では多くの場合、車両の積載量には制限が与えられており、制限量以内の荷物しか一度に運搬できないといった制約が課されている。しかしながら、現実のトラックでの荷物の運搬を考えると、積載量だけではなく、コンテナ内の荷物の配置についても十分に考慮をするべきである。例えば、荷物の形状が様々な大きさの直方体であった場合、荷物の配置を無作為に行うと、コンテナに荷物が密に配置されず、結果として期待された積載量を大幅に下まわる量の荷物しか一度に運搬できないことが考えられる。一度に運搬できる荷物量が少なくなれば、配送経路は増大してしまう可能性が高い。

無論、コンテナ内へ密に荷物を配置するだけであれば、大きな直方体領域に小さな直方体オブジェクトを配置する「直方体パッキング問題」を考えればよいが、配送順を考慮しないと、他の荷物が邪魔で取り出しが困難となる位置に荷物を配置してしまうことが考えられる。この場合、目的の荷物を降ろすため、降ろす必要のない邪魔な荷物を一

時的に積み下ろす等の操作を行う必要がある。一時的に積み下ろす荷物の数によっては、多くのコストが必要となってしまうことが考えられる。

以上のように、効率のよい荷物運搬の為にはコンテナ内への荷物の配置を考慮した上で、「運搬経路の経路長の最小化」と「積み付け後の荷物の移動回数の最小化」の2つの項目について考える必要がある。積み付け後の荷物の移動回数の最小化については、荷物の積み込み順序、積み下ろし順序が与えられたときに、荷物の一時取り出しを少なくする研究が行われている[2]。

他方、「運搬経路の経路長の最小化」については、コンテナへの荷物の積み付けを考慮しつつ運搬経路の短縮を行う手法がEmmanouilらによって提案されている[1]。この手法では、積み込み、積み下ろし以外では荷物の移動は行わず、一時的な積み下ろし等を行わないように荷物を配置するが、荷物やコンテナは2次元平面上で表現されており、3次元上での荷物の配置については十分に考慮することができない。また、集積所も1箇所のみの場合しか想定されておらず、集積所が複数存在する問題についても対応していない。

そこで本論文では、複数個所の集積所が与えられている配送計画問題について、上述したコンテナへの3次元上での荷物配置を考慮しつつ、運搬経路を短縮する手法を提案する。

なお、将来的には「運搬経路の経路長の最小化」と「積み付け後の荷物の移動回数の最小化」の2目的最適化を目指しているが、本論文ではEmmanouilらの手法と同様に、

<sup>†1</sup> 東京農工大学  
Tokyo University of Agriculture and Technology

目的の荷物の積み下ろしを除いて積み付け後の荷物の移動が行われない荷物配置のみを考えることにする。

## 2. 問題設定

積み込み地点と積み下ろし地点のそれぞれ設定された荷物集合を1台の車両を用いて運搬を行う際、運搬経路が最短となる「各顧客への訪問順」と「それぞれの荷物の配置位置」を考える。

車両のコンテナと荷物はともに直方体であり、コンテナの前後方向をx軸、左右方向をy軸、上下方向をz軸にとる。荷物の取り出し口はコンテナの手前に位置しており、各荷物は取り出し口からのみ積み込み、積み下ろしを行う。図1にコンテナと荷物の積み付け方向、積み下ろし方向を示す。

