3次元パッキング問題の実用的研究

2007MI030 原雅典　2007MI106 小木曽匠
指導教員：鈴木敦夫

1　はじめに

1.1　背景と研究目的

本研究では，サイズの異なる多数の直方体状の荷物をトラックへ効率的に積み込む問題について考える．一般に企業では荷物の仕分けを配送先別に行い，トラックに積み込んで出荷を行っており，データの提供を受けた企業でも一日あたり 3,895 個の荷物を 14 便のトラックに分けて配送している．積み込み作業は現場での経験に基づいて行われており，熟練者の勘に頼る傾向にある．そのため，熟練者不在の場合は適した積み込みができない，全社的に方法を共有できない，技術の伝承が困難であるなどといった問題が考えられる．そこで本研究では効率的な荷物の積み込みを行うための分析を行う．

1.2　研究対象

研究対象とするのは企業から提供された様々な荷物やパレット，トラックの大きさに関するデータである．パレットとは，運搬の際にフォークリフトの爪を差し込んで持ち上げるための簀の子のことであり，大きさは幅・奥行き・厚さがそれぞれ 1,200・1,000・130 (mm) である．荷物はこれに積みつけられてトラックへ載せられる．その際，荷物はパレットからできるだけはみ出ないように配置させるが，荷物の幅の 1 割まではパレットからはみ出させてもよい．奥行きについてはパレットからはみ出ないように荷物を置く必要がある．さらに，パレットに荷物を載せられる高さには上限があり，おおよそ 1,000(mm) である．また，荷物については周りを囲む外枠はあるが上面を覆うふたがないため，上に荷物が載せられるかどうかについては下の荷物の大きさや形によることになる．

トラックの容積については，それを横から見たときの幅・奥行き・高さはそれぞれ 9,900・2,410・2,400(mm) である．トラックを正面から見て縦に二等分し，右側と左側にそれぞれ 8 枚のパレットを置くことができる．また，全てのパレットに載せる荷物の高さを上限近くまで合わせると，さらに左右で 8 枚ずつのパレットを載せられることになる．以上より，使用するパレットの枚数が 32 枚以下となれば全てのパレット，すなわち全ての荷物を載せられることになる．ただし，1 つの荷物でパレットの 2 倍近い幅の長さの荷物もあり，その荷物はパレットを 2 枚使うのではなく 1 枚のパレットの中央に載せる．つまり，その個数によりトラックに載せきることのできるパレット枚数の評価も変わることになる．本研究ではそれらを考慮した上でトラックに全ての荷物を載せられるかどうかを分析する．

2　問題解決の手順

企業から提供されたデータに対し，各トラックへの荷物の割り当てから効率的な荷物の積み込みまでの分析を行う．

荷物の割り当てについては，あらかじめ使用するトラックの便数を指定した上で行い，ある一便分の荷物のデータを用いて積み込みについて分析する．荷物の積み込みについては 3 次元の積み込みを考える必要があるが，2 次元と 1 次元の 2 段階に分けての積み込みを提案した．第 1 段階では 2 次元パッキングを行う．ここでは，荷物を平面上に敷き詰めて複数の層を作成する．第 2 段階の 1 次元パッキングでは，2 次元パッキングで作成した層を上下に重ね合わせ，パレットごとに層の組み合わせを求める．その後，層単位で上下に重ね合わせられるかどうかの判定を行い，下から上に向けて層の重ねる順番を求める．最後に全てのパレットをトラックに載せられるかどうか分析する．分析には最適化ソフト What's Best!9.0 を用いた．

3　荷物の割り当て

荷物の割り当ては，全てのトラックへ均等に割り当てる必要がある．研究対象の企業では，荷物の箱の中身を区別して早い便から順番に均等に割り当てを行い，使用するトラックの便数より荷物数が少なくなった場合は早い便から順番に載せるといったものである．その方法で割り当てを行った結果，早い便ほど荷物数が多く，遅い便ほど荷物数が少なくなるといった極端な偏りが出た．そこで，今回は全ての荷物のデータから荷物の箱の中身を区別した上で，荷物の数と体積の平均を基に各トラックへの荷物の割り当てを行う．

