



An Extension of the Berth Allocation Problem to the Berth Template Problem

Imai, Akio

(Citation)

Journal of maritime researches, 5:35-59

(Issue Date)

2015-06

(Resource Type)

departmental bulletin paper

(Version)

Version of Record

(JaLCD0I)

<https://doi.org/10.24546/81009086>

(URL)

<https://hdl.handle.net/20.500.14094/81009086>



AN EXTENSION OF THE BERTH ALLCATION PROBLEM TO THE BERTH TEMPLATE PROBLEM

Akio IMAI^{*,**}

ABSTRACT

A marine container terminal operator may have a situation with excessive calling requests to be served especially when some new service contracts are under consideration. For this situation, we propose a strategic berth template problem (BTPS) that selects the ships among the requesting ones to be served and arranges their berth-windows within a limited planning horizon. The BTPS employs the subgradient optimization procedure, which is the one the author developed for the operational berth allocation problem. A wide variety of numerical experiments indicate the subgradient procedure works well for the BTPS.

Keywords: Berth template problem; Berth allocation problem; Subgradient optimization; Lagrangian relaxation; Container terminal; Heuristics

* Professor, Graduate School of Maritime Sciences, Kobe University. 5-1-1 Fukaeminamimachi, Higashinada, Kobe, Hyogo, 658-0022, Japan. Email : imai@maritime.kobe-u.ac.jp,

**Researcher, Port and Airport Research Institute, Nagase, Yokosuka 239-0826 Japan

1. INTRODUCTION

Sea-borne container shipping plays an important role in the world transportation system and the global supply chain these days. A marine container terminal acts as an interchange of the different modes of sea and land involved in the overall transportation process. Therefore, improvements in the efficiency and productivity of terminal operations are crucial in reducing the costs and the overall trip duration and thus have been gaining more attention lately.

The primary aim of a terminal is a quick turnaround or a secured departure deadline of calling ships. Major container ports feature the so-called “multi-user container terminals (MUTs)” that serve a lot of calling ships of different shipping lines with a long quay and vast yard space to provide a large ship handling capacity. In an era of cost-cutting and competition, shipping lines are less inclined, without enormous cargo demands, to operate private terminals than they used to be (Mongelluzzo, 2013). In this background, the need to operate MUTs more efficiently as well as the issues pertaining to the efficient berth scheduling at an MUT have been receiving more attention these days.

Most decision makings can be classified as three broad categories: strategic, tactical and operational. As far as the berth scheduling is concerned, the existing literature may fall into two categories in a relative sense: long-term (tactical) and short-term (operational). Tactical berth scheduling (or berth template problem, BTP) finds an optimal set of berth-windows (i.e., berthing locations with the start and end times for service) within the fixed length of planning horizon, while operational berth scheduling (or berth allocation problem, BAP) finds an optimal set of berth-windows within an open-ended planning horizon. Most of the existing papers about berthing decision fall into the category of operational scheduling. Very few are in the tactical scheduling.

For a terminal operator, the service contracts with shipping lines are reviewed and renewed on a regular interval or whenever needed. Alternatively, the terminal operator may receive the berthing request from a new customer. The typical contract negotiation process between a shipping line and a terminal operator is illustrated by JICT (2014), a webpage of a South Asian terminal operator. Throughout the negotiation process, the operator arranges a template for berthing. As briefly depicted above, the berth template problem (BTP) in the literature determines the template for berthing, i.e., a set of berth-windows serving ships during a fixed planning horizon, given a long-term calling request profile from shipping lines. In particular, as the most notable distinction between BTP and BAP, the fixed planning horizon is repeated in a cyclic fashion in the BTP and hereafter referred to as the *cylinder*, as used in one of the pioneering works on the BTP by Moorthy and Teo (2006). In general, most container shipping services are provided weekly on a fixed day of the week, thus the BTP normally arranges berth-windows to meet all the calling requests within a week.

In most of the cases, for such a new contract or contract renewal there might be no significant change in the number of overall calling ships when updating the berth template design. This scenario leads to the situation in which all calling ship requests can

be accommodated in any berth template design. In fact, as will be reviewed in the next section, all of the existing BTP studies assume such a full coverage of calling ships. Their focuses are mainly on reducing the operational cost and/or meeting the requirements/performances of the calling ships

In contrast, under the inauguration of a new MUT or the completion of major capacity expansion at an existing MUT, the terminal operator may need to design a brand new berth template to incorporate all prospective demands of calling ships. In addition to the decision factors similar to those in the existing BTP literature, the terminal operator may face the issue of excessive demand and require a decision making methodology for determining which part of the demand to be satisfied. This scenario is not addressed in the existing BTP studies. We hereby propose a strategic level of berth scheduling: the strategic berth template problem (BTPS), which chooses ships to be served and those not to be and finds a set of berth-windows for the served ships within a pre-determined fixed length of planning horizon so as to maximize the service objective. For convenience, the berth template problem at the tactical level addressed in the existing papers is hereafter referred to as BTPT.

This paper introduces an integer programming model for the BTPS with discrete berthing locations and develops a Lagrangian relaxation-based heuristic for it. The BTPS formulation is structured based on the formulation of the dynamic BAP (or DBAP) in Imai et al. (2001) to take advantages of the established solution methodology: the subgradient procedure with Lagrangian relaxation.

2. LITERATURE REVIEW

As the issues related to efficient terminal operations have been constantly gaining importance, there have been a growing number of studies that deal with the BAP models. On the other hand, the BTP is a relatively new research topic with few research works. These two types of problems are reviewed in this section. In particular, to the author's knowledge, there is no existing BTP research work that focuses on the strategic decision of selecting the shipping lines such as the BTPS in this study.

One of the earliest works of the BAP is Imai et al. (1997) who addressed a BAP in discrete location indices (hereafter referred to as BAPD in this section) for commercial ports. Most service queues are in general processed on an FCFS (First-Come-First-Served) basis. They concluded that in order to achieve high port productivity, an optimal set of ship-to-berth assignments should be determined, instead of relying on the FCFS rule. Their study assumed a static situation where ships to be served for a planning horizon had all arrived at a port before one planned the berth allocation. Thus, their study can be applied only to tremendously busy ports. As far as container shipping is concerned, such busy ports are neither competitive nor realistic because of the long delay in the interchange process at ports. In this context, Imai et al. (2001, 2005a) extended the static version of the BAPD to a dynamic treatment that is similar to the static treatment, but with the

difference that some ships arrive while work is in progress. Due to the difficulty in finding an exact solution, they developed a heuristic by using a subgradient method with the Lagrangian relaxation. Their study assumed the same water depth for all the berths, while in practice there are berths with different water depths in certain ports. Nishimura et al. (2001) further extended the dynamic version of the BAPD for the multi-water depth configuration. They employed genetic algorithm (GA) to solve that problem. In some real situations, the terminal operator assigns different priorities to calling vessels. For instance, at a terminal in China, small feeder ships have priority, as handling work associated with them is completed in a short period of time and larger vessels do not have to wait for a long time. On the other hand, a terminal in Singapore treats large vessels with higher priority because they are good customers to the terminal. Imai et al. (2003) extended the dynamic BAPD in Imai et al. (2001, 2005a) to treat the ships with different priorities and see how the extended BAPD differentiates the handling of ship in terms of the service time associated with ships. Imai et al. (2007) proposed the BAPD with simultaneous berthing of multiple ships at the indented berth, which was potentially useful for a fast turnaround of mega-containerships. Cordeau et al. (2005) developed a tabu search heuristic for the dynamic BAP in two versions with both discrete and continuous location indexes. They analyzed the solution quality of the proposed heuristic for the discrete location with the exact solution by CPLEX; however, the applied problem cases were relatively small sized ones. Monaco and Sammarra (2007), inspired by the dynamic BAPD of Imai et al. (2001), proposed an improvement in its formulation and also developed the Lagrangian relaxation-based subgradient optimization, which was the same approach for Imai et al. (2001, 2005a) but with some modifications. Imai et al. (2001, 2005a) proposed three heuristics embedded in the subgradient procedure. Monaco and Sammarra reported that their algorithm outperformed that of Imai et al. (2001, 2005a). However, they did not mention which one of the three heuristics embedded in the subgradient procedure in Imai et al. (2001, 2005a) was used for performance comparison. Hansen et al. (2008) developed a variable neighborhood search method for the BAPD. Mauri et al. (2008) applied the Population Training Algorithm with Linear Programming to the dynamic BAPD, which was formulated in Cordeau et al. (2005). Imai et al. (2008) extended the BAPD developed in Imai et al. (2001, 2005a) for a terminal which assigned some calling ship to another terminal when the terminal was congested. Golias et al. (2009) proposed the dynamic BAPD with customer service differentiation based on respective agreements. They formulated their BAPD as a multi-objective problem and developed a GA-based heuristic. They also proposed, in Golias et al. (2010), another heuristic based on a lamda optimal. Buhrkal et al. (2011) treated the dynamic BAPD and formulated the problem as the improved heterogeneous VRP with time windows based on the discrete version of BAP of Cordeau et al. (2005). Saharidis et al. (2010) proposed a hierarchical optimization for the BAPD with two conflicting objectives terminal operators face. Xu et al. (2012) proposed the BAPD with different water depths at berths and tidal conditions. Imai et al. (2013) discussed a terminal efficiency in terms of berthing ships in

different types of innovative terminal designs by comparing the total service time of calling ships when their berth-windows are optimally scheduled with ad-hoc berth allocation problems for those different terminals in discrete location indexes. Recently some studies such as de Oliveira et al. (2012), Lalla-Ruiz et al. (2012), and Ting et al. (2014) proposed new heuristics for the dynamic BAPD that had been discussed in Imai et al. (2001), Cordeau et al. (2005), and Monaco and Sammarra (2007). All the three papers tested their heuristics with problem instances that were provided in Cordeau et al. (2005). Ting et al. (2014) indicated that their algorithm outperformed the others.