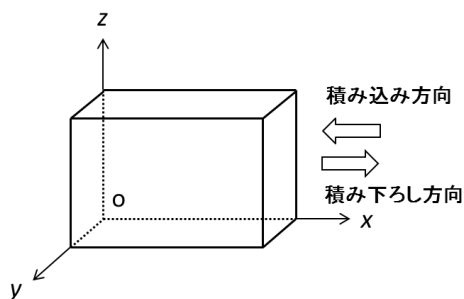


図1 コンテナと荷物の積み付け方向、積み下ろし方向

なお、各荷物はコンテナに対して平行に積み付けることにするが、どの面を上にして積み込んでよいことにする。また、重心等については考慮しないことにする。

荷物の積み込み（積み下ろし）を行う際、既に積み付けられている荷物の移動を行わないようにするため、制約条件として、直接積み付けられない位置への積み込みや直接取り出せない荷物の積み下ろしは行えないことにする。本論文では、荷物の取り出し口から見て別の荷物の奥側や下にある荷物は直接取り出せない荷物とみなす。具体的にはベクトル $(x, 0, z)$  ( $x, z$ は非負の任意の実数)の方向に荷物を移動させたとき、他のどの荷物にも重ならない荷物を、積み下ろし可能な荷物と定義することにする。積み下ろし可能な荷物と同様に、荷物を積み付け可能な位置も定義できる。例として、図2の(a)のようにコンテナ内に荷物**b**が積み付けられていた場合、荷物**b**を積み下ろすまで**a**の位置には新たな荷物を積み付けることはできない。また、図2の(b)のように荷物**c**の上に荷物**d**が積み付けられていた場合、荷物**d**を積み下ろすまで荷物**c**を積み下ろすことはできない。

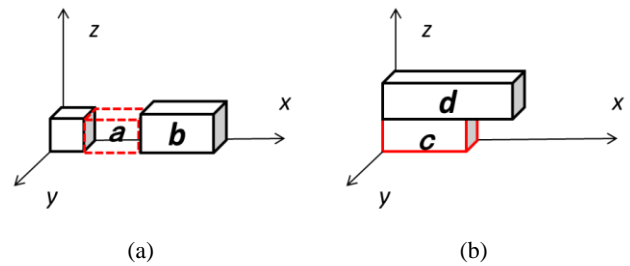


図2 積み付けられない位置と積み下ろせない荷物

## 3. スライス木によるパッキング表現

本論文では、スライス構造を用いて荷物の配置を表現する。スライス構造では表現できない配置も存在するが、3次元上ではある程度密な配置を得られることが確認されている[5]。

### 3.1 2次元スライス構造

2次元スライス構造とは、矩形領域（全体矩形）を鉛直線分（鉛直分割線分）、または水平線分（水平分割線分）で2つの矩形領域に再帰的に分割することで得られる構造のことである。この分割によって得られる各小矩形領域（部屋）に矩形を1対1で割り当てることにより、矩形パッキングを表現することができる。図3に2次元スライス構造の例を、図4に図3のスライス構造に基づく矩形パッキングを示す。

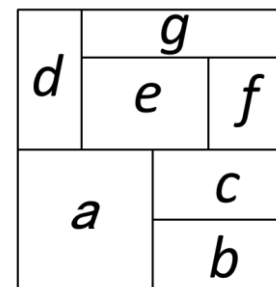


図3 2次元スライス構造

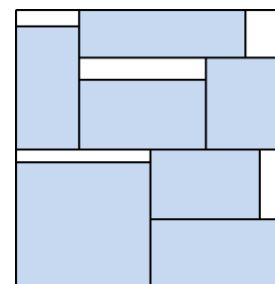


図4 スライス構造に基づく矩形パッキング

スライス構造はスライス木と呼ばれる2分木によって表現可能である。スライス木の内部節点は分割線に対応し、葉は部屋に対応している。内部節点には「X 節点」と「Y

「X 節点」の 2 種類があり、分割線が鉛直線分の場合は X 節点、水平線分の場合は Y 節点によって表現される。X 節点の左の部分木は鉛直分割線の左の矩形領域を表し、右の部分木は右の矩形領域を表している。同様に、「Y 節点」の左の部分木は水平分割線の下側の矩形領域を、右の部分木は上側の矩形領域を表す。図 5 に図 3 のスライス構造を表現したスライス木を示す。

