

マルチコンテナターミナルにおけるバース割当問題*
Efficient berth allocation at a multi-user container terminal*

孫しん**
Xin SUN**

1. 研究の目的と背景

バース管理の目的のために、全体の岸壁をいくつかのバースに分割し、適切な岸壁ロケーションに船を割り当てるのはバース単位で行っている研究がある。このような研究では多くのバース割当法が各バースに同時に1隻しか係留できないことを仮定している。しかしながら、現実には2隻の船が1つのバースに同時期に係留される場合が多く存在しており、1隻のみの係留はあまり現実的ではない。なぜならば、これでは物理的なバース空間に無駄が出てくる可能性が高いといえる。既往の研究でそのような方法をとった理由は割り当て計算での計算の効率化のために、このようなバース単位の割り当てを考えざるを得なかったからである。

本研究では、港湾管理者を主体として、利用船社への配慮も考慮したコンテナ船のバースへの効率的な割当方法を提案する。バース割当の評価は、バースの公共性の点から総在港時間（つまり各船のバース待ち時間と荷役時間の合計の総和）の最小化が適切であると考えられる。

2. モデルの定式化

(1) 前提条件

- a. 各船は必ずいずれかのバースに1回係留される。
- b. 当該船の係留バースにおいて、当該船の全長に係留の際必要となる延長を考慮した必要延長と、他船の必要延長の和が当該バース岸壁長を超えなければ、この2隻船は1つのバースに同時期に係留されること。
- c. それぞれのバースでの各々の船の荷役時間は既知である。
- c. 各々の船の荷役時間は、バースによって異なる。

* キーワード：バース割当、ターミナル計画、港湾計画

** 学生会員、工修、神戸大学自然科学研究科海事科学専攻

(658-0022 神戸市東灘区深江南町5-1-1, TEL:078-431

-6591, E-mail: sunxin@maritime.kobe-u.ac.jp)

(2) 定式化

目的関数：

$$\text{Min} \sum_{j \in V} \left\{ \sum_{i \in B} \left(m_{ij} + \sum_{t \in U} x_{ijt} \right) - A_j \right\} \quad (1)$$

制約条件：

$$\sum_{i \in B} y_{ij} = 1 \quad \forall j \in V \quad (2)$$

$$\sum_{i \in B} m_{ij} \geq A_j \quad \forall j \in V \quad (3)$$

$$\sum_{j \in V} L_j x_{ijt} \leq W_i \quad \forall i \in B, t \in U \quad (4)$$

$$m_{ij} \leq t x_{ijt} \leq m_{ij} + C_{ij} y_{ij} \quad \forall j \in V, i \in B, t \in U \quad (5)$$

$$x_{ijt} \in \{0,1\} \quad \forall j \in V, i \in B, t \in U \quad (6)$$

$$y_{ij} \in \{0,1\} \quad \forall j \in V, i \in B \quad (7)$$

$$m_{ij} \geq 0 \quad \forall j \in V, i \in B \quad (8)$$

i : バース

j : 対象船

t : 単位時間

B : 全体のバースの集合

V : 全体船の集合

U : 単位時間の集合

A_j : 船 j の到着時間

C_{ij} : 船 j がバース i での荷役時間

x_{ijt} : 船 j がバース i での荷役時間 t であれば1、

その他の場合は0

y_{ij} : 船 j がバース i で係留すれば 1、その他の場合は 0

m_{ij} : 船 j がバース i での係留開始時刻

L_j : 船 j の延長

W_i : バース i の延長

式 (1) が目的関数総在港時間を表しています。式 (2) 船 j が必ずバースに係留することを意味しています。式 (3) は船 j が入港してから、係留されることを保証しています。式 (4) はあるバースに 2 隻の船が係留されるとき、船の延長の和が、バースの岸壁長を超えないことを保証します。式 (5) は係留時刻、荷役時間、出港時刻の関係を表した。