There is another type of the BAP, which is the one with a continuous location index (referred to as BAPC). In the aforementioned studies the entire terminal space is partitioned into several parts (or berths) and the allocation is planned based on the divided berth space. This may result in having some unused berthing space. Under the continuous location approach, ships are allowed to be served wherever the empty spaces are available to physically accommodate the ships via a continuous location system. This type of problem resembles more or less the cutting-stock problem where a set of commodities is packed into some boxes in an efficient manner. A ship in service at a berth can be shown by a rectangle in a time-space representation, therefore efficient berth usage is a sort of packing “ship rectangles” into a berth-time availability as a box with some limited packing scheme such as that no rotation of ship rectangles is allowed. For this type of the BAP, Lim (1998) first addressed a problem with the objective of minimizing the maximum amount of quay space used at any time with the assumption that once a ship is berthed, it will not be moved to any place else along the quay before it departs. He also assumed that every ship was berthed as soon as it arrived at the port. On the other hand, Li et al. (1998) solved the BAPC both with and without the ship’s movement restriction. Their objective is to minimize the makespan of the schedule. Park and Kim (2002) developed a subgradient procedure with Lagrangian relaxation for the BAPC. Imai et al. (2005b) addressed a BAPC, but with a major difference from the other BAPCs in that the handling time depended on the berthing location of ship. They developed a heuristic for that problem in cooperation with a heuristic for the dynamic BAPD in Imai et al. (2001, 2005a). The conclusion of their study was that the best approximate solution was identified with the best solution in discrete location where the berth length was the maximum length of ships involved in the problem. This implies that the solution in discrete location is applicable for practice in berth allocation planning and the improved solution can be obtained from the solution in discrete location. As mentioned before, Cordeau et al. (2005) developed a tabu search heuristic for the dynamic BAP in both discrete and continuous location indexes. For the continuous location version, the solution quality was assessed by comparisons with solutions by the straightforward heuristic. Lee et al. (2010) developed two greedy randomized adaptive search procedures for the BAPC. Cheong et al. (2010) dealt with BAPC with multi-objectives of makespan, waiting time and degree of deviation from a predetermined priority schedule. They developed a multi-objective evolutionary algorithm for that problem.

There are quite few papers dealing with the tactical berth scheduling. Moorthy and Teo (2006) was the first one to present the BTP, by which this study is greatly inspired. Their BTP defines berth-windows of serving calling ships in a continuous space within the predetermined length of the planning horizon. The berth template design takes into account the scheduling of periodicity, that is, the wrap-around effect of the cylinder. Their problem had two objectives: one is to maximize the service level, which is simply defined as the percentage of vessels served within two hours of their arrival, and the other is to minimize the connectivity cost, which is related to the distances between berths within vessel transshipment groups. Another tactical berth scheduling problem is studied by Giallombardo et al. (2010). They proposed the BTPT in discrete location indexes with the integration of quay crane (QC) allocation decision. In addition to the cost associated with QCs, they introduced the other cost component in the objective function, the yard cost that depends on the berthing location. Their study arranges all berth-windows within the time duration, similar to the concept of the cylinder length. Zhen et al. (2011) proposed an integrated template planning model for both berthing location in continuous indexes and yard container stack arrangement. In addition, the cyclic scheduling consideration and the QC allocation were considered. They developed a heuristic with a recursive process based on two stages: berth template and yard template. Hendriks et al. (2012) addressed a BTPS under a unique berthing service circumstance where ships can berth at any terminal in a port with inter-terminal service agreements, which allow containers to be unloaded from a ship at a remote partner terminal and transferred by trucks to the terminal the ship was originally scheduled to berth. Their BTP implicitly imposed the cylinder on the model since it assumed to serve cyclically calling ships. It takes into account the QC assignment to ships, resulting in the inclusion of the associated QC utilization cost in the objective function, which also considers the inter-terminal container transfer cost. Whereas their model is referred to as strategic, it may be categorized into a tactical model according to our hierarchical scheme of berth scheduling since all the ship calls are assumed to be served. Thus, it still can be thought as a tactical model if categorized by our hierarchical scheme of berth scheduling. Hendriks et al. (2013) addressed a BTPT that deals with berth allocation and yard planning within the cylinder. Lee and Jin (2013) studied a BTPT for feeder vessels to determine berth allocation for feeders in discrete locations and yard storage assignment for their transshipment cargoes. Whereas it considered cyclically calling feeders, it did not impose the cylinder on the model. Following the framework of Giallombardo et al. (2010), Vacca et al. (2013) developed an exact-solution algorithm based on the technique of branch and price for the integrated problem of berth and QC planning. Their study does not apply the cylinder, within which all berth-windows are planned to be placed. Instead, every ship calling request has preferred start and end times of the handling service. This preferred time duration is wide enough to place an actual berth-window of the ship appropriately so as to minimize the objective function. Finally, note that all the above BTP studies implicitly assume that the berthing capacity is large enough to cover all the calling requests.

3. PROBLEM FORMULATIONS

3.1. Problem Overview

The BTPS of this study focuses only on berthing decision. As reviewed in the previous section, quite a few existing integrated BTP models not only schedule berth-windows but also make decisions for other facility and/or equipment usage such as yard container slot allocation and QC assignment. These modelling approaches implicitly assume the availability of precise calling ship profiles, such as the amount of cargo to be loaded and unloaded. Otherwise, the ship handling time, an input for the BTPs, cannot be estimated by linking the number of QCs used for ship handling. For the cases with precise ship information, we think that the BTPT models in these existing papers are very suitable.

The main feature of this study is the strategic selection of the calling ships (or shipping lines) to be served within a spatial (berth) and temporal (cylinder length) berthing capacity of the terminal and the determination of the associated berth-windows for handling the ships. Under the assumption that ship profiles are not necessarily accurate or precise for long-term planning, the BTPS, unlike the BTPT, can exclude other decision making factors such as QC and storage yard arrangement in the decision process. Nonetheless, in the following tactical phase, additional and more detailed scheduling issues may be taken into account. For example, if the berth template determined in the strategic level is found to be non-executable in the subsequent tactical phase due to QC availability, the terminal operator may invest in some more QCs as a long-term decision or re-arrange the berth-windows subject to the QC restriction based on the approach in those BTPT studies with the integration of the QC decision.

The BTPS is basically the same as the BAP but with a cylinder constraint that ensures the berth-windows of all ships are packed within the predetermined cylinder length. As mentioned before, most container ships call at a terminal on a weekly basis; the cylinder length is one week in general. Because of its relatively long-term planning aspect, the BTPS assumes the same handling time of a ship regardless of its assigned berth. Thus, for this reason the BTPS aims to minimize the total ship start time delay (TDT), which is the sum of the deviation between ship arrival time and actual ship service start time (or berthing time), while the BAP minimizes the total ship service time (TST), in which the service time of a ship consists of its handling time and delay (or waiting) time for berthing.

The BTPS in discrete berthing locations that is discussed in this study is based on the BAP models proposed in Imai et al. (1997) and in Imai et al. (2001). The BTPS assumes (i) all ships can be served at any berth, (ii) each berth serves up to one ship at any time, (iii) handling time for a specific ship is constant regardless of its berthing location, (iv) ships are served within the cylinder, (v) mother and feeder ships with transshipment relations are both served or neither of them is served in order to address the practical issue under the hub-and-spoke operation, and (vi) a ship can be excluded from the berth-window planning at a price of incurring the associated discard cost/penalty.

Assumption (iii) is made since the container storage yard decision is not part of the consideration for optimization in this study. It is assumed that the corresponding yard

location can be close to the assigned berth and does not have a serious impact on the handling time, which is consequently assumed to be a constant, regardless of its assigned berth.

