平成１８年度　海運ロジスティクス専攻　修士論文要旨
自動車専用船における配置割当て問題に関する研究
学籍番号　0555015　氏名　若林　篤　　指導教官　鶴田三郎　黒川久幸

1.はじめに

1.1 研究背景

近年,日米欧の成熟市場だけではなく新興市場の出現によって,完成車の海上輸送に変化をもたらすことになった。先進国間の拠点から拠点という大量輸送中心であった完成車海上輸送は域内輸送を含めた多極間,小ロット輸送へと変化することになった。この為,拠点から拠点への単純なピストン輸送ではなく,数カ所の港から数カ所の港へ自動車を積む必要性が生まれてくる為に,船の構造を変更するだけでなく,今までのように単純に自動車を積み込む作業ではなく,いかにして積載率を高め,効率よく積付け,積降ろし作業の時間を短縮することが必要となってくる。

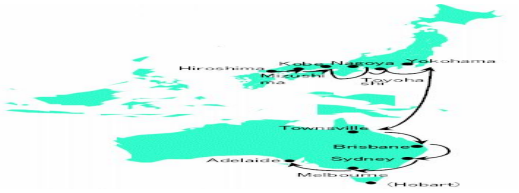


図１　航路の一例

自動車産業からの視点で見ると,日本だけで生産されるのではなく,年をおうごとに様々な地域で工場が建設され年々,海外に工場を増やす計画にシフトしていることが以下の表 2 から,日本メーカーの海外生産シフトしていることが分かる。これは,一つの工場に車種を限定することによる集中生産を行い,生産性を高めるほかに,ロジスティクスの視点からも製品を集中管理することにより保管費などが安くすることができるからである。このため,ある工場では車種が決まっているために,生産されていない車種に関しては,輸入しなければならない。このため多港積み・多港下ろしが存在することになる。

表２　海外工場の一例

ブラジル	Toyota do Brasil Ltda.	●	1959	カローラ
アルゼンチン	Toyota Argentina S.A.	●	1997	ハイラックス
イギリス	Toyota Motor Manufacturing (UK) Ltd.	●	1992	アベンシス、カローラ、エンジン
フランス	Toyota Motor Manufacturing France S.A.S.	●	2001	ヤリス

1.2 研究目的

わが国での先行研究では,2003 年 OR 学会の斉藤らが発表した１件のみである。自動車の積み込み・下ろし,入替え時間の最小化を目的としたローカルサーチを用いた近似解法で行っている。[4]しかし,自動車の配置場所によって,船体の傾きが変化し安全に輸送活動を行う為には,船にかかるモーメントを考慮する必要があるが考慮されていないと思われる。そこで，安全に輸送活動を行うためにモーメントを考慮した,自動車専用船における配置割当ての支援を行うことが目的である。まず,自動車専用船における配置割当て問題の定式化を行いモデリング言語によるプログラムの実装を行う。しかし,数理計画法は問題の規模が大きくなるほど求解時間が長くなることが一般的に知られている。そのため,定式化を行い,プログラム実装を行うだけではなく,実験を通して,実務時間内で求解するには,どのようにパラメータを与えればよいのかを示す。

2. 自動車専用船について

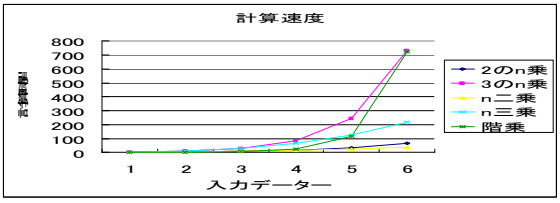
2..1　自動車専用船の特徴

自動車専用船では以下のことが知られている。

・１隻で 4000～6000 台積載可能。
・大体 10 階層で,40 区画程度に分けられている。
・階層によって高さが異なるため,階層よりも背高の品種は,その場所に配置することができない。
・配置場所によって船体の傾きが変化する。
・複数積み込み・下ろし港が存在するため，自動車の配置場所によって船体の傾きが変化するため安全に輸送活動を行うための配置場所は積み込み・下ろし時間,入替え時間が非常にかかる。

2..2　自動車専用船における配置割当てに対しての定式化の留意点

2.1 の情報を考慮し自動車専用船における配置割当て問題に対して２種類の定式化を行った。非線形関数を用いた定式化と線形関数を用いた二段階による定式化である。非線形関数を用いた数理計画法は,変数×変数では求解時間がかかることが一般的に知られている(グラフ 3)。積み込み・下ろし時間と,入替え時間を一つの定式化で表現すると非線形関数となるため求解時間がかかってしまう。そのため今回は線形関数を用いた二段階による定式化の実験を行った。まず,第一段階では積み込み・下ろし時間の最小化の配置場所を決定する。第二段階では,この第一段階で得られた,配置場所情報のデータを使用することによって,入替え時間と積み込み・下ろし時間を考慮した定式化ができる。３章では,第一段階の定式化と第二段階の定式化を示す。