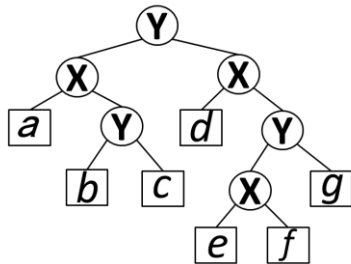


図 5 スライス構造を表現したスライス木

### 3.2 3次元スライス構造

2次元スライス構造を拡張することにより3次元スライス構造が得られる。6つの面がx軸、y軸、z軸に対してそれぞれ垂直な直方体領域（全体直方体）を、いずれかの軸に垂直な平面（分割面）によって再帰的に分割して得られる構造が3次元スライス構造である。この分割によって得られる各直方体領域に直方体（荷物）を1対1で割り当てることにより、直方体パッキングを表現することができる。図 6 に3次元スライス構造の例を示す。

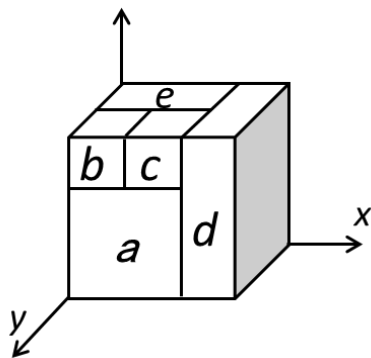


図 6 3次元スライス構造

2次元スライス構造と同様に、3次元スライス構造もスライス木による表現が可能である。スライス木の内部節点は分割面に対応し、葉は各直方体領域に対応する。内部節点の対応する分割面がx軸（y軸/z軸）と垂直な場合は「X節点（Y節点/Z節点）」となる。「X節点（Y節点/Z節点）」の左の部分木は分割面の前（左/下側）の直方体領域に、右の部分木は分割面の後ろ（右/上側）の直方体領域に対応する。図 7 に図 6 のスライス構造を表現したスライス木を示す。

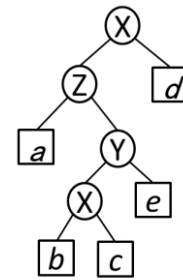


図 7 スライス構造を表現したスライス木

3次元スライス木の各部分木はスライス構造内の直方体領域に対応している。また、葉に対応する直方体領域には荷物が割り当てられるため、その直方体領域には割り当てられている荷物以上の大きさが必要となる。ここで、任意の X 節点 P を根とする部分木を考える。P の左の子の直方体領域の各辺の長さを  $X_L, Y_L, Z_L$  とし、右の子の直方体領域の各辺の長さを  $X_R, Y_R, Z_R$  とすると、部分木 P の対応する直方体領域の大きさ  $X_P, Y_P, Z_P$  は以下の式より求まる。

$$X_P = X_L + X_R, \quad Y_P = \max(Y_L, Y_R), \quad Z_P = \max(Z_L, Z_R)$$

Y 節点、Z 節点の部分木に関しても同様にして直方体領域の大きさを求めることができる。この性質より、スライス木に対応する荷物配置は、スライス木を後順に走査することで  $O(n)$  時間でデコード可能である。

### 3.3 拡張3次元スライス構造

本論文では、荷物の積み込みや積み下ろしを繰り返し行うので、ある時間の荷物配置だけではなく、各訪問地点上での荷物配置の移り変わりについて扱う必要がある。これは、x, y, z 軸に加え、時間軸も考慮にいった、いわば4次元上での荷物の配置として考えればよい。

4次元上でも、2次元上、3次元上と同様に、4次元直方体を x, y, z 軸、あるいは時間軸に垂直な面でも再帰的に分割して得られる4次元スライス構造が定義できる。分割によって得られる小4次元直方体領域に各荷物を割り当てることで、荷物配置の移り変わりについて表現を行う。

4次元スライス構造もまた、3次元スライス木に時間軸に垂直な面による分割を表す「T 節点」を追加することで、スライス木による表現が可能である。小畑らはこれを拡張3次元スライス木と定義した。拡張3次元スライス木の T 節点の左の部分木に含まれるどの荷物の積み下ろしも、右の部分木に含まれるどの荷物の積み込みより先に行われることにする。