Assumption (vi) is made in order for the BTPS to, in addition to berth-window placement, identify ships to be served and those not to be served when the calling request exceeds the berthing capacity at a terminal. By manipulating the penalty level, the priority of the ships of the strategic consideration over the shipping lines can be incorporated into the model. For example, an extremely high value can be imposed on the ships of the customer that the terminal operator cannot afford to lose. In summary, the BTPS can be defined as: selecting ships to be served or un-served and determine the berth-windows for the served ships in order to minimize the sum of the total penalty for un-served ship and the total cost associated with the deviation between the actual service start times and their preferred start times.

3.2. BTPS Formulation

As described before, the BTPS is based on the model structure of DBAP developed in Imai et al. (2001) to take technical advantages for the model formulation and solving tips. For selection of ships to be served or un-served, we prepare a virtual berth (berth zero) in addition to physical berths that actually serve ships. By allocating unserved ships to berth zero, the BTPS can facilitate the ship selection with computational advantages that were found through the DBAP.

For the treatment of mother and feeder ships we define a 0-1 parameter, $R_{jj'}$, which indicates a connection of a pair of ships (one is a mother and the other is a feeder). $R_{jj'} = 1$ if ships j and j' are connected and $R_{jj'} = 0$ otherwise. To avoid the redundancy due to the symmetry of j and j' (i.e., $R_{jj'} = R_{j'j}$), $R_{jj'}$ is defined for $j, j' (> j) \in V$. This scheme does not explicitly distinguish which of j and j' is the mother ship. This allows multiple feeders to belong to a mother ship and/or a feeder ship to belong to multiple mother ships.

[BTPS]

$$\text{Minimize} \quad \sum_{i \in B} \sum_{j \in V} \sum_{k \in U} \{(k-1)C_j - A_j\}x_{ijk} + \sum_{i \in B} \sum_{j \in V} \sum_{k \in U} ky_{ijk} + \sum_{j \in V} \sum_{k \in U} G_j x_{0,jk} \quad (1)$$

$$\text{subject to} \quad \sum_{i \in B \cup \{0\}} \sum_{k \in U} x_{ijk} = 1 \quad \forall j \in V, \quad (2)$$

$$\sum_{j \in V} x_{ijk} \leq 1 \quad \forall i \in B \cup \{0\}, k \in U, \quad (3)$$

$$\sum_{l \in V} \sum_{m \in P_k} (C_l x_{ilm} + y_{ilm}) + y_{ijk} - A_j x_{ijk} \geq 0 \quad \forall i \in B, j \in V, k \in U, \quad (4)$$

$$\sum_{j \in V} \sum_{k \in U} (C_j x_{ijk} + y_{ijk}) - Y_i \leq CT \quad \forall i \in B, \quad (5)$$

$$Y_i \leq \left(1 - \sum_{j \in V} x_{ijk} \right) CT + \sum_{l \in V} \sum_{m \in P_k} (C_l x_{ilm} + y_{ilm}) + \sum_{j \in V} y_{ijk} \quad \forall i \in B, k \in U, \quad (6)$$

$$R_{jj'} \sum_{i \in B} \sum_{k \in U} x_{ijk} = R_{jj'} \sum_{i \in B} \sum_{k \in U} x_{ij'k} \quad \forall j, j' (> j) \in V, \quad (7)$$

$$Y_i \leq CT \quad \forall i \in B, \quad (8)$$

$$x_{ijk} \in \{0, 1\} \quad \forall i \in B \cup \{0\}, j \in V, k \in U, \quad (9)$$

$$0 \leq y_{ijk} \leq A_j \quad \forall i \in B, j \in V, k \in U, \quad (10)$$

where

$i (= 1, \dots, I) \in B$: set of berths

$j (= 1, \dots, T) \in V$: set of ships

$k (= 1, \dots, T) \in U$: set of service orders that are numbered in descending order from the last

ship to be served

P_k : subset of U such that $P_k = \{p | p > k \in U\}$

A_j : preferred arrival time of ship j

C_j : handling time spent by ship j

CT : cylinder length (or the cycle time of the planning horizon)

G_j : penalty cost (or priority) of ship j when it is not served.

Y_i : start of the cylinder for berth i

x_{ijk} : =1 if ship j is served as the k th ship at berth i , and =0 otherwise

y_{ijk} : idle time of berth i between the departure of the $(k+1)$ th ship and the arrival of the k th ship when ship j is served as the k th ship

Note that $i = 0$ is a dummy berth for ships not to be served. The decision variables are x_{ijk} s, y_{ijk} s and Y_i s.

The objective (1) minimizes two evaluation criteria in different dimensions: time and cost. This study assumes G_j is a time equivalent for convenience, while it is worth to

discuss how the penalty cost is converted to time factor. Constraint set (2) ensures that every ship that is selected to be served must be moored at one of the berths in one of the service orders. Constraints (3) enforce that every berth serves up to one ship at any time. Constraints (4) assure that ships are served after their preferred arrival time. Constraints (5), (6) and (8) guarantee that berth-windows of served ships are located within the cylinder. Equalities (7) ensure that a couple of a mother ship and a feeder in a transshipment contract are both served or neither of them is served.

A formulation that is comprised of the ship service delay part of the objective $\sum_{i \in B} \sum_{j \in V} \sum_{k \in U} \{(k-1)C_j - A_j\}x_{ijk} + \sum_{i \in B} \sum_{j \in V} \sum_{k \in U} ky_{ijk}$ associated with constraints (2) to (4), was originally developed for the DBAP in Imai et al. (2001) which the reader is asked to see for its derivation, while its formulation is overviewed in Appendix 1. Note that the DBAP model in Imai et al. assumes: (i) the objective is to minimize the TST (the sum of delay time for service and ship handling time), (ii) a ship's handling time depends on the allocated berth, (iii) the start time of berth availability, denoted by S_i , can be set as any arbitrary constant while this paper suggests that either it does not need to be set or it can be set as zero due to the cyclic scheduling nature, and (iv) the ship service order is numbered in ascending order from the first ship to be served while this paper utilizes the reverse order scheme for a simpler objective function structure.

It seems that $x_{0,jk}$ for the ship selection part of the objective is redundant since the service order k is irrelevant for un-served ships. As described in section 1, the solution procedure for [BTPS] is based on the subgradient method with Lagrangian relaxation. The lower bound can be easily obtained by the Lagrangian relaxation problem, which is equivalent to the classical assignment problem (AP), if we use $x_{0,jk}$ for the dummy berth just as using x_{ijk} for real berths.

Note that the berth idle time y_{ijk} can take any value. However, it is bounded to the arrival time of ship j as constraints (10), because it is at most A_j due to $S_i = 0$ being assumed in [BTPS].

It is also noteworthy that a mother ship and its associate feeders do not have to berth at the same time for transshipment. Ship loading and unloading is a very complicated job. So, for instance transshipment cargoes unloaded from a feeder are marshaled together with other cargoes (originating from that port) to be loaded onto a mother ship and then stacked at the yard for a while before being loaded on the mother ship arriving at a later time. Of course, another direction of transshipment is also possible. Therefore, both mother and feeder ships do not necessarily need to berth at the same time. Even if they are scheduled

to berth simultaneously, a direct transshipment between them is usually not performed.

The same cylinder can be applied to all the berths for a terminal where the start of the cylinder may be set as zero. However, for more flexible and efficient scheduling we allow each berth to have a cylinder starting from a different time slot (i.e., a berth-dependent Y_i).

4. SOLUTION PROCEDURES

The author developed an approximation algorithm for operational berth scheduling: DBAP, by using the subgradient procedure with the Lagrangian relaxation problem. The subgradient procedure with Lagrangian relaxation is widely used for optimization problems. The BAP is an example for its application, as seen in Imai et al. (2001), Park and Kim (2002) and Monaco and Sammarra (2007). This study also employs the subgradient procedure for the BTPS.