3.1　記号の定義

添字
- I：トラックの集合
- J：荷物の集合

定数
- $|I|$：使用するトラックの便数
- b_j：荷物 j の箱数
- V_j：荷物 j の一個あたりの体積
- n ：荷物数

変数
- x_{ij}：トラック i 便の荷物 j の箱数

3.2　定式化

制約条件

$$\sum_{i\in I} x_{ij} = b_j \qquad (j \in J) \tag{1}$$

$$\sum_{j\in J} x_{ij} \le \{(\sum_{j\in J} b_j)/|I|\} + \alpha \qquad (i \in I) \tag{2}$$

$$\sum_{j\in J} V_j x_{ij} \le \{(\sum_{j\in J} V_j b_j)/|I|\} + \beta \qquad (i \in I) \tag{3}$$

$$\lfloor b_j/|I| \rfloor \leq x_{ij} \leq \lceil b_j/|I| \rceil \qquad (i \in I, j \in J) \tag{4}$$

制約条件 (1)　トラックに割り当てる各荷物の総和はその荷物の箱数と等しい制約

制約条件 (2)　各トラックに割り当てる荷物数を平均に近づける制約

制約条件 (3)　各トラックに割り当てる荷物の総体積を平均に近づける制約

制約条件 (4)　各荷物を全てのトラックへ均等に割り当てる制約

4　2 次元パッキングについて

ここでは荷物を平面上に敷き詰めることで，複数の層を作成する 2 次元のパッキングを行う．層の作成方法として，パレットの大きさに入るようなあらゆる詰め込みパターンから必要なパターンのみを取り出す列挙法を提案した．パターンの列挙の対象は，高さの同じ荷物同士，および荷物を重ね合わせることで高さを揃えられる荷物同士である．全てのパターンの列挙は手動で行い，荷物の横への 90 度回転やパレットからの荷物のはみ出しを考慮した上で定式化した．このようにパターンを列挙する問題はカッティングストック問題と呼ばれる [3].

4.1　記号の定義

添字

I：荷物の集合

J：積み込みパターンの集合

N：自然数全体の集合

定数

h_j：積み込みパターン j の高さ

a_{ij}：積み込みパターン j にある荷物 i の個数

b_i：荷物 i の個数

変数

x_j：積み込みパターン j を選び出す回数

4.2　定式化

目的関数

$$\min. \quad \sum_{j \in J} h_j x_j$$

制約条件

$$\sum_{j \in J} a_{ij} x_j = b_i \qquad (i \in I) \tag{5}$$

$$x_j \in N \qquad (j \in J) \tag{6}$$

目的関数　使用するパターンの高さの総和を最小にする式

制約条件 (5)　各荷物の個数は使用するパターンに含まれる荷物数と等しい制約

制約条件 (6)　自然数制約

4.2.1　列挙法の問題点とそれに対する改善策

荷物の個数により様々な積み込みパターンが選ばれることになるが，中には荷物の数が少なく隙間が大きいパターンも選ばれることがある．そのようなパターンが選ばれた際にその隙間に更に荷物を詰め込むために，パターンごとにそこに含まれる複数の荷物を一つの大きな荷物として見立て，それらを対象に再度 2 次元パッキングを行う．その際，レベル法 [1] と呼ばれる手法を用いる．

4.3　レベル法

レベル法とは，荷物を詰め込む順番を奥行きの降順に並び換えた上で荷物を詰め込む容器をレベルと呼ばれる領域に分割し，レベルごとに荷物を左から右，手前から奥に向かって隙間を見ながら配置を繰り返す方法である．レベルの大きさは荷物の大きさによって決まり，荷物の奥行きの最も大きいものがそのレベルの奥行きの長さとなる．あるレベルに初めて荷物が配置された時，その荷物はレベルを規定し，ある容器に初めてレベルが配置された時，そのレベルは容器を規定する．

新たな荷物を配置する場所の候補は，レベルの奥行きが新たに置く荷物の奥行きより小さい場合は，そのレベルには収まらないので新たなレベルを規定する．また，新たに配置する荷物の奥行きがレベルの奥行きより小さい場合はそのレベルの既配置の荷物の横に配置するか，あるいは新しいレベルを規定し，そこに配置することになる．このような手順を繰り返して全ての荷物の積み込みを行う．