3. 遺伝的アルゴリズムを用いた解法

(1) 全体のフロー

本問題は線形の問題として定式化されており、また制約が多く最適解を得ることが困難であるので、遺伝的アルゴリズム (GA) を用いた近似解法で検討する。

GA というのは、自然界における生物の進化モデルである。すなわち、世代を形成している個体の集合の中で、環境への適応度の高い個体が次世代によって多く生き残り、また交叉および突然変異を起こしながら次の世代を形成していく過程を模した最適化手法である。ここで本問題の目的関数を適応度に、解の候補を個体にそれぞれ対応させる。おおまかな流れを以下に示す。

ステップ 1 : 初期の個体群を生成させる。

ステップ 2 : 目的関数値を求め適応度に変換する。

ステップ 3 : 1 世代目であれば、ステップ 4 へ。そうでなければ、親の個体群と子の個体群の中で同じ型の一方を省き、残った個体の中で適応度が上位のものを新たな親個体群とする。現世代が最終世代であれば、終了。

ステップ 4 : 遺伝演算 (交叉、突然変異) を行い、新たな個体を生成させ、ステップ 2 へ。

(2) GA パラメータ

個体群サイズを 20、30、40 および 50 の 4 ケース用意し、突然変異率は 0.01 ~ 0.1 の間で 0.01 刻みで変化させ、それらを組み合わせて合計 $4 \times 10 = 40$ ケースにおいて各ケース 5 個の問題とコロンボ港の実際のデータで実験を行い、GA パラメータの推定を行った。

どの個体群サイズ、どの突然変異率のとき最善解が得られるかを一意に決定することはできないが、個体群サ

イズについては目的関数値を調べると、個体数が少ないと解は良くなり、多いと計算時間がかかることがわかったので、本研究では 40 個に設定した。突然変異率については、変異率 0.08 のときに比較的良い解が得られた。

解の収束状況については、世代交代数は目的関数値を比較したところ、20000 世代より少ないと値がまだ収束しておらず、多いと収束が終わっており、ほとんど値が良くなっていなかった。また世代数が多いと計算時間がかかることから、20000 世代に設定した。

したがって本研究で用いる GA パラメータは、個体群サイズ 40、世代数 20000、突然変異率 0.08 と設定し、次章でコロンボ港、神戸、イエメン港、シンガポール港の実際データでバース割当実験を行う。

4. 実績データによる計算

コロンボ港 Jaya コンテナターミナルに 2003 年 6 月 12 日 ~ 6 月 21 日の 10 日間に寄港したコンテナ船 61 隻の入港時刻データおよび荷役時間データ及び、神戸、イエメン港、シンガポール港の 10 日間分の実績データを分析する。ここでは 10 日間を 1 スパンとするケース、5 日間を 1 スパンとし合計 2 スパンとするケース、3 日間を 1 スパンとするケース (計 3 スパン) の 3 ケースの問題規模で合計の評価値がどのように変化するかを見る。

実験結果 :

港	期間	船の数	バース数	総在港時間(日)
イエメン	10 日	7 隻	2	1.53
コロンボ	10 日	61 隻	5	57.38
神戸	10 日	104 隻	7	258.96
シンガポール(B)	10 日	70 隻	8	190.03

1 隻あたりの在港時間 :

港	船の数	バース数	1 隻あたり在港時間(日)
イエメン	7 隻	2	0.22
コロンボ	61 隻	5	0.94
神戸	104 隻	7	2.49
シンガポール(B)	70 隻	8	2.71

5. 結論

本研究では、マルチユーザーターミナルにおけるバース割当を行う際、得られた結果から明らかになったことを

以下に示す。

- a. バース数が多いと当然のことながらターミナル利用時間、隻数、実際に生じた待ち時間は短くなる傾向にある。
- b. 複数スパンに区切った場合、多くのスパンに区切る方が良い結果を得られる。

以上のことから、提案した方法を使ってマルチユーザターミナルの運用方法の改善を行えば、実質的なコストの削減につながる上、同程度の生産性と経済規模を得るためのコンテナ荷役処理ができる。