グラフ 3 非線形関数による計算量爆発の図

3.定式化

3. 1 段階の定式化

集合とパラメータ

・*Node* : 各点(港)の集合

・*Route* : 点*i*の前の点*k*と次の点*j*の集合

・*Edge* : 枝の集合

・*DummyNode*1: 閉路を避けるダミーノード

・*DummyNode*2: 閉路を避けるダミーノード

・*ShipInvClass* : 船の配置場所

・$Loading_{i,j,zz}$: $Edge \times ShipInvClass$:
積み込みの集合

・$UnLoading_{zz,i,j}$: $ShipInvClass \times ShipInvClass$:
積み下ろしの集合

・$LaU_{i,j,zz}$: $Loading_{i,j,zz} \cup UnLoading_{zz,i,j}$

・$Load1_{i,j}$: $Node \cup DummyNode1$

・$Load2_{i,j}$: $Node \cup DummyNode2$

・Commodity : 品種の集合

・*Height* : Commodity > *ShipHeight*
*ShipInvClass*より背高の高い品種の 集合

・$Origin_{i,c}$: 点*i*における品種*c*の需要の発生点

・$Destination_{i,c}$: 点*i*における品種*c*の需要の集中点

・$ShipInv_{zz}$: 船における各クラスの 在庫上限

・$Shipheight_{zz}$: 船の各クラスの高さ

・*ShipWT* : 船の自重

・WT_c : 各品種*c*の重さ

・$height_c$: 品種*c*の高さ

・$longitudial_{zz}$: 各クラスの縦の長さ

・*TrimMUpper* : 各クラスの縦の梃子の 上限

・*TrimMLower* : 各クラスの縦の梃子の 下限

$time_{zz}$: 積み込み，積み下ろし作業にかかる時間

変数

$x_{i,j,zz} \geq 0$　各点の積み込み，積 み下ろしの数

$z_{i,j,zz,c} \geq 0$　各点同士の流入出量 関係

目的関数と制約条件

最小化

Minimize

$$\sum_{zz \in ShipInvCla\ \ ss} \sum_{c \in \mathrm{Commodity}} \sum_{(i,j) \in Edge} (x_{i,j,zz,c} + x_{zz,i,j,c}) time_{zz}$$

制約条件

$$Origin_{j,c} = \sum_{zz \in ShipInvCla\ \ ss} \sum_{(i,j) \in Edge} x_{i,j,zz,c} \qquad \forall j \in Node$$
$$c \in \mathrm{Commodity} \qquad (1)$$

$$Destinatio\ n_{j,c} = \sum_{zz \in ShipInvCla\ \ ss} \sum_{(i,j) \in Edge} x_{zz,i,j,c}$$
$$\forall j \in Node$$
$$c \in \mathrm{Commodity} \qquad (2)$$

$$z_{k,i,zz,c} + x_{k,i,zz,c} = z_{i,j,zz,c} + x_{zz,i,j,c}$$
$$\forall (k,i,j) \in Route$$
$$zz \in ShipInvClass$$
$$c \in \mathrm{Commodity} \qquad (3)$$

$$z_{i,j,zz,c} = 0 \qquad \forall (i,j) \in Load1$$
$$zz \in ShipInvClass$$
$$c \in \mathrm{Commodity} \qquad (4)$$

$$z_{zz,i,j,c} = 0 \qquad \forall (i,j) \in Load1$$
$$zz \in ShipInvCla\ \ ss$$
$$c \in \mathrm{Commodity} \qquad (5)$$

$$z_{i,j,zz,c} = 0 \qquad \forall zz \in ShipInvCla\ \ ss$$
$$(zz,c) \in Height \qquad (6)$$

$$\sum_{c \in \mathrm{Commodity}} z_{i,j,zz,c} \leq ShipInv_{zz} \qquad \forall (i,j) \in Edge$$
$$zz \in ShipInvCla\ \ ss \qquad (7)$$

$$\sum_{c \in \mathrm{Commodity}} z_{zz,i,j,c} \leq ShipInv_{zz} \qquad \forall (i,j) \in Edge$$
$$zz \in ShipInvClass \qquad (8)$$