4.3.1　記号の定義

添字

I：荷物の集合

定数

W：パレットの幅

D：パレットの奥行き

w_i：荷物 i の幅

d_i：荷物 i の奥行き

n ：荷物数

変数

$$y_j = \begin{cases} 1 & \text{荷物 } j \text{ がレベル } j \text{ を規定する} \\ 0 & \text{荷物 } j \text{ がレベル } j \text{ を規定しない} \end{cases} \quad j \in I$$

$$q_k = \begin{cases} 1 & \text{レベル } k \text{ が層 } k \text{ を規定する} \\ 0 & \text{レベル } k \text{ が層 } k \text{ を規定しない} \end{cases} \quad k \in I$$

$$x_{ij} = \begin{cases} 1 & \text{荷物 } i \text{ がレベル } j \text{ にある} & j \in I \backslash \{n\} \\ 0 & \text{荷物 } i \text{ がレベル } j \text{ にある} & j < i \end{cases}$$

$$z_{jk} = \begin{cases} 1 & \text{レベル } j \text{ が層 } k \text{ にある} & k \in I \backslash \{n\} \\ 0 & \text{レベル } j \text{ が層 } k \text{ にない} & k < j \end{cases}$$

$$s_i = \begin{cases} 1 & \text{荷物 } i \text{ を回転させない} \\ 0 & \text{荷物 } i \text{ を回転させる} \end{cases}$$

4.3.2　定式化

目的関数

$$\min. \quad \sum_{k=1}^{n} q_k$$

制約条件

$$\sum_{j=1}^{i-1} x_{ij} + y_i = 1 \qquad (i = 1, \cdots, n) \tag{7}$$

$$\begin{aligned}&\sum_{i=j+1}^{n} \{w_i x_{ij} s_i + d_i x_{ij}(1-s_i)\} \\ &\qquad \leq [W - \{w_j s_j + d_j(1-s_j)\}] y_j \\ &\qquad\qquad (j = 1, \cdots, n-1)\end{aligned} \tag{8}$$

$$\sum_{k=1}^{j-1} z_{jk} + q_j = y_j \qquad (j = 1, \cdots, n) \tag{9}$$

$$\begin{aligned}&\sum_{j=k+1}^{n} \{d_j z_{jk} s_j + w_j z_{jk}(1-s_j)\} \\ &\qquad \leq [D - \{d_k s_k + w_k(1-s_k)\}] q_k \\ &\qquad\qquad (k = 1, \cdots, n-1)\end{aligned} \tag{10}$$

$$\begin{aligned}&d_j y_j s_j + w_j y_j (1-s_j) \\ &\qquad \geq d_i x_{ij} s_i + w_i x_{ij}(1-s_i) \\ &\qquad (j = 1, \cdots, n-1; i = j+1, \cdots, n)\end{aligned} \tag{11}$$

$$y_j,\, q_k,\, x_{ij},\, z_{jk},\, s_i \in \{0, 1\} \qquad \forall i,\, j,\, k \tag{12}$$

目的関数　層の数を最小にする式
制約条件 (7)　各荷物がいずれかのレベルに入る制約
制約条件 (8)　各レベルの幅がパレットの幅以下である制約
制約条件 (9)　各レベルがいずれかの層に入る制約
制約条件 (10)　各層に入るレベルの奥行きの総和がパレットの奥行き以下である制約
制約条件 (11)　各レベルについて，レベルを規定する荷物の奥行きは他の荷物の奥行き以上である制約
制約条件 (12)　バイナリー変数制約

5　1 次元パッキングについて

第 2 段階の 1 次元パッキングでは，2 次元パッキングで求めた層の高さのみに着目して層の重ね合わせを行う．つまり，パレットごとに層の組合せを選ぶことになる．これをパレットパターンと呼ぶ．2 次元パッキングで提案した列挙法と同様に，全てのパレットパターン列挙して必要なもののみを選び出すカッティングストック問題として定式化した．

5.1　記号の定義

添字
　I：層の集合
　J：パレットパターンの集合
　N：自然数全体の集合
定数
　a_{ij}：パレットパターン j にある層 i の個数
　b_i：層 i の個数
変数
　x_j：パレットパターン j を選び出す回数