4.1. Lagrangian Relaxation

We introduce a Lagrangian relaxation problem to [BTPS]. Letting α_{ijk} , β_i , γ_{ik} and $\delta_{jj'}$ be Lagrangian multipliers for four constraint sets, (4)-(7), the Lagrangian relaxation is formulated as follows:

[RBTPS]

$$\begin{aligned}
 \text{Minimize} \quad & \sum_{i \in B} \sum_{j \in V} \sum_{k \in U} \{(k-1)C_j - A_j\}x_{ijk} + \sum_{i \in B} \sum_{j \in V} \sum_{k \in U} ky_{ijk} \\
 & - \sum_{i \in B} \sum_{j \in V} \sum_{k \in U} \alpha_{ijk} \left\{ \sum_{l \in V} \sum_{m \in P_k} (C_l x_{ilm} + y_{ilm}) + y_{ijk} - A_j x_{ijk} \right\} \\
 & - \sum_{i \in B} \beta_i \left\{ CT + Y_i - \sum_{j \in V} \sum_{k \in U} (C_j x_{ijk} + y_{ijk}) \right\} \\
 & - \sum_{i \in B} \sum_{k \in U} \gamma_{ik} \left\{ \left(1 - \sum_{j \in V} x_{ijk} \right) CT + \sum_{l \in V} \sum_{m \in P_k} (C_l x_{ilm} + y_{ilm}) + \sum_{j \in V} y_{ijk} - Y_i \right\} \\
 & - \sum_{j \in V} \sum_{j'(>j) \in V} \delta_{jj'} R_{jj'} \left\{ \sum_{i \in B} \sum_{k \in U} x_{ij'k} - \sum_{i \in B} \sum_{k \in U} x_{ijk} \right\}
 \end{aligned} \tag{11}$$

subject to (2), (3), (8)-(10)

Objective (11) can be re-written as follows:

$$\sum_{i \in B} \sum_{j \in V} \sum_{k \in U} \{(k-1)C_j - A_j\}x_{ijk} - \sum_{i \in B} \sum_{j \in V} \sum_{k \in U} \sum_{l \in V} \sum_{m \in P_k} \alpha_{ijk} C_l x_{ilm} + \sum_{i \in B} \sum_{j \in V} \sum_{k \in U} \alpha_{ijk} A_j x_{ijk}$$

$$\begin{aligned}
 & + \sum_{i \in B} \sum_{j \in V} \sum_{k \in U} \beta_i C_j x_{ijk} + CT \sum_{i \in B} \sum_{k \in U} \sum_{j \in V} \gamma_{ik} x_{ijk} - \sum_{i \in B} \sum_{j \in V} \sum_{k \in U} \sum_{m \in P_k} \gamma_{ik} C_l x_{ilm} \\
 & + \sum_{j \in V} \sum_{j'(>j) \in V} \sum_{i \in B} \sum_{k \in U} \delta_{jj'} R_{jj'} x_{ijk} - \sum_{j \in V} \sum_{j'(>j) \in V} \sum_{i \in B} \sum_{k \in U} \delta_{jj'} R_{jj'} x_{ij'k} \\
 & + \sum_{i \in B} \sum_{j \in V} \sum_{k \in U} k y_{ijk} - \sum_{i \in B} \sum_{j \in V} \sum_{k \in U} \sum_{l \in V} \sum_{m \in P_k} \alpha_{ijk} y_{ilm} - \sum_{i \in B} \sum_{j \in V} \sum_{k \in U} \alpha_{ijk} y_{ijk} + \sum_{i \in B} \sum_{j \in V} \sum_{k \in U} \beta_i y_{ijk} \\
 & - \sum_{i \in B} \sum_{j \in V} \sum_{k \in U} \sum_{m \in P_k} \gamma_{ik} y_{ilm} - \sum_{i \in B} \sum_{k \in U} \sum_{j \in V} \gamma_{ik} y_{ijk} - \sum_{i \in B} \beta_i Y_i + \sum_{i \in B} \sum_{k \in U} \gamma_{ik} Y_i - \sum_{i \in B} \beta_i CT - \sum_{i \in B} \sum_{k \in U} \gamma_{ik} CT
 \end{aligned}$$

In the above objective, $\sum_{i \in B} \sum_{j \in V} \sum_{k \in U} \sum_{l \in V} \sum_{m \in P_k} \alpha_{ijk} C_l x_{ilm}$, $\sum_{i \in B} \sum_{j \in V} \sum_{k \in U} \sum_{l \in V} \sum_{m \in P_k} \alpha_{ijk} y_{ilm}$,

$$\sum_{i \in B} \sum_{j \in V} \sum_{k \in U} \sum_{m \in P_k} \gamma_{ik} C_l x_{ilm} \quad \text{and} \quad \sum_{i \in B} \sum_{j \in V} \sum_{k \in U} \sum_{m \in P_k} \gamma_{ik} y_{ilm} \quad \text{can be} \quad \sum_{i \in B} \sum_{j \in V} \sum_{k \in U \setminus \{T\}} \sum_{l \in V} \sum_{m \in P_k} \alpha_{ijk} C_l x_{ilm},$$

$$\sum_{i \in B} \sum_{j \in V} \sum_{k \in U \setminus \{T\}} \sum_{l \in V} \sum_{m \in P_k} \alpha_{ijk} y_{ilm}, \quad \sum_{i \in B} \sum_{j \in V} \sum_{k \in U \setminus \{T\}} \sum_{m \in P_k} \gamma_{ik} C_l x_{ilm} \quad \text{and} \quad \sum_{i \in B} \sum_{j \in V} \sum_{k \in U \setminus \{T\}} \sum_{m \in P_k} \gamma_{ik} y_{ilm},$$

respectively, because $m > k$.

Due to the property shown in Appendix 2, they are further transformed to

$$\sum_{i \in B} \sum_{j \in V} \sum_{k \in U \setminus \{1\}} \sum_{l \in V} \sum_{m < k} \alpha_{ilm} C_j x_{ijk}, \quad \sum_{i \in B} \sum_{j \in V} \sum_{k \in U \setminus \{1\}} \sum_{l \in V} \sum_{m < k} \alpha_{ilm} y_{ijk}, \quad \sum_{i \in B} \sum_{j \in V} \sum_{k \in U \setminus \{1\}} \sum_{m < k} \gamma_{im} C_j x_{ijk} \quad \text{and}$$

$$\sum_{i \in B} \sum_{j \in V} \sum_{k \in U \setminus \{1\}} \sum_{m < k} \gamma_{im} y_{ijk}. \quad \text{As a result, the objective function becomes}$$

$$\begin{aligned}
 & \sum_{i \in B} \sum_{j \in V} \sum_{k \in U} \{(k-1)C_j - A_j\} x_{ijk} - \sum_{i \in B} \sum_{j \in V} \sum_{k \in U \setminus \{1\}} \sum_{l \in V} \sum_{m < k} \alpha_{ilm} C_j x_{ijk} + \sum_{i \in B} \sum_{j \in V} \sum_{k \in U} \alpha_{ijk} A_j x_{ijk} \\
 & + \sum_{i \in B} \sum_{j \in V} \sum_{k \in U} \beta_i C_j x_{ijk} + CT \sum_{i \in B} \sum_{k \in U} \sum_{j \in V} \gamma_{ik} x_{ijk} - \sum_{i \in B} \sum_{j \in V} \sum_{k \in U \setminus \{1\}} \sum_{m < k} \gamma_{im} C_j x_{ijk} \\
 & + \sum_{j \in V} \sum_{j'(>j) \in V} \sum_{i \in B} \sum_{k \in U} \delta_{jj'} R_{jj'} x_{ijk} - \sum_{j \in V} \sum_{j'(>j) \in V} \sum_{i \in B} \sum_{k \in U} \delta_{jj'} R_{jj'} x_{ij'k} \\
 & + \sum_{i \in B} \sum_{j \in V} \sum_{k \in U} k y_{ijk} - \sum_{i \in B} \sum_{j \in V} \sum_{k \in U \setminus \{1\}} \sum_{l \in V} \sum_{m < k} \alpha_{ilm} y_{ijk} - \sum_{i \in B} \sum_{j \in V} \sum_{k \in U} \alpha_{ijk} y_{ijk} + \sum_{i \in B} \sum_{j \in V} \sum_{k \in U} \beta_i y_{ijk} \\
 & - \sum_{i \in B} \sum_{j \in V} \sum_{k \in U \setminus \{1\}} \sum_{m < k} \gamma_{im} y_{ijk} - \sum_{i \in B} \sum_{k \in U} \sum_{j \in V} \gamma_{ik} y_{ijk} - \sum_{i \in B} \beta_i Y_i + \sum_{i \in B} \sum_{k \in U} \gamma_{ik} Y_i \\
 & - \sum_{i \in B} \beta_i CT - \sum_{i \in B} \sum_{k \in U} \gamma_{ik} CT
 \end{aligned}$$