$$\frac{\left[\left(\sum_{c \in Commodity} \sum_{(i,j) \in Edge} WT_c z_{i,j,zz,c} longitudia\ \ l_{zz}\right)\right]}{\sum_{c \in Commodity} z_{i,j,c} WT_c + \mathrm{ShipWT}} \leq TrimMUpper$$
$$\forall (i,j) \in Edge \qquad (9)$$

$$\frac{\left[\left(\sum_{c \in Commodity} \sum_{(i,j) \in Edge} WT_c z_{i,j,zz,c} longitudia\ \ l_{zz}\right)\right]}{\sum_{c \in Commodity} z_{i,j,c} WT_c + \mathrm{ShipWT}} \leq TrimMLower$$
$$\forall (i,j) \in Edge \qquad (10)$$

$$\frac{\left(\sum_{c \in Commodity} \sum_{zz \in ShipInvClass} z_{i,j} WT_c altitude_{zz} + GravityCenterShipWT\right)}{\sum_{c \in Commodity} \sum_{zz \in ShipInvClass} z_{i,j,zz,c} WT_c + ShipWT} \leq AlMUpper$$
$$\forall (i,j) \in Edge \qquad (11)$$

$$\frac{\left(\sum_{c \in Commodity} \sum_{zz \in ShipInvClass} z_{i,j} WT_c altitude_{zz} + GravityCenterShipWT\right)}{\sum_{c \in Commodity} \sum_{zz \in ShipInvClass} z_{i,j,zz,c} WT_c + ShipWT} \geq AlMLower$$
$$\forall (i,j) \in Edge \qquad (12)$$

式(1)では，発地 i における品種 c の量は,発地 c における各クラスに割当てられた各品種の総和と同じ。

式(2)では，着地 i における品種 c の量は,着地 i における各クラスに割当てられた各品種 c の総和と同じ。

式(3)では，フロー整合条件である。点 i において,以下のように表現できる。

左辺の $z_{k,i,zz,c}$ が,前の点 k から点 i にくる流入量であり,$x_{j,i,zz,c}$ は点 i における積込み量である。また右辺の $z_{i,j,zz,c}$ は,点 i から次の点 j への流出量であり $x_{zz,i,j,c}$ は点 i における積み下ろし量である。

式(4),(5)始点の前,終点の後ろに閉路を避ける為にダミー点を加える。
式(6)では，クラスより背高の高い品種は置くことが出来ないことを意味する。

式(7),(8)では，割り当てられた数ShipInvClass の上限以下であることを意味する。
式(9),(10)では，モーメントの縦の上下限に対して，バラスト水槽に水を注入して値を変えることのできる，梃子の長さの上下限値である。各クラスに配置された，品種 c の量と重さによって，モーメント値が変わる。そのため，安全に輸送するにはモーメントの上限を超えてはならない。そのために，バラスト水槽に水を注入して値を変えて，梃子の長さを変化させることの出来る上下限の制約を計算している。
式(11),(12)では，上下方向のモーメントの上下限に対して，バラスト水槽に水を注入して値を変えることのできる,梃子の長さの上下限値である。しかし，縦横の場合と異なり，上下方向を考えるときは，船の場合は重心地と，実際の船にかかる重心地とは異なるので，実際にかかる重心地と船の中心地との距離の調節できる上下限の制約を計算している。これによって積み込み,積み下ろし時間の最小化の配置割当てが決定する。この配置結果を用いて第二段階で入替えを考慮した定式化を行うことができる。

3.2 二段階の定式化

第二段階の定式化は第一段階の定式化を式の追加や拡張を行うだけである。

新たな集合を追加

- $Path1_{zz}$: 積み込み時に,ある品種cがある*ShipInvClass*に配置するときに通過する集合
- $Path2_{zz}$: 積み下ろし時に,ある品種cがある*ShipInvClass*に配置するときに通過する集合

新たなパラメータの追加

- $reshipmnettime1_s$: 積み込み時に割当てられた場所までに入替えがあった場合の入替え時間
- $reshipmnettime2_s$: 積み込み時に割当てられた場所までに入替えがあった場合の入替え時間
- $y1_{i,j,zz}$: 積み込み時における 0 -1定数
- $y2_{zz,i,j}$: 積み下ろし時における 0 – 1定数