拡張3次元スライス木において、X, Y, Z 節点を根とする部分木に対応する（時間軸を除いた3次元の）直方体領域の大きさは、3次元スライス木の場合と同様に考えることができる。T 節点については、左の子の直方体領域の各辺の長さを  $X_L, Y_L, Z_L$  とし、右の子の直方体領域の各辺の長さを  $X_R, Y_R, Z_R$  とすると、T 節点 Q を根とする部分木

に対応する直方体領域の大きさ  $X_Q, Y_Q, Z_Q$  は以下の式により求まる。

$$X_Q = \max(X_L, X_R), Y_Q = \max(Y_L, Y_R), Z_Q = \max(Z_L, Z_R)$$

これより、3次元スライス木と同様に、スライス木に対応する荷物配置は、スライス木を後順に走査することで  $O(n)$  時間でデコード可能である。なお、全体直方体の各辺の長さがコンテナの各辺の長さより短いとき、そのスライス木で表現された荷物配置はコンテナ内にすべて収まることになる。[2]

### 3.4 スライス木に対応した運搬経路

#### 3.4.1 スライス木と積み下ろし順序

本論文では、既に積み付けられている荷物の移動は行われないと仮定しているため、荷物の配置によって荷物の積み下ろし順序がある程度制約される。つまり、スライス木によって運搬経路がある程度決定されることになる。ここで、任意の節点  $P$  を根とする部分木において、 $P$  の節点の種類 (X 節点, Y 節点, Z 節点, T 節点) ごとに荷物の積み込み, 積み下ろし順序を示す。なお、 $P$  の左右の部分木と対応する直方体領域をそれぞれ  $L, R$  とする。

(1)  $P$  が X 節点の場合, Z 節点の場合

$P$  が X 節点の場合や Z 節点の場合、 $P$  を根とする部分木の直方体領域において、直方体領域  $L, R$  の位置関係はそれぞれ図 8, 図 9 のようになる。

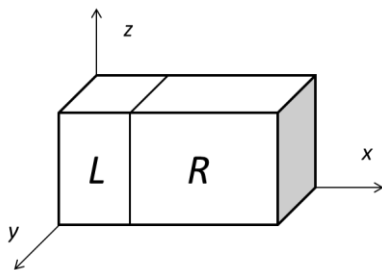


図 8 X 節点の場合の  $L, R$  の位置関係

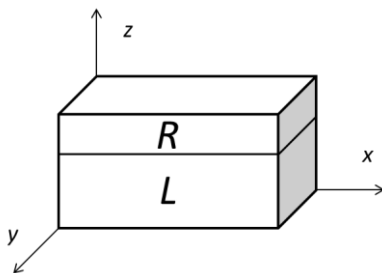


図 9 Z 節点の場合の  $L, R$  の位置関係

既に積み付けられた荷物の移動が行われないための制約条件を満たすためには、荷物の積み下ろしを、直方体領域  $L$  に含まれる荷物の積み込み, 直方体領域  $R$  に含まれる荷物の積み込み, 直方体領域  $R$  に含まれる荷物の積み下ろし, 直方体領域  $L$  に含まれる荷物の積み下ろしの順で行えば十

分である。このとき、 $P$  の左の子の直方体領域において、最後に積み込まれる荷物の積み込みを  $L_{last\_in}$ , 最初に積み下ろされる荷物の積み下ろしを  $L_{first\_out}$  とし、また、 $P$  の右の子の直方体領域において、最初に積み込まれる荷物の積み込みを  $R_{first\_in}$ , 最後に積み下ろされる荷物の積み下ろしを  $R_{last\_out}$  としたとき、各操作の順序は  $L_{last\_in}, \dots, R_{first\_in}, \dots, R_{last\_out}, \dots, L_{first\_out}$  となる。

(2)  $P$  が T 節点の場合

$P$  が T 節点の場合、T 節点の定義から、直方体領域  $L$  内の荷物の積み下ろしがすべて完了した後、直方体領域  $R$  内の荷物の積み込みを行う。したがって、 $P$  の左の部分木が対応する直方体領域において、最後に積み下ろされる荷物の積み下ろしを  $L_{first\_out}$  とし  $P$  の右の部分木が対応する直方体領域において、最初に積み込まれる荷物の積み込みを  $R_{first\_in}$  としたとき、操作の順序は  $L_{last\_out}, \dots, R_{first\_in}$  となる。

