

Berthing ships at a multi-user container terminal with a limited quay capacity

Akio Imai ^{a,b,*}, Etsuko Nishimura ^a, Stratos Papadimitriou ^c

^a Faculty of Maritime Sciences, Kobe University, Fukae, Higashinada, Kobe 658-0022, Japan

^b World Maritime University, P.O. Box 500, S-201 24 Malmö, Sweden

^c Department of Maritime Studies, University of Piraeus, 80 Karaoli & Dimitriou Street, GR185 32 Piraeus, Greece

Received 27 October 2005; received in revised form 10 March 2006; accepted 26 May 2006

Abstract

This paper addresses a variation of the berth allocation problem at multi-user terminals, as ships which would normally be served at the terminal but their expected wait time exceeds the time limit, are assigned to an external terminal. The objective of the problem is to minimize the total service time of ships at the external terminal. A genetic algorithm based heuristic is developed and a wide variety of numerical experiments showed that the heuristic developed performed well in reducing external terminal usage and thus may be helpful in the efficient management of busy ports during extreme peaking conditions.

© 2006 Elsevier Ltd. All rights reserved.

Keywords: Berth allocation; Terminal management; Container transportation; External terminal; Heuristic; Mathematical programming

1. Introduction

While there are many container ports in the world with heavy traffic that serve as *Dedicated Terminals* to specific shipping lines, there is also another type of container terminals, operating as *Common User Terminals* also known as *Multi-User Terminals (MUT)* offering services to any shipping lines. These terminals are generally configured with a long quay comprising several berths, having the capability of serving more than one ship at a time and it is not very likely that a particular calling ship is served at a specific berth every time it calls. Most of such terminals employ the First-Come-First-Served (FCFS) principle for their daily operations. It is worth to note that essentially the concept of MUT is independent from the number of berths at the terminal of concern; however, from the viewpoint of berth scheduling complexity, we base this study on an MUT with a specific number of berths, e.g., 4–8 berths. The MUTs are mostly state-owned in developing countries and privately-owned in developed countries.

* Corresponding author. Tel.: +81 78 431 6261.

E-mail address: imai@maritime.kobe-u.ac.jp (A. Imai).