Further,

$$\begin{aligned}
 & = \sum_{i \in B} \sum_{j \in V} \sum_{k \in U} \{(k-1)C_j - A_j\} x_{ijk} - \sum_{i \in B} \sum_{j \in V} \sum_{k \in U \setminus \{1\}} \sum_{l \in V} \sum_{m < k} \alpha_{ilm} C_j x_{ijk} + \sum_{i \in B} \sum_{j \in V} \sum_{k \in U} \alpha_{ijk} A_j x_{ijk} \\
 & + \sum_{i \in B} \sum_{j \in V} \sum_{k \in U} \beta_i C_j x_{ijk} + CT \sum_{i \in B} \sum_{k \in U} \sum_{j \in V} \gamma_{ik} x_{ijk} - \sum_{i \in B} \sum_{j \in V} \sum_{k \in U \setminus \{1\}} \sum_{m < k} \gamma_{im} C_j x_{ijk} \\
 & + \sum_{i \in B} \sum_{k \in U} \sum_{j \in V} \left(\sum_{l(>j) \in V} \delta_{jl} R_{jl} - \sum_{l(<j) \in V} \delta_{lj} R_{lj} \right) x_{ijk}
 \end{aligned}$$

$$\begin{aligned}
 & + \sum_{i \in B} \sum_{j \in V} \sum_{k \in U} k y_{ijk} - \sum_{i \in B} \sum_{j \in V} \sum_{k \in U \setminus \{1\}} \sum_{l \in V} \sum_{m < k} \alpha_{ilm} y_{ijk} - \sum_{i \in B} \sum_{j \in V} \sum_{k \in U} \alpha_{ijk} y_{ijk} + \sum_{i \in B} \sum_{j \in V} \sum_{k \in U} \beta_i y_{ijk} \\
 & - \sum_{i \in B} \sum_{j \in V} \sum_{k \in U \setminus \{1\}} \sum_{m < k} \gamma_{im} y_{ijk} - \sum_{i \in B} \sum_{k \in U} \sum_{j \in V} \gamma_{ik} y_{ijk} - \sum_{i \in B} \beta_i Y_i + \sum_{i \in B} \sum_{k \in U} \gamma_{ik} Y_i \\
 & - \sum_{i \in B} \beta_i CT - \sum_{i \in B} \sum_{k \in U} \gamma_{ik} CT .
 \end{aligned}$$

Given a set of Lagrangian multipliers α_{ijk} , β_i , γ_{ik} and $\delta_{jj'}$, [RBTPS] is completely separable. That is, it can be restructured (without loss of solution accuracy) into the following three subproblems:

[SUB-1]

$$\begin{aligned}
 \text{Minimize} \quad & \sum_{i \in B} \sum_{j \in V} \sum_{k \in U} \{(k-1)C_j - A_j\} x_{ijk} - \sum_{i \in B} \sum_{j \in V} \sum_{k \in U \setminus \{1\}} \sum_{l \in V} \sum_{m < k} \alpha_{ilm} C_j x_{ijk} + \sum_{i \in B} \sum_{j \in V} \sum_{k \in U} \alpha_{ijk} A_j x_{ijk} \\
 & + \sum_{i \in B} \sum_{j \in V} \sum_{k \in U} \beta_i C_j x_{ijk} + CT \sum_{i \in B} \sum_{j \in V} \sum_{k \in U} \gamma_{ik} x_{ijk} - \sum_{i \in B} \sum_{j \in V} \sum_{k \in U \setminus \{1\}} \sum_{m < k} \gamma_{im} C_j x_{ijk} \\
 & + \sum_{j \in V} \sum_{j'(>j) \in V} \sum_{i \in B} \sum_{k \in U} \delta_{jj'} R_{jj'} x_{ijk} - \sum_{j \in V} \sum_{j'(>j) \in V} \sum_{i \in B} \sum_{k \in U} \delta_{jj'} R_{jj'} x_{ij'k} \tag{12}
 \end{aligned}$$

subject to (2), (3), (9)

[SUB-2]

$$\begin{aligned}
 \text{Minimize} \quad & \sum_{i \in B} \sum_{j \in V} (1 + \beta_i - \alpha_{ij,1} - \gamma_{i,1}) y_{ij,1} + \\
 & \sum_{i \in B} \sum_{j \in V} \sum_{k \in U \setminus \{1\}} \left(k + \beta_i - \alpha_{ijk} - \gamma_{ik} - \sum_{l \in V} \sum_{m < k} \alpha_{ilm} - \sum_{m < k} \gamma_{im} \right) y_{ijk} \tag{13}
 \end{aligned}$$

subject to (10)

[SUB-3]

$$\text{Minimize} \quad \sum_{i \in B} \left(\beta_i - \sum_{k \in U} \gamma_{ik} \right) Y_i \tag{14}$$

subject to (8)

[SUB-1] is optimally solved with the AP. In [SUB-2], $y_{ij,1} = 0$ if $1 + \beta_i - \alpha_{ij,1} - \gamma_{i,1} \geq 0$, and otherwise $y_{ij,1} = A_j$. Further, $y_{ijk} (\forall k \in U \setminus \{1\}) = 0$ if $k + \beta_i - \alpha_{ijk} - \gamma_{ik} - \sum_{l \in V} \sum_{m < k} \alpha_{ilm} - \sum_{m < k} \gamma_{im} \geq 0$, and otherwise $y_{ijk} = A_j$. Note that

$y_{ijk}(\forall k \in U) = 0$ if corresponding $x_{ijk} = 0$. Similarly, for [SUB-3], $Y_i = 0$ if

$$\beta_i - \sum_{k \in U} \gamma_{ik} \geq 0, \text{ while } Y_i = CT \text{ if } \beta_i - \sum_{k \in U} \gamma_{ik} < 0.$$

Subgradients to be used are $\phi_{\alpha,ijk} = \sum_{l \in V} \sum_{m \in P_k} (C_l x_{ilm} + y_{ilm}) + y_{ijk} - A_j x_{ijk}$,

$$\phi_{\beta,i} = CT + Y_i - \sum_{j \in V} \sum_{k \in U} (C_j x_{ijk} + y_{ijk}), \quad \phi_{\gamma,ik} = \left(1 - \sum_{j \in V} x_{ijk}\right) CT + \sum_{l \in V} \sum_{m \in P_k} (C_l x_{ilm} + y_{ilm}) + \sum_{j \in V} y_{ijk} - Y_i$$

and $\phi_{\delta,jj'} = R_{jj'} \left\{ \sum_{i \in B} \sum_{k \in U} x_{ijk} - \sum_{i \in B} \sum_{k \in U} x_{ijk} \right\}$. Lagrangian multipliers α_{ijk} , β_i , γ_{ik} and $\delta_{jj'}$,

are updated by $\alpha^{n+1} = \alpha^n - t_n \phi_{\alpha,ijk}^{(n)}$, $\beta^{n+1} = \beta^n - t_n \phi_{\beta,i}^{(n)}$, $\gamma^{n+1} = \gamma^n - t_n \phi_{\gamma,ik}^{(n)}$ and

$\delta^{n+1} = \delta^n - t_n \phi_{\delta,jj'}^{(n)}$ where the step size t_n is defined by

$$t_n = d_n (\bar{Z} - Z(\alpha^{(n)}, \beta^{(n)}, \gamma^{(n)}, \delta^{(n)})) / \|\phi^{(n)}\|.$$

4.2. Subgradient Procedure

The outline of the subgradient optimization with the above relaxed problem is shown in Appendix 3. This procedure is basically the same as the one used for the DBAP in Imai et al. (2001), which proposed three Lagrangian heuristics, SIMPLE, INDIVIDUAL and INTERACT, in order to derive a feasible solution to the DBAP by modifying the solution of its relaxed problem. The relaxed problem defined in their study is equivalent to [SUB-1] in this study. SIMPLE merely determines the start time of a ship service based on ship service orders for each berth, which are obtained by the relaxed problem (called SBAP), but with a constraint that ship services start after arrival. See Appendix 1 for the SBAP formulation. INDIVIDUAL is basically the same as SIMPLE, but it changes service orders of ships if there is a berth idle time between a specific pair of ships in a feasible solution by SIMPLE. INTERACT swaps ships across berths so as to fulfill the berth idle time for a reduction of the total service time. See Imai et al. (2001) for the details of those heuristics.

4.3 Lagrangian Heuristic for Finding a Feasible Solution

In Step 3 of the subgradient procedure shown in Appendix 3, a feasible solution is found from an optimal solution to the Lagrangian relaxation problem of [BTPS] by a Lagrangian heuristic. This section describes the outline of the Lagrangian heuristic.

In the subgradient procedure for [DBAP] in Imai et al. (2001), a feasible solution is

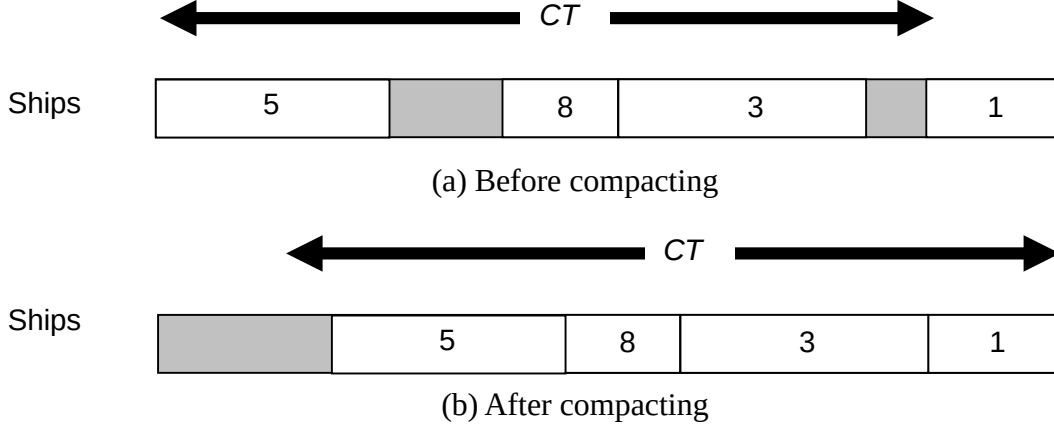


Fig. 1 Compacting ship services in the cylinder

found by one of three different processes: SIMPLE, INDIVIDUAL and INTERACT. The Lagrangian heuristic first performs one of those processes for [DBAP]. If the resulting solution satisfies the relaxed constraints (5), i.e., the cylinder constraint, the Lagrangian heuristic terminates. If it does not, the Lagrangian heuristic continues the process that shifts some ships from their berths without cylinder satisfaction to other berths, so that all the berths satisfy the cylinder constraint. If some berths are still not satisfactory despite of this shifting, some ships are dropped from the service, so that berth-windows for all ships to be served are placed within the cylinder.