(2)  $P$  が Y 節点の場合

$P$  が Y 節点の場合、 $P$  を根とする部分木の直方体領域において、直方体領域  $L, R$  の関係は図 10 のようになる。

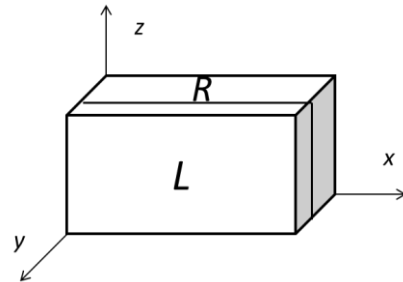


図 10 Y 節点の場合の  $L, R$  の位置関係

このとき、直方体領域  $L, R$  にそれぞれ含まれる荷物集合は直方体領域  $L, R$  内でそれぞれ制約される相対的な順序さえ満たせば、どのような順序でも荷物を積み込み, 積み下ろしが可能である。本論文では拡張3次元スライス木を用いて積み込み地点や積み下ろし地点の訪問順を一意に表現するため、各 Y 節点に補助的なリストを持たせることにする。このリストは、「左」と「右」を表す2つの要素で構成されており、リストの長さは、その Y 節点を根とする部分木に含まれる荷物数の2倍となっている。「左」の要素は Y 節点の左の部分木に含まれる荷物数の2倍の要素数、「右」の要素は右の部分木に含まれる荷物数の2倍の要素数となっている。リストは左の部分木に含まれる荷物の積み下ろしや積み込みの操作の順序と、右の部分木に含まれる荷物の積み下ろしや積み込みの操作の順序をどのようにマージするかを示すものである。なお、以後はこのリストのことを順序リストと表記する。

例として、図 11 の拡張3次元スライス木について積み込み, 積み下ろし操作の順序について示す。なお、図 11 の2分木内の  $Y_1$  節点は図 12 のリストを持ち、 $Y_2$  節点は図 13 のリストを持つものとする。