新たに変数の追加

- $xyz_{i,j,zz} \geq 0$, integer : 入替えが発生する量

また,第一段階では変数は全て実数であったが第二段階では全て整数となる。
目的関数の拡張

$$\begin{aligned} &Minimize \\ &\sum_{zz \in ShipInvCla} \sum_{ss\ c \in \mathrm{Commodity}} \sum_{(i,j) \in Edge} (x_{i,j,zz,c} + x_{zz,i,j,c})time_{zz} \\ &+ \sum_{(i,j) \in Edge} \sum_{zz \in ShipInvCla} xyz_{ss\ i,j,zz} \\ &+ \sum_{(i,j) \in Edge} \sum_{zz \in ShipInvCla} xyz_{ss\ zz,i,j} \end{aligned}$$

制約式の追加

$$\sum_{s \in Path1_{zz}} \sum_{c \in \mathrm{Commodity}} y1_{i,j,zz}\, reshipment\_itime1_s\, x_{i,j,zz,c} = xyz_{i,j,zz}$$

$$\forall (i,j) \in Edge,\ zz \in ShipInvCla\ \ ss \qquad (13)$$

$$\sum_{s \in Path1_{zz}} \sum_{c \in \mathrm{Commodity}} y2_{zz,i,j}\, reshipment\_itime1_s\, x_{zz,i,j,c} = xyz_{i,j,zz}$$

$$\forall (i,j) \in Edge,\ zz \in ShipInvCla\ \ ss \qquad (14)$$

式(13),(14)は第一段階で得た値を枝間 (i, j) で使用された *ShipInvClass* を $y1_{i,j,s}$, $y2_{zz,i,j}$ によって使用した場合は 1，使用しない場合は 0 と表現し，積み込み，積み下ろし時において品種が流れるときに入れ替えが発生したときの量を計算する式である。

4.数値実験

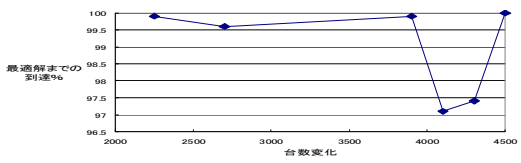
4.1 実験環境と数値実験の概要

今回は glpk4.8 を用いてプログラムに実装し，実験を行う。計算機環境は CPU が 3.20GHz，メモリ 1.00GB である。実験の検証について台数,品種の変化による最適解までの到達%と求解時間の差異を検証する。求解時間の設定を 2 時間とした。

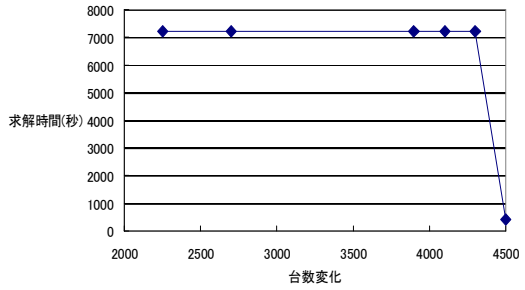
数値実験 1：品種を固定し,台数の増加によって上記のことを検証する。
数値実験 2：台数を固定し,品種を増加し上記のことを検証する。
数値実験 3：総量,品種も増加によって上記のことを検証する。
数値実験 4：0―1 定数の範囲を拡張し数値実験 1～3 を行い上記のことを検証する。

4.2 数値実験 1

品種を 16 と固定し総量 2250～4500 台と変化させた。グラフ 4 では最適解の到達%を示す。グラフ 5 では求解時間の比較である。



グラフ 4　最適解までの到達%の比較

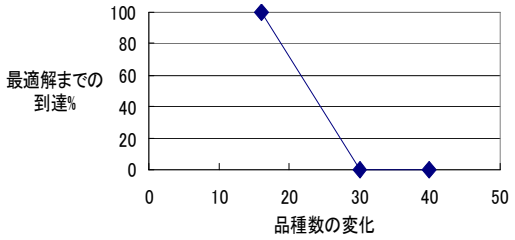


グラフ 5　求解時間の比較

台数増加による求解困難とはいえそうにない結果が得られた。

4.3 数値実験 2

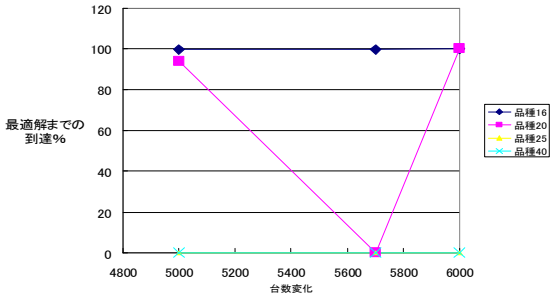
総量を 4500 台と固定し,品種を 16,30,40 と増加させる。グラフ 6 では最適解までの到達%を示すが品種 16 以外は時間内に求解できなかった。このことから,台数増加より,品種の増加が求解困難になるのではないかと思われる結果になった。