The Lagrangian heuristic employs the similar procedure to the one for the Bin-Packing Problem, which packs items of various sizes into a set of bins while minimizing the number of bins used. Many heuristics have been exploited for this problem since the early 1970s. For further details, see Bramel and Simchi-Levi (1997). We employ one of the popular Bin-Packing heuristics, First-Fit Decreasing, which first sorts the items in non-increasing order of their size and places them in the lowest indexed bin at the moment of packing whose current content does not exceed the bin capacity. Items and bins correspond to ships and berths in our problem background.

A more formal description of the heuristic follows:

[LH]

- Step 1. In the solution to [SUB-1], if a ship selected not to be served is connected with other ships with $R_{jj'} = 1$, those ships are discarded too. Perform a DBAP's Lagrangian heuristic (SIMPLE, INDIVIDUAL or INTERACT) with those ships to be served. If all berths satisfy the cylinder constraint, then STOP.

- Step 2. Setting the end time of the cylinder as the $k(=1)$ th ship's completion time of service, set services of all the preceding ships in time without berth idle between any adjacent two ships, as shown in Fig. 1. If all berths serve all ships assigned to them within the cylinder, go to Step 8.
- Step 3. Mark those berths without cylinder satisfaction and refer to them as violating berths.
- Step 4. For all violating berths, remove ships in ascending order of its handling time C_j until the total ship handling time is no more than CT . Register them on a ship list.
- Step 5. Arrange ships in non-increasing order of their handling times on the ship list.
- Step 6. Examine a ship from the ship list. Find a berth, from berth 1, with idle berth time within CT that lets the ship stay. If such a berth is found, place the ship at the time closest to its arrival time in the berth and delete it from the ship list. Otherwise, the ship is registered as an un-served ship. If the un-served ship is associated with other ships with $R_{jj'} = 1$ either in the ship list or being as placed in the berths, those connected ships are also registered as un-served ones and deleted from the list and the berths. Update the idle berth time for berths if a ship is inserted in a berths and/or ships are deleted from berths in the above process.
- Step 7. Repeat Step 6 until all the ships in the ship list are examined.
- Step 8. Arrange ships' services in respective berths so that they start their services as soon as arrival time within CT .

For an arrangement of ship services for each berth in Step 8 of [LH], the following procedure is used:

- Step 8-1. The end time of the cylinder, CT_{end} , is set as the $k(=1)$ th ship's completion time of service. The start time of the cylinder, CT_{start} , is set as $CT_{start} = CT_{end} - CT$.
- Step 8-2. From the last ship to the first ship in service order k for the berth, compute the start time (ST) and completion time (FN) of ship's service as follows:
 If ship j is the last ship, $ST = \text{Max}(CT_{start}, A_j)$, $FN = ST + C_j$; otherwise,
 $ST = \text{Max}(FN \text{ of the previous ship}, A_j)$, $FN = ST + C_j$.

The heuristic initially deals with all the served ships that are selected by solving [SUB-1]. However, some of those ships may be discarded in the heuristic since the solution to [SUB-1] does not satisfy the cylinder constraints (5).

5. NUMERICAL EXPERIMENTS

5.1. Outline of the Experiments

The solving algorithm for the BTPS is constructed with theoretical implications of the DBAP algorithm in Imai et al. (2001). From this viewpoint, we first perform preliminary experiments to examine how the subgradient procedure works by using the DBAP instances. Subsequently, we perform a wide variety of experiments for the BTPS. All the solution procedures for the DBAP and BTPS are coded in “C” language on a Panasonic Let’s note CF-B11 computer.

5.2. DBAP Experiments

Before the experiments for the BTPS, we compare the three Lagrangian heuristics of SIMPLE, INDIVIDUAL and INTERACT for the DBAP. All computational instances assume four berths with 100 calling ships. We prepared three patterns of interval of the ship arrival time, A_j , being generated by exponential random variable with an average

of 1, 5 and 8h. The cargo handling time of a ship, C_{ij} , varies on its potential berthing

locations, but it was generated based on uniform random variables with different average times of 4 and 12h. Also, we prepared three different fluctuations between the maximum and minimum amounts of the cargo handling time of a ship: 0, 100 and 200% of the average time. In total, 18 different calling ship scenarios were prepared with those data factors. For each computation instance of the 18 scenarios, we set three different start times of the planning horizon, S_i , where all berths have the same value of S_i . The

earliest time is S_i equivalent to the first quarter of the 100 ships (S1), the next is at half (S2) and the third is the three quarters (S3). The combination of three different times and 18 scenarios leads to 54 problem instances in total. For every single instance, the procedure was terminated with 200 iterations.

Table 1 shows the average values (over the 18 instances) of UB (upper bound), LB (lower bound), GAP $(=(UB-LB)/LB*100)$ and CPU time, and the total values (of the 18 instances) of count of the best solutions among other methods for different S_i s. The bottom line in Table 1 shows the grand average UB and count of optimal solutions over cases S1 to S3. A value for each line in bold is the best among the nine solution methods. Note that the GAP is not computed with average UB and LB, but it is the average over GAPs for all individual problem instances.

First of all, the solution quality in terms of GAP is improving with larger S_i . Typically, in the case of S1, LB is negative (LBs of some individual problem instances are positive); this implies that the DBAP with an early start time of planning horizon is hard to solve.

The GAP is not shown for S1 instances because a correct indication of solution quality is not provided by the negative LB. For S2, one problem instance out of the 18 has a negative LB, so its GAP is an average of other 17 instances. This trend of lower solution quality

Table 1 DBAP results by different S_i

	SIMPLE	INDIVIDUAL	INTERACT	
S1	UB(h)	16915.8	4007.9	3218.9
	LB(h)	-5853.0	-5845.3	-5844.9
	GAP(%)	-	-	-
	CPU(s)	30.7	30.1	31.5
S2	UB(h)	18661.8	8908.8	8606.9
	LB(h)	5004.5	5005.8	5006.2
	GAP (%)*	1289.1	405.4	396.8
	CPU(s)	28.5	28.2	25.9
S3	UB(h)	21817.6	16752.3	16726.6
	LB(h)	15664.0	15666.0	15665.8
	GAP(%)	39.1	7.5	7.4
	CPU(s)	25.1	22.0	22.0
Ave	UB(h)	19131.7	9889.7	9517.5

UB(h): Upper bound

LB(h): Lower bound

GAP(%)=(UB-LB)/LB*100

* one out of 18 instances for S2 has a negative LB. So, GAP is average of 17 instances.

with smaller S_i is also reported in Imai et al. (2001). The reason why the GAP improves

with larger S_i is that the DBAP with larger S_i reduces to the SBAP, which easily finds an optimal solution.

Generally speaking, SIMPLE is the worst among the three Lagrangian heuristics. Both INDIVIDUAL and INTERACT result in almost the same performance, but INTERACT is superior to the others.

5.3. BTPS Experiments

Next, we perform numerical experiments for BTPS. Providing four berths, we have four scenarios that serve the number of calling ships (T) ranging from 50 to 65. Also, to examine the BTPS procedures with large problem instances, we prepare instances of 100 ships with eight berths. Thus, we have five scenarios in total. Each of all the problem instances has ten mother-feeder connections, where two of the ships in each connection are served or neither of them is.

The tactical BTP (BTPT) is a hard problem to be solved because of the cylinder length. It is likely that we do not find a feasible solution of the BTPT with an enormous number of calling ships. The BTPS overcomes this drawback by eliminating less important ships from service. However, the BTPS should be examined in two cases: one when all ships are served and the other when all of them are not due to the cylinder. Therefore, we carefully design problem instances. To do so, for each scenario the given ships are spread during the cylinder length ($CT = 150h$, which is equivalent to almost one week) in terms of arrival time whose interval follows an exponential distribution. For the arrival time we created three sets with different seeds for random numbers. The ship handling time, C_j ,

was generated based on uniform random variables with different average times of 4, 8 and 12h. Also, we prepared three different fluctuations between the maximum and minimum amounts of the cargo handling time of a ship: 0, 100 and 200% of the average time. In total, we have 27 problem instances for a scenario with a specific number of calling ships.