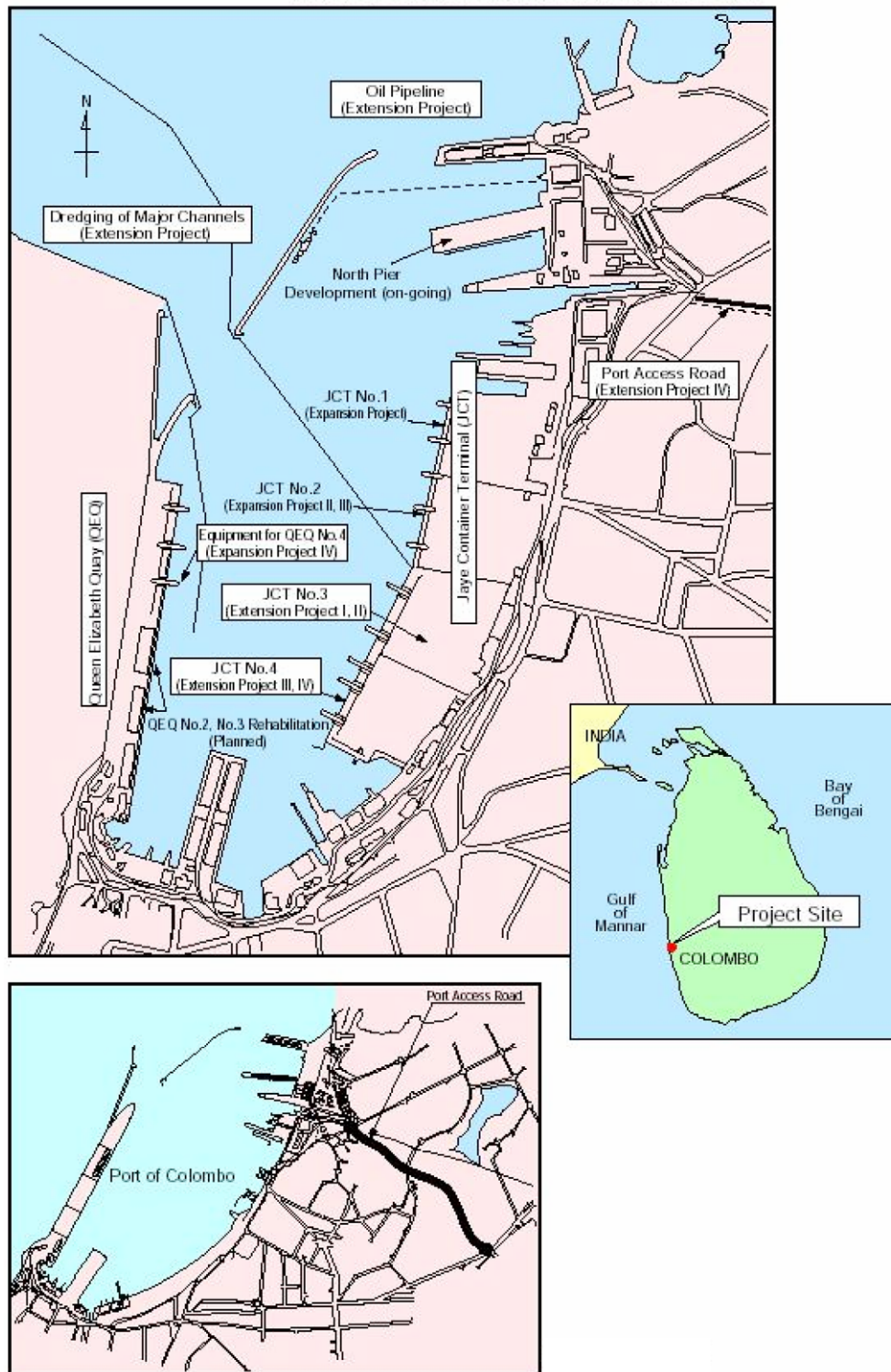


Fig. 1. Port of Colombo.

As liner shipping is characterized by fixed sailing schedules and fixed port-of-calls by named ships, minimizing the port-to-port transit time is essential; thus, the productivity of the container terminal plays a major role

Table 1
Statistics of Jaya Container Terminal in Port of Colombo

Berth #	1	2	3	4
Berth length (m)	300	322	330	330
Depth alongside (m)	12	13	13.5	14
Storage area (ha)	8.6	8.4	14	14
# of quay cranes	3	3	3	3
# of transfer cranes	5	5	–	–

in logistic chains involving container liner shipping. While one of the metrics used to measure terminal productivity is the box-handling rate per hour, a ship's waiting time for a berth is also crucial indicator of a port's efficiency as it has a significant effect on the total time (or turnaround time) spent by a ship in the port-of-call.

In berth allocation at an MUT, calling vessels are usually allocated to one of the berths managed by the terminal operator. This study addresses a berth allocation problem (with flexible allocation of ships to berths ignoring the FCFS rule) in a different operational setting as applied by the port of Colombo, Sri Lanka, where the port infrastructure is insufficient to meet growing traffic of container ships. Due to its strategic and ideal location in the Indian Ocean for container traffic between the East and Southeast Asian regions and Europe, the port of Colombo (see Fig. 1) is a major hub port and features a modern container terminal called Jaya (or Jaye) Container Terminal (JCT) and another container terminal known as South Asia Gateway Terminal (SAGT). The former has been in operation with four berths since 1980 (see Table 1), while the latter, which used to be a general cargo terminal called Queen Elisabeth Quay (QEQ), has been operating since 1996.

In a congested situation at JCT, calling or scheduled ships, which are projected to experience a long waiting time, are directed to SAGT. Though this exercise may be advantageous for shipping lines in terms of waiting time savings, it has many drawbacks for JCT. Besides the fact that it is redirecting customers elsewhere, such an exercise results in additional costs to JCT. When a ship is assigned to the external terminal (SAGT in this case), JCT has to,

- (a) haul containers between SAGT and JCT,
- (b) perform all the necessary mounting and demounting of containers in the yard in order to send/stack containers in JCT, and
- (c) perform shifting of containers in the stacks if necessary.

All these activities entail huge monetary and time costs. All the relevant costs have to be born by JCT. In addition, JCT pays significant amounts of money to SAGT for such urgent and fast ship handlings, while the details of this expenditure are not made public. Usage of the external terminal for overflowing traffic of calling ships is unavoidable at present due to the limited capital investments for enlarging the handling capacity of JCT.

In light of the above discussion, this paper addresses the berth allocation problem (BAP) at an MUT with a limited own quay capacity and usage of an external terminal.

The paper is organized as follows. The next section provides a literature review on the berth allocation planning. The problem formulation is discussed in Section 3. The fourth section introduces a solution algorithm, followed by a number of computational analyses in the fifth section. The final section concludes the paper.

2. Literature review of the BAP

As there is an ever growing demand of operating MUTs more efficiently due to the continuous increasing container traffic, the issues pertaining to the efficient berth allocation at an MUT has been receiving much attention recently.

Lai and Shih (1992) propose a heuristic algorithm for berth allocation, which is motivated by more efficient terminal (actually berth) usage in the HIT terminal of Hong Kong. Their problem considers an FCFS allocation strategy, which is not the case in our problem. Brown et al. (1994, 1997) examine ship handling in naval ports; however they cannot be applied for commercial ports because of their assumption that a berth shift

occurs when for proper services, a newly arriving ship must be assigned to a berth where another ship is already being serviced.