グラフ 6 最適解までの到達%の比較

4.4 数値実験 3

総量を 5000,5700,6000 台とし,品種を 20,25,40 とした。グラフ 7 から分かるように品種 16 のときは全て求解できたのに対し,品種の増加によって求解できなくなっていることが分かる。



グラフ 7　最適解までの到達%の比較

4.5 数値実験 4

次に,少しパラメータの $y1_{i,j,s}$, $y2_{zz,i,j}$ の与える範囲を拡張し数値実験 1～3 を再び行う。解に差異はないが以下の表 8 ように時間内に求解できなかった問題も求解可能になった。

表 8　数値実験 4 結果一覧

予約考慮せず

数値実験1

台数	求解時間	使用メモリ(M)	最適解までの到達%
2250	7200	54	99.9
2700	7200	46.9	99.6
3900	7200	63.1	99.9
4100	7200	63	97.1
4300	7200	42.5	97.4
4500	41.9	41.6	100

数値実験2

品種	求解時間	使用メモリ(M)	最適解までの到達%
品種16	41.9	41.6	100
品種30	7200	81.6	0
品種40	7200	105.6	0

数値実験3

品種16

台数変化	求解時間	使用メモリ(M)	最適解までの到達%
5000	7200	46.6	99.9
5700	7200	50.4	99.9
6000	21.0	41.5	100

品種20

台数変化	求解時間	使用メモリ(M)	最適解までの到達%
5000	7200	93.2	94.2
5700	7200	51.8	0
6000	814.2	57.7	100

品種25

台数変化	求解時間	使用メモリ(M)	最適解までの到達%
5000	7200	70.6	0
5700	7200	35	0
6000	7200	63.6	0

品種40

台数変化	求解時間	使用メモリ(M)	最適解までの到達%
5000	7200	110	0
5700	7200	107.3	0
6000	7200	105.5	0

予約考慮

数値実験1

台数	求解時間	使用メモリ(M)	最適解までの到達%
2250	7200	52.4	99.9
2700	7200	47.1	99.7
3900	7200	58.6	99.9
4100	7200	100.4	99.5
4300	7200	44.8	98.8
4500	7200	46.4	99.3

数値実験2

品種	求解時間	使用メモリ(M)	最適解までの到達%
品種16	7200	54	99.9
品種30	7200	97.9	99.9
品種40	7200	106.4	0

数値実験3

品種16

台数	求解時間	使用メモリ(M)	最適解までの到達%
5000	7200	48	99.9
5700	7200	51.7	99.9
6000	349.4	42.1	100

品種20

台数	求解時間	使用メモリ(M)	最適解までの到達%
5000	7200	102.6	99
5700	7200	54.4	96.8
6000	2477	52.8	100

品種25

台数	求解時間	使用メモリ(M)	最適解までの到達%
5000	7200	71.5	88.6
5700	7200	65.8	0
6000	7200	63.6	0

品種40

台数	求解時間	使用メモリ(M)	最適解までの到達%
5000	7200	108.2	96.9
5700	7200	109.7	97.1
6000	7200	106.4	0

5.考察

設定時間内で,ほぼ 100%の解を得ることに成功した。100%の求解ができなかったのは,船舶特有のモーメントに関する制約条件が厳しく,実行可能領域が,非常に狭いなかで最適化を行っていると推測される。実際に,梃子の値を緩和し実験をすると求解時間が速いことが仮実験で得られた。

6.終わりに

自動車専用船における配置割当ての定式化,実務時間内に求解することに成功した。台数による増加よりも品種数の増加のほうが求解することが困難であることを明らかにした。品種を 1 種類として台数を m,配置場所を n,積み込み港,積み下ろし港を x とすると，計算量 1 は以下のように表現できる。計算量 1＝mnx となる。次に品種を複数とし c として,台数を m と単純に表現することが出来ない(台数の合計となる)つまり，計算量 2 =cmnx と表現できない。この cm の部分が,計算量が多く,計算量が爆発的に増え,そのため時間内に求解することができないと推測される。

7.参考文献

(1)特許庁 HPhttp://www.jpo.go.jp/indexj.htm

(2)トヨタ自動車株式式会社の HP

http://www.toyota.co.jp/jp/facilities/manufacturing/worldwide.html

(3) 日本船主協会の HP

http://www.jsanet.or.jp/newship/html/11car/026.html

(4) 海難審判庁の HP

http://www.mlit.go.jp/maia/04saiketsu/16nen/yokohama/yh1602/14yh119yaku.htm

引用文献

(1)　船の豆知識

http://www.h.do-up.com/home/jhizumi/index.html