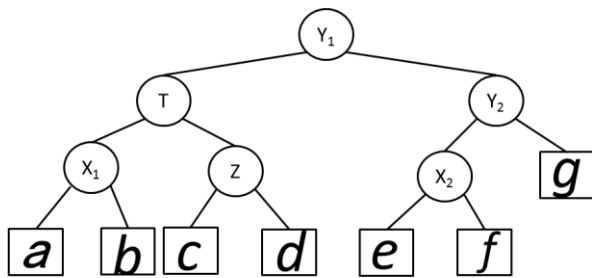


図 11 Y 節点を根とする部分木

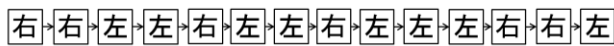


図 12 図 11 の Y<sub>1</sub> 節点を持つリスト

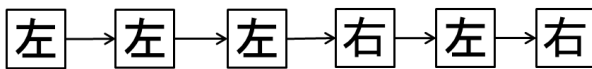


図 13 図 11 の Y<sub>2</sub> 節点を持つリスト

荷物  $a$  の積み込みを  $A_{in}$ , 積み下ろしを  $A_{out}$  とする. 同様に荷物  $b$  から  $g$  においても積み込みと積み下ろし操作を定義する. まず,  $Y_1$  節点の左の部分木について積み込み, 積み下ろし操作の順序を考える. 荷物  $a, b$  の親は  $X_1$  節点であるため, 荷物  $a, b$  の積み込み, 積み下ろし操作の順序は  $A_{in}, \dots, B_{in}, \dots, B_{out}, \dots, A_{out}$  となる. 荷物  $c, d$  の親は  $Z$  節点であるため, 荷物  $c, d$  の積み込み, 積み下ろし操作の順序は  $C_{in}, \dots, D_{in}, \dots, D_{out}, \dots, C_{out}$  となる. ここで,  $X_1$  節点と  $Z$  節点の親は  $T$  節点であることより, 荷物  $a, b, c, d$  の積み込み, 積み下ろし操作の順序は,  $A_{in}, \dots, B_{in}, \dots, B_{out}, \dots, A_{out}, \dots, C_{in}, \dots, D_{in}, \dots, D_{out}, \dots, C_{out}$  となる. 次に,  $Y_1$  節点の右の部分木について積み込み, 積み下ろし操作の順序を考える. 荷物  $e, f$  の親は  $X_2$  節点であるため, 荷物  $e, f$  の積み込み, 積み下ろし操作の順序は  $E_{in}, \dots, F_{in}, \dots, F_{out}, \dots, E_{out}$  となる. また, 節点  $X_2$  と荷物  $g$  の親は  $Y_2$  節点である. ここで,  $Y_2$  節点は図 13 のリストを持ち, このリストは左右の部分木の積み込み, 積み下ろし操作の順序を「左の部分木から 3 つ」「右の部分木から 1 つ」「左の部分木から 1 つ」「右の部分木から 1 つ」という手順でマージすることを示している. よって, 荷物  $e, f, g$  の積み込み, 積み下ろし操作の順序は,  $E_{in}, \dots, F_{in}, \dots, F_{out}, \dots, G_{in}, \dots, E_{out}, \dots, G_{out}$  となる. 最後に, 図 12 のリストに従って  $Y_1$  の左右の部分木の積み込み, 積み下ろし操作の順序をマージする. このリストに従うと, 荷物  $a$  から  $g$  の積み込み, 積み下ろし操作の順序は,  $E_{in}, F_{in}, A_{in}, B_{in}, F_{out}, B_{out}, A_{out}, G_{in}, C_{in}, D_{in}, D_{out}, E_{out}, G_{out}, C_{out}$  となる.

## 4. 探索手法

拡張 3 次元スライス木と順序リストを用いて, 荷物の積み込み, 積み下ろし順序を考慮したパッキング表現を行い, 荷物の運搬経路を作製する. 拡張 3 次元スライス木と順序リストを変更することで運搬経路を変化させ, 経路長の最小値を求める. 探索は Simulated Annealing (SA 法) によって行う.

### 4.1 初期解

初期解として図 14 に示す, すべての節点が  $T$  のスライス木を用いる.

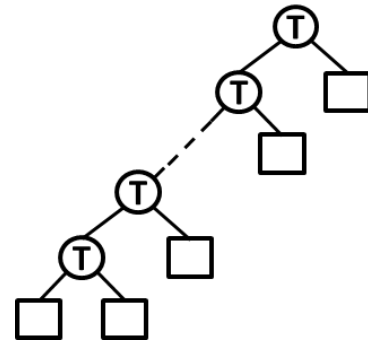


図 14 初期解

このスライス木は荷物の積み込みと積み下ろしを交互に行うことを表しており, コンテナ内にはせいぜい 1 個の荷物しか存在しないため, 通常はどの荷物も必ずコンテナ内に収まるので初期解は許容解となる.

### 4.2 近傍

以下の 5 つの操作からランダムに 1 つ選択し, 近傍解の生成を行う:

1. 荷物回転: ランダムに 1 つの荷物を選択し, 荷物の積み付け向きを変更する;
2. 荷物交換: 拡張スライス木からランダムに 2 つの葉を選択し, それらを交換する;
3. 配置変換: 拡張スライス木からランダムに 1 つの節点を選択し, その節点の種類 ( $X$  節点,  $Y$  節点,  $Z$  節点,  $T$  節点) を変換する. なお,  $Y$  節点に変換した場合は, その  $Y$  節点に持たせる順序リストをランダムに作成する;
4. 部分木交換: 拡張スライス木からランダムに 2 つの節点を選択し, それらの節点を根とする部分木同士を交換する. ただし, 片方の部分木にもう片方の部分木が含まれていた場合は交換を行わない. また, 選択した 2 つの節点の先祖をそれぞれ辿り, 共通の先祖となる節点までに  $Y$  節点が存在する場合, それらの  $Y$  節点を持つ順序リストをランダムに作成し直す;
5. 順序リストの変更: ランダムに  $Y$  節点を 1 つ選択し, その節点を持つ順序リストから 1 つ要素を選び, 選んだ要素をリスト内の別の位置へ挿入する.