For BTPS experiments, penalty costs are carefully designed; otherwise, experimental results may be meaningless. The BTPS minimizes the total of delay time and penalty (in terms of time). If the value of the ship penalty is not so large compared to the ship handling time, a great number of ships may not be selected to be served. More concretely speaking, if much delay is expected by serving lots of ships while the berthing capacity is not large enough to cover all of them, the BTPS model refuses many of them to reduce the delay time. To avoid such an unrealistic solution, the penalty is set to a much larger value than the ship handling time. In our experiments, the penalty value is 10000 times as much as the handling time.

Like DBAP experiments, we have three solution procedures: SIMPLE, INDIVIDUAL and INTERACT. All the procedures were run with 200 iterations.

Table 2 illustrates computation results for those procedures. The results are summarized as the average values (over the 27 problem instances) of OBJ (the objective function value, i.e., the total of penalty cost and TDT (the total delay time)), TDT (h) and CPU time (s), and the total values (of the 27 instances) of the count of best solutions among the nine procedures and the count of optimal solutions. Like the DBAP analysis, for each line, figures in bold are the best.

Like the DBAP, the BTPS procedures produced negative LB since the BTPS procedures (the DBAP with the cylinder constraint) assume $S_i = 0$; for this reason, LB and GAP are not shown in Table 2.

The five scenarios with different calling ships correspond to the range of different arrival intervals from 1.4 to 2.8h, as indicated in the table. At the bottom are the average values of those evaluation terms over the five scenarios.

First, we examined the average values over the five scenarios. Comparing SIMPLE, INDIVIDUAL and INTERACT, SIMPLE is worst like in the DBAP computation. The second best is INTERACT and the best is INDIVIDUAL. This tendency is not the same

as the one for the DBAP.

Table 2 BTPS results

T	interval(h)		SIMPLE	INDIVIDUAL	INTERACT
50	2.8	OBJ	29460.6	28480.6	28473.0
		TDT(h)	2793.9	1813.9	1806.3
		CPU(s)	5.4	5.3	5.4
55	2.7	OBJ	111134.6	109997.8	109970.2
		TDT(h)	3356.9	2220.0	2192.4
		CPU time(s)	7.7	7.6	7.7
60	2.5	OBJ	319374.4	316879.8	316953.1
		TDT(h)	3448.5	2065.0	2138.3
		CPU(s)	10.5	10.2	10.5
65	2.3	OBJ	475158.8	473390.1	473574.6
		TDT(h)	4019.3	2279.0	2463.5
		CPU(s)	13.6	13.5	13.6
100	1.4	OBJ	59075.2	55869.6	55915.3
		TDT(h)	4462.7	2536.3	2581.9
		CPU(s)	153.9	150.6	151.7
Average		OBJ	198679.0	196923.6	196977.2
		TDT(h)	3616.3	2182.8	2236.5
		CPU(s)	38.2	37.4	37.8

OBJ(h): Total delay time + penalty cost

TDT(h): Total delay time

6. CONCLUSIONS

This paper addressed the relatively long-term decision making of berth schedule, i.e., the berth template problem in discrete berthing locations. Two of the problems, namely the strategic and tactical berth template problems, deal with the situations with different planning intervals and strategic importance. The strategic one (BTPS) chooses ships to be served and those not to be served for the case with excessive calling requests when compared to the berthing capacity. For such a case, the tactical one (BTPT) usually cannot find a feasible solution. Therefore, the former is in general more applicable to congested berthing situations since it always finds a solution. In particular, we incorporate in the model the practical consideration about the simultaneous treatment of mother and feeder ships under the hub-and-spoke operation. Although some BTPT studies have been done for the last few years, we have not found a BTP study addressing these strategic issues as the BTPS in this paper does.

Regarding the solution algorithm, we developed a subgradient procedure with

Lagrangian relaxation to find an approximate solution to the BTPS. In a previous study by the author, the solution algorithms of the berth allocation problem (DBAP) for operational scheduling were developed based on the subgradient procedure with the Lagrangian relaxation. Since the BTPS shares the key structure of problem formulation with the DBAP, the subgradient approach has also been applied to the BTPS.

The modeling framework and the solution algorithms that are developed in this paper should be useful for the terminal operators to better manage their valuable resources. In particular, since the BTPS model incorporates the decision of selecting the calling ships/shipping lines strategically, it is very suitable for the situations with excessive demand or the cases of capacity expansion. Thus, we believe this study can be useful in supporting the decision-making capabilities of terminal operators from a practical point of view.

ACKNOWLEDGEMENT

This research is supported by the JSPS Grant-in-Aid for Scientific Research-B Grant 25282093.

APPENDIX 1. SBAP and DBAP Formulations

[SBAP]

$$\text{Minimize} \quad \sum_{i \in B} \sum_{j \in V} \sum_{k \in U} (kC_{ij} + S_i - A_j)x_{ijk} \quad (\text{A.1})$$

$$\text{subject to} \quad \sum_{i \in B} \sum_{k \in U} x_{ijk} = 1 \quad \forall j \in V, \quad (\text{A.2})$$

$$\sum_{j \in V} x_{ijk} \leq 1 \quad \forall i \in B, k \in U, \quad (\text{A.3})$$

$$x_{ijk} \in \{0, 1\} \quad \forall i \in B, j \in V, k \in U, \quad (\text{A.4})$$

[DBAP]

$$\text{Minimize} \quad \sum_{i \in B} \sum_{j \in V} \sum_{k \in U} (kC_{ij} + S_i - A_j)x_{ijk} + \sum_{i \in B} \sum_{j \in W_i} \sum_{k \in U} ky_{ijk} \quad (\text{A.5})$$

subject to (A.2)-(A.4)

$$\sum_{l \in V} \sum_{m \in P_k} (C_{il}x_{ilm} + y_{ilm}) + y_{ijk} - (A_j - S_i)x_{ijk} \geq 0 \quad \forall i \in B, j \in W_i, k \in U, \quad (\text{A.6})$$

$$y_{ijk} \geq 0 \quad \forall i \in B, j \in V, k \in U, \quad (\text{A.7})$$

where S_i is the start time of the availability of berth i , W_i is the set of ships who arrive at port after S_i and C_{ij} is the handling time being spent by ship j at berth i . Other

parameters and variables are the same as the ones for [BTPS]. Constraints (A.2)-(A.6) are basically the same as (2)-(4), (9), (10). All the calling ships are already in port in the SBAP, while they are not all so in the DBAP.

Note here that the formulations above are slightly different from the one in Imai et al. (2001). The difference arises from the structure of parameters in the objective function due to the service order scheme. That is, Imai et al. (2001) has $\{(T - k' + 1)C_{ij} + S_i - A_j\}x_{ijk}$ in the objective function where the service order k' is numbered in ascending order from the first one to be served. However, the x_{ijk} variable-associated coefficient in Imai et al. and the one in this paper are completely equivalent, since $(T - k' + 1)$ for k' increasing from 1 to T turns to be k that decreases from T to 1. The objective (A.1) is used in this paper because of the formulation simplicity.

APPENDIX 2. Transformation of the Objective Function

$$\text{We derive } \sum_{i \in B} \sum_{j \in V_i} \sum_{k \in U \setminus \{T\}} \alpha_{ijk} \sum_{l \in V} \sum_{m \in P_k} x_{ilm} = \sum_{i \in B} \sum_{j \in V} \sum_{k \in U \setminus \{1\}} \sum_{l \in V} \sum_{m < k} \alpha_{ilm} x_{ijk}.$$

The formulation can simply be transformed as $\sum_{i \in B} \sum_{j \in V} \sum_{k \in U \setminus \{T\}} \alpha_{ijk} \sum_{l \in V} \sum_{m \in P_k} x_{ilm} = \sum_{i \in B} \sum_{j \in V_i} \sum_{k \in U \setminus \{T\}} \alpha_{ijk} \sum_{l \in V} \sum_{m > k} x_{ilm}$. We substitute k' for m and m' for k , then we have $\sum_{i \in B} \sum_{j \in V} \sum_{m' \in U \setminus \{T\}} \alpha_{ijm'} \sum_{l \in V} \sum_{k' > m'} x_{ilk'}$. Since element k' is more than m' , k' belongs to a subset $U \setminus \{1\}$. Reversely, m' is less than k' . Consequently, we have $\sum_{i \in B} \sum_{j \in V} \sum_{m' \in U \setminus \{T\}} \alpha_{ijm'} \sum_{l \in V} \sum_{k' > m'} x_{ilk'} = \sum_{i \in B} \sum_{j \in V} \sum_{k' \in U \setminus \{1\}} \sum_{l \in V} \sum_{m' < k'} \alpha_{ijm'} x_{ilk'}$. Once again substituting k and m for k' and m' , respectively, we have $\sum_{i \in B} \sum_{j \in V} \sum_{k \in U \setminus \{1\}} \sum_{l \in V} \sum_{m < k} \alpha_{ilm} x_{ijk}$.