5.2　定式化

目的関数

$$\min. \quad \sum_{j \in J} x_j$$

制約条件

$$\sum_{j \in J} a_{ij} x_j = b_i \quad (i \in I) \tag{13}$$

$$x_j \in N \qquad (j \in J) \tag{14}$$

目的関数　使用するパレット枚数を最小にする式
制約条件 (13)　各層の個数は使用するパターンに含まれる層数と等しい制約
制約条件 (14)　自然数制約

6　荷物の重ね合わせの判定

全ての荷物について，上面を覆うふたがないために荷物を上に載せられるかどうかを調べる必要がある．そこで，1 次元パッキングでパレットパターンを求めた後に，層をどのような順番で下から上に載せれば全て載せきることができるかを考察した．つまり，2 次元パッキングで求めた層単位で，重ね合わせの判定を行う．

層を載せる順番を求めるには，有向ハミルトン閉路問題として以下のように定式化した．ハミルトン閉路問題とは与えられたグラフについて，全ての頂点を丁度一回だけ通る閉路が存在するか否かを求める問題である [2]．ただし，今回は閉路を求める必要はなく，始点と終点が一致するという条件は不必要である．このような問題をハミルトン路問題といい，本来はその問題として解く必要があるが，全頂点とつながる辺を持つ頂点を一つ加えることでハミルトン路問題はハミルトン閉路問題に帰着する．二層間で重ね合わせられるか否かの定数とし a_{ij} を用いる．a_{ij} の 2 値の判定は手動で行った．

6.1　記号の定義

添字
　V：層の集合
　S：V の真部分集合
定数

$$a_{ij} = \begin{cases} 1 & \text{層 } i \text{ の上に層 } j \text{ を載せることができる} \\ 0 & \text{層 } i \text{ の上に層 } j \text{ を載せることができない} \end{cases}$$

変数

$$x_{ij} = \begin{cases} 1 & \text{層 } i \text{ の上に層 } j \text{ を載せる} \\ 0 & \text{層 } i \text{ の上に層 } j \text{ を載せない} \end{cases}$$

6.2　定式化

制約条件

$$\sum_{j \in V - \{i\}} x_{ij} = 1 \qquad (i \in V) \tag{15}$$

$$\sum_{i \in V - \{j\}} x_{ij} = 1 \qquad (j \in V) \tag{16}$$

$$\sum_{i \in S} \sum_{j \in V - S} x_{ij} \geq 1 \qquad (\emptyset \neq \forall S \subset V) \tag{17}$$

$$x_{ij} \leq a_{ij} \qquad (i, j \in V) \tag{18}$$

$$x_{ij} \in \{0, 1\} \qquad (i, j \in V) \tag{19}$$

制約条件 (15)　各層の上に，ある層が一つ存在する制約
制約条件 (16)　各層の下に，ある層が一つ存在する制約
制約条件 (17)　部分巡回路をなくす制約
制約条件 (18)　層を重ねることができるか否かの制約
制約条件 (19)　バイナリー変数制約

7　パレットの配置場所について

最後に，荷物を詰めた各パレットのトラックへの配置場所について考察する．パレットからの荷物のはみ出しや，パレットの幅より大きな荷物も存在するため，層の幅のみに着目した 1 次元パッキングを行う．パレットを並べた列が 4 本以内なら 1 台のトラックに収まることになる．積み込み方法は 4.3 で提案した 2 次元のレベル法を 1 次元に拡張したものを用いる．パレットを並べる列をラインと呼び，各ラインでトラックの先頭に来るパレットはそのラインを規定することになる．パレットの大きさについては各パレットに存在する層の中で最も大きな幅を持つ層の幅をそのパレットの幅とする．

7.1　記号の定義

添字
　B：パレットの集合
定数
　W：トラックの容積（幅）
　w_b：パレット b の幅
　m ：パレットの数
変数

$$q_p = \begin{cases} 1 & \text{パレット } p \text{ がライン } p \text{ を規定する} \\ 0 & \text{パレット } p \text{ がライン } p \text{ を規定しない} \end{cases} \quad p \in B$$

$$z_{bp} = \begin{cases} 1 & \text{パレット } b \text{ がライン } p \text{ にある} \quad p \in B \backslash \{m\} \\ 0 & \text{パレット } b \text{ がライン } p \text{ にない} \quad p < b \end{cases}$$