なお、非許容解が生成された場合、近傍解の作成をやり直す。また、目的関数は運搬経路の経路長であるが、近傍の「荷物回転」では経路長が変化しないため、この近傍を用いたときのみ経路長の代わりに「荷物がコンテナで占める総体積」を評価する。

## 5. 計算機実験

提案手法の有効性を評価するため、計算機実験を行った。実験環境は CPU が Intel Core i7-2600 3.40GHz、メモリが 4.00GB、実装言語には C 言語を用いた。

インスタンスとして、 $50 \times 50$  の座標上からランダムに 10 地点の顧客や集積所の位置を設定した。荷物は 36 個用意し、それぞれの積む地点と降ろす地点を 10 地点からランダムに選択した。なお、設定した座標を基に配送計画問題を単純な経路問題として解いたとき（コンテナ内の荷物配置を考慮しないとき）、経路長は 176 であった。

荷物の形状は直方体であり、各辺の長さが、(65, 65, 65) の荷物が 15 個、(65, 65, 130) の荷物が 7 個、(65, 130, 130) の荷物が 7 個、(130, 130, 130) の荷物が 7 個となっている。また、コンテナとして 5 種類の直方体を用意した。各コンテナのサイズを表 1 に示す。インスタンス 1 から 3 は実際のコンテナの縦横比を参考にし、 $x:y:z=5:2:3$  とした。一方、インスタンス 4, 5 では、荷物の取り出し口がトラックの進行方向から見て側面に位置するコンテナを想定し、 $x:y:z=2:5:3$  とした。なお、パッキング率とは、コンテナの体積に対するすべての荷物の体積和の割合である。

図 15 にインスタンス 1, 2, 3 の、図 16 にインスタンス 4, 5 の運搬経路の経路長と探索時間の関係を示す。また、図 17, 図 18 にインスタンス 1 と 4 の経路を示す。

表 1 各インスタンスのコンテナのサイズ

#	パッキング率	コンテナのサイズ
1	100%	$500 \times 200 \times 300$
2	50%	$650 \times 260 \times 390$
3	25%	$812 \times 325 \times 488$
4	100%	$200 \times 500 \times 300$
5	50%	$260 \times 650 \times 390$