APPENDIX 3. Subgradient Procedure for the BTPS

- Step 1. Maxiter = 200, $d = 2$, $\bar{Z} = 1 \times 10^8$, Iter = 1, $n = 1$, BestLB = 0, $(\alpha = \alpha^* = 0, \beta = \beta^* = 0, \gamma = \gamma^* = 0, \delta = \delta^* = 0)$.
- Step 2. Solve problem [RBTP], and calculate its objective function. Let Z_{RBTPD} be the solution value of [RBTP]. If $Z_{\text{RBTPD}} > \text{BestLB}$, let BestLB = Z_{RBTPD} , Iter = 1, $(\alpha^* = \alpha, \beta^* = \beta, \gamma^* = \gamma, \delta^* = \delta)$, otherwise Iter = Iter + 1.
- Step 3. Perform a heuristic by using an optimal solution to [SUB-3] to find a feasible solution to [BTPT]. If the feasible solution is not found, STOP; otherwise, let FEAS be the objective function value of the feasible solution. If $\text{FEAS} < \bar{Z}$, let $\bar{Z} = \text{FEAS}$, If $\bar{Z} - \text{BestLB} < 1$, STOP.
- Step 4. Let $n = n + 1$. If $n > \text{Maxiter}$, STOP; otherwise continue.

Step 5. If $\text{Iter} > 20$ let $\text{Iter} = 1$, $(\alpha^* = \alpha, \beta^* = \beta, \gamma^* = \gamma, \delta^* = \delta)$, $d_n = d_n/2$, otherwise

calculate step size t_n and update multipliers α_{ijk} , β_i , γ_{ik} and $\delta_{jj'}$.

Step 6. If α_{ijk} , β_i , γ_{ik} or $\delta_{jj'} < 0$, then set it=zero. Go to Step 2.

REFERENCES

- Bramel, J. and Simchi-Levi, D. (1997) *The Logic of Logistics*, Springer-Verlag, New York, NY.
- Buhrkal, K., Zuglian, S., Ropke, S., Larsen, J. and Lusby, R. (2011) Models for the discrete berth allocation problem: a computational comparison, *Transportation Research Part E* 47: 461-473.
- Cheong, C.Y., Tan, K.C., Liu, D.K. and Lin, C.J. (2010) Multi-objective and prioritized berth allocation in container ports, *Annals of Operations Research* 180: 63-103.
- Cordeau, J.-F., Laporte, G., Legato, P. and Moccia, L. (2005) Models and tabu search heuristics for the berth-allocation problem, *Transportation Science* 39: 526-538.
- De Oliveira, R.M., Mauri, G.R. and Lorena, L.A.N. (2012) Clustering search for the berth allocation problem, *Expert Systems with Applications* 39: 5499-5505
- Giallombardo, G., Moccia, L., Salani, M. and Vacca, I. (2010) Modeling and solving the tactical berth allocation problem, *Transportation Research Part B* 44: 232-245.
- Golias, M.M., Boile, M. and Theofanis, S. (2009) Berth scheduling by customer service differentiation: a multi-objective approach, *Transportation Research Part E* 45: 878-892.
- Golias, M.M., Boile, M. and Theofanis, S. (2010) A lamda-optimal based heuristic for the berth scheduling problem, *Transportation Research Part C* 18: 794-806.
- Hansen, P., Oguz, C. and Mladenovic, N. (2008) Variable neighborhood search for minimum cost berth allocation, *European Journal of Operational Research* 191: 636-649.
- Hendriks, M.P.M., Armbruster, D., Laumanns, M., Lefebvre, E. and Udding, J.T. (2012) Strategic allocation of cyclically calling vessels for multi-terminal container operators, *Flexible Services and Manufacturing Journal* 24: 248-273.
- Hendriks, M.P.M., M., Lefebvre, E. and Udding, J.T. (2013) Simultaneous berth allocation and yard planning at a tactical level, *OR Spectrum* 35: 441-456.
- Imai, A., Nagaiwa, K. and Chan, W.T. (1997) Efficient planning of berth allocation for container terminals in Asia, *Journal of Advanced Transportation* 31: 75-94.
- Imai, A., Nishimura, E. and Papadimitriou, S. (2001) The dynamic berth allocation problem for a container port, *Transportation Research Part B* 35: 401-417.
- Imai, A., Nishimura, E. and Papadimitriou, S. (2003) Berth allocation with service priority, *Transportation Research Part B* 37: 437-457.

- Imai, A., Nishimura, E. and Papadimitriou, S. (2005a) Corrigendum to “The dynamic berth allocation problem for a container port” [Transportation Research Part B 35 (2001) 401-417], *Transportation Research Part B* 39: 197.
- Imai, A., Sun, X., Nishimura, E. and Papadimitriou, S. (2005b) Berth allocation in a container port: using a continuous location space approach, *Transportation Research Part B* 39: 199-221.
- Imai, A., Nishimura, E., Hattori, M. and Papadimitriou, S. (2007) Berth allocation at indented berths for mega-containerships, *European Journal of Operational Research* 179: 579-593.
- Imai, A., Nishimura, E. and Papadimitriou, S. (2008) Berthing ships at a multi-user container terminal with a limited quay capacity, *Transportation Research Part E* 44: 136-151.
- Imai, A., Nishimura, E. and Papadimitriou, S. (2013) Marine container terminal configurations for efficient handling of mega-containership, *Transportation Research Part E* 49: 141-158.
- JICT (2014) *Jakarta International Container Terminal, Berthing Contract*, <http://jict.co.id/?x0=berthing%2Bcontract&x1=26&x2=article>.
- Lalla-Ruiz, E., Melian-Batista, B. and Moreno-Vega, J.M. (2012) Artificial intelligence hybrid heuristic based on tabu search for the dynamic berth allocation problem, *Engineering Applications of Artificial Intelligence* 25: 1132-1141
- Lee, D.-H., Chen, J.H. and Cao, J.X. (2010) The continuous berth allocation problem: a greedy randomized adaptive search solution, *Transportation Research Part E* 46: 1017-1029.
- Lee, D.-H. and Jin, J.G. (2013) Feeder vessel management at container transshipment terminals, *Transportation Research Part E* 49: 201-216.
- Li, C.-L., Cai, X. and Lee, C.-Y. (1998) Scheduling with multiple-job-on-one-processor pattern, *IIE Transactions* 30: 433-445.
- Lim, A. (1998) The berth planning problem, *Operations Research Letters* 22: 105-110.
- Mauri, G.R., Oliveira, A.C.M. and Lorena, L.A.N. (2008) A hybrid column generation approach for the berth allocation problem, *Lecture Notes in Computer Science* 4972: 110-122.
- Monaco, M.F. and Sammarra, M. (2007) The dynamic berth allocation problem: a strong formulation solved by Lagrangean approach, *Transportation Science* 41: 265-280.
- Mongelluzzo, B. (2013) Decline of the single-user terminal, *Journal of Commerce* 14: 36-38.
- Moorthy, R. and Teo, C.-P. (2006) Berth management in container terminal: the template design problem, *OR Spectrum* 28: 495-518.
- Nishimura, E., Imai, A. and Papadimitriou, S. (2001) Berth allocation planning in the public berth system by genetic algorithms, *European Journal of Operational Research* 131: 282-292.
- Park, K.T. and Kim, K.H. (2002) Berth scheduling for a container terminals by using sub-

- gradient optimization technique, *Journal of the Operational Research Society* 53: 1054-1062.
- Pinedo, M. (2012) *Scheduling: Theory, Algorithms, and Systems*, Springer US, Boston, MA.
- Saharidis, G.K.D., Golias, M.M., Boile, M., Theofanis, S. and Ierapetritou, M.G. (2010) The berth scheduling problem with customer differentiation: a new methodological approach based on hierarchical optimization, *International Journal of Advanced Manufacturing Technology* 46: 377-393.
- Ting, C.-J., Wu, K.-C. and Chou, H. (2014) Particle swarm optimization algorithm the berth allocation problem, *Expert Systems with Applications* 41: 1543-1550.
- Vacca, I., Salani, M. and Bierlaire, M. (2013) An exact algorithm for integrated planning of berth allocation and quay crane assignment, *Transportation Science* 47: 148-161.
- Xu, D., Li, C.-L. and Leung, J. Y.-T. (2012) Berth allocation with time-dependent physical limitations on vessels, *European Journal of Operational Research* 216: 47-56.
- Zhen, L., Chew, E.P. and Lee, L.H. (2011) An integrated model for berth template and yard template planning in transshipment hubs, *Transportation Science* 45: 483-504.