7.2　定式化

制約条件

$$\sum_{p=1}^{m} q_p \leq 4 \tag{20}$$

$$\sum_{p=1}^{b-1} z_{bp} + q_b = 1 \qquad (b = 1, \cdots, m) \tag{21}$$

$$\sum_{b=p+1}^{m} w_b z_{bp} \leq (W - w_p) q_p \qquad (p = 1, \cdots, m-1) \tag{22}$$

$$q_p,\, z_{bp} \in \{0, 1\} \qquad \forall b, p \tag{23}$$

制約条件 (20)　ラインの本数を 4 本以下にする制約
制約条件 (21)　各パレットがいずれかのラインに入る制約
制約条件 (22)　各ラインに入るパレットの幅の総和がトラックの容積（幅）以下である制約
制約条件 (23)　バイナリー変数制約

8　まとめ

8.1　実行結果

表 1 ：荷物の割り当て結果
（荷物の大きさの単位は mm）

	トラックの便数			14	13
	制約条件 (2)			α =0.8	α =0.4
	制約条件 (3)			β =36692638	β =134899764
荷物	幅	奥行き	高さ	荷物数	
	870	330	910	2	4
	1,195	500	660	4	4
	1,000	400	660	6	5
	860	330	660	16	16
	860	330	630	11	12
	1,000	400	530	20	23
	2,200	500	520	5	4
	600	400	510	5	5
	1,470	480	460	1	0
	600	500	360	3	3
	1,500	500	350	4	5
	1,200	930	350	3	4
	1,180	500	330	9	14
	765	450	320	29	31
	1,500	1,000	300	3	4
	590	345	300	1	0
	503	335	288	18	20
	670	335	288	13	15
	1,020	495	260	18	14

トラックの便数			14	13
幅	奥行き	高さ	荷物数	
590	345	260	4	3
503	335	241	13	15
670	335	241	6	6
840	490	225	7	11
628	388	225	6	6
1,020	495	210	2	2
503	335	195	27	28
670	335	195	10	15
628	388	185	0	0
670	335	149	1	0
503	335	149	2	3
335	335	149	2	1
370	300	130	1	1
368	149	130	4	3
610	370	125	3	3
503	335	103	5	3
335	335	103	15	16
335	168	103	0	1

表 2 ：積み込み結果

トラックの便数	2 次元パッキングの層数		1 次元パッキングのパレット枚数
	改善策適用前	改善策適用後	
14	80	78	29
13	89	86	30

8.2　考察

トラックの便数を現状の 14 便，1 便減らした 13 便を使用するとそれぞれ仮定したうえで荷物を割り当て，積み込みについて分析した．1 次元パッキングの際の荷物を載せられる高さの上限を 1,070(mm) とした．荷物の重ね合わせについては両仮定ともに全パレットにおいてパレットパターンを全て載せることができ，全てのパレットをトラックに載せることができるという結果を得た．

9　おわりに

本研究ではトラックへの荷物の効率的な積み込みを分析し，トラックの使用便数の削減を目標に進めた．荷物の割り当て後，ある 1 便の荷物データを対象に分析を行っただけで全てのトラックに荷物を載せられるとは限らず，また毎日同じ荷物を扱うわけではないため効率的な積み込みについて更なる分析が必要がある．更に，我々が提案した数理モデルを用いて企業の現場で実践することで現状との比較を行い，実際に企業で取り入れられている手法を取り入れ，企業で実用化されるようなシステムの構築が課題である．

参考文献

[1] Andrea Lodi, Silvano Martello, Michele Monaci: Two-dimensional packing problems: A survey, *European Journal of Operational Research*, vol.141, pp.241-244 (2002)

[2] 近山隆：アルゴリズム設計 講義資料 2009，pp.21-22 http://www.logos.t.u-tokyo.ac.jp/www/home/chik/algorithm-design/09%20Graph%20Algorithms.pdf

[3] 梅谷俊治，柳浦睦憲，茨木俊秀：カッティングストック問題に対する線形計画法に基づく局所探索法の提案，数理解析研究所講究録，1297 巻，pp.116 (2002)