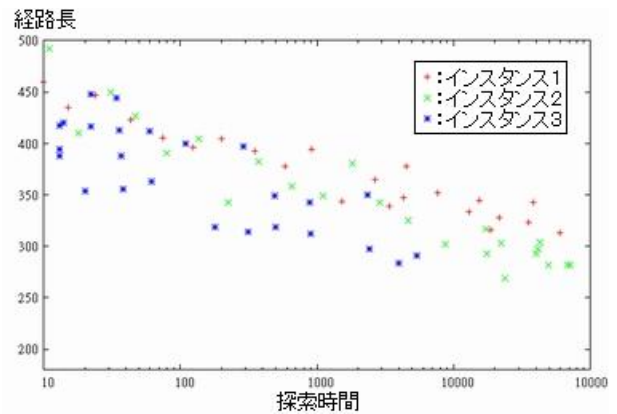


図 15 インスタンス 1, 2, 3 の探索時間と経路長の関係

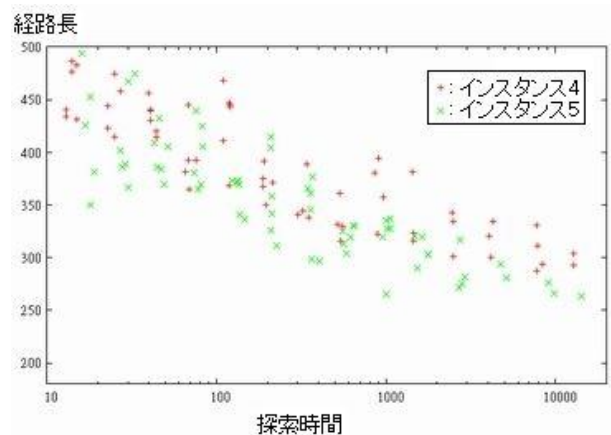


図 16 インスタンス 4, 5 の探索時間と経路長の関係

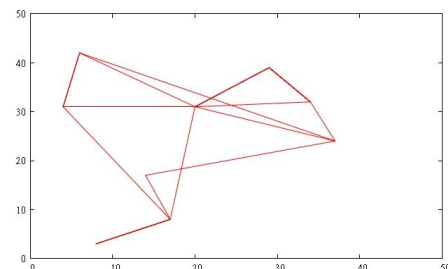


図 17 インスタンス 1 の経路 (経路長: 317)

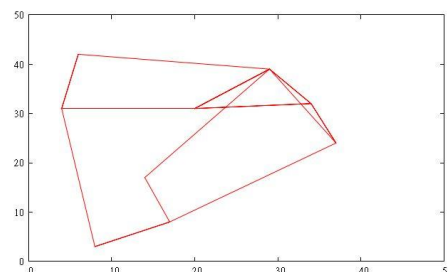


図 18 インスタンス 4 の経路 (経路長: 287)

探索時間が短い場合は各パッキング率で経路長に差はないが、探索時間が長くなるにつれてパッキング率が低いほど経路長が短くなっている。また、荷物の取り出し口が

側面に位置しているコンテナの方が短い探索時間で経路を短縮できている。

インスタンス 1, 2, 3, 4, 5 の経路長は、単純な経路問題として解いた場合に対し 1.49 倍から 1.77 倍の経路長を求められている。これは厳密解にどの程度近い値であるかは不明だが、コンテナ内の荷物配置を考慮していることを考えると、ある程度は良い経路を得られていると考えられる。

## 6. まとめ

本論文では、コンテナ内の荷物配置を考慮した配送計画問題について論じ、SA を用いた探索手法を提案した。

本論文で用いた手法では、巡回経路は制約条件を十分満たしているが、荷物の配置に対して最適な経路に到達できていない可能性があるため、荷物の配置が与えられたときの巡回経路の作成の改善が今後の課題として挙げられる。また、スライス木では表現できない荷物の配置を行うことや、「運搬経路の経路長の最小化」と「荷物の積み付け後の移動回数の最小化」の 2 目的を最適化する手法の提案が挙げられる。

## 参考文献

- [1] E. E. Zachariadis, C. D. Tarantilis and C. T. Kiranoudis “ A Guided Tabu Search for the Vehicle Routing Problem with two-dimensional loading constraints” European Journal of Operational Research, Volume 195, pp.729–743, 2009.
- [2] 小畑尚輝, 太田秀典, 中森真理雄: “積み込み, 積み下ろしを考慮した 3 次元パッキング問題に関する研究”, 情報処理学会研究報告, 2011-MPS-85(17), pp. 1-7
- [3] D. F. Wong, C. L. Liu, “A new algorithm for floorplan design,” IEEE DAC, pp.101-107, 1986.
- [4] L. Cheng, L. Deng and M. D. F. Wong, “Floorplan Design for 3-D ICs,” Proc. SASIMI, pp.395-401, 2004.
- [5] 小平行秀, 児玉親亮, 藤吉邦洋, 高橋篤司: “計算資源割り当てスケジューリングのための直方体パッキング表現の検討”, 第 18 回 回路とシステム (軽井沢) ワークショップ, pp.211-216, 2005
- [6] G. Laporte and I. H. Osman, “Routing Problems: A Bibliography,” Annals of Operations Research, vol. 61, pp.227-262, 1995.