Imai et al. (1997) address a BAP for commercial ports. Most service queues are in general processed on the FCFS basis. They concluded that in order to achieve high port productivity, an optimal set of ship-to-berth assignments should be found without considering the FCFS rule. However, this service principle may result in certain ships being dissatisfied with the order of service. In order to deal with the two criteria to evaluate, i.e., berth performance and dissatisfaction with the order of service, they developed a heuristic to find a set of non-inferior solutions while maximizing the former and minimizing the latter. Their study assumes a static situation where ships to be served for a planning horizon have all arrived at a port before one plans the berth allocation. Thus, their study can be applied only to extremely busy ports. As far as container shipping is concerned, such busy ports are not competitive and this type of situation is not “realistic” due to the long delays experienced in the interchange process at the ports of call. In this context, Imai et al. (2001, 2005a) extend the static version of the BAP to a dynamic treatment that is similar to the static treatment, but with the difference that some ships arrive while work is in progress. As the first step in this dynamic treatment, only one objective, berth performance, is considered. Due to the difficulty in finding an exact solution, they developed a heuristic by using a subgradient method with Lagrangian relaxation. Their study assumes 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 extend the dynamic version of the BAP for the multi-water depth configuration. They employ genetic algorithms 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 are priority, as handling work associated with them does not keep the other big vessels waiting 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. From this point of view, Imai et al. (2003) extend the BAP of Imai et al. (2001, 2005a) to treat the ships with different priorities and see how the extended BAP differentiates the handling of ship in terms of the service time associated with ships. Furthermore, Imai et al. (in press) recently introduce the BAP for an indented terminal, which is considered as a possible efficient container terminal for prospective mega container ships. They developed a genetic algorithm based heuristic for a BAP at an indented terminal and also made comparisons in terminal performance between indented and conventional terminals.

There is also another type of the BAP, which is the one with a continuous location index (referred to as BAPC). While in the above mentioned studies the entire terminal space is partitioned into several parts (or berths) and the allocation is planned based on the divided berth space, under this 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 wait and in service at a berth can be shown by a rectangle in a time–space representation or Gantt chart, 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 that no rotation of ship rectangles is allowed. Lim (1998) addresses 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 assumes that every ship is berthed as soon as it arrives at the port. On the other hand, Li et al. (1998) solve the BAPC both with and without the ship’s movement restriction. Their objective is to minimize the makespan of the schedule. Guan et al. (2002) developed a heuristic for the BAPC with the objective that minimizes the total weighted completion time of ship services. Park and Kim (2002) study the BAPC with an objective that minimizes the costs of delayed departures of ships due to the undesirable service order and those of additional complexity in handling containers when ships are served at non-optimal mooring locations in port. Their work is more practical than the aforementioned BAPC studies since the factors assessed in the objective depend on the quay locations of ships. Kim and Moon (2003) address the same BAPC as the one tackled by Park and Kim (2002), though the former study employs the simulated annealing method while the latter one applies the subgradient optimization method. Notice that all the above BAPC studies assume the independent ship handling time from the berthing location. Park and Kim (2003) study the BAPC with a similar objective to those of Park and Kim (2002), and Kim and Moon (2003). The difference in objective is that in addition to the costs considered in previous studies, Park and Kim (2003) take into account the cost resulting from early or late start of ship handling against

the estimated times of ship arrival. Their study features an interesting characteristic in that they determine the optimal start times of ship service and associated mooring locations and at the same time they determine the optimal assignment of quay cranes to those ships. In their study, the handling time of a particular ship is a function of the number of quay cranes engaged in the ship; however, the handling time is independent from the mooring location of the ship. Imai et al. (2005b) address a BAPC, which noticeably has a major difference from the other BAPCs as they consider that the handling time depends on the ship's berthing location. They developed a heuristic for that problem in conjunction with a heuristic for the dynamic BAP in Imai et al. (2001, 2005a). The conclusion of their study is that for ship lengths that follow a uniform distribution the best approximate solution is identified with the best solution in discrete location where the berth length is 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. Cordeau et al. (2005) develop a tabu search heuristic for the dynamic BAP in two versions with both discrete and continuous location indexes. They analyze the solution quality of the proposed heuristic for the discrete location with the exact solution by CPLEX; however, the applied problem instances are relatively small sized ones. For the continuous location version, the solution quality is assessed by comparisons with solutions by the straightforward heuristic.

3. Problem formulation

3.1. Assumptions

This study develops a BAP with usage of an external terminal based on the static and dynamic versions of the BAP in Imai et al. (2001, 2005a), which find the optimal assignment of ship-berth-service order with the minimization of the total service time (service time is comprised of ship's wait time for berth availability and handling time at the allocated berth). For convenience, we hereafter refer to them as the Static Berth Allocation Problem with an External Terminal (SBAPE) and the Dynamic Berth Allocation Problem with an External Terminal (DBAPE), respectively.

In the example described in Section 1, the terminal operator defines a time limit and switches from his own terminal to the external one, those ships that are expected to wait more than the time limit. This study assumes that these ships have their handling work started at the external terminal as soon as possible, as otherwise there would be no compelling reason for using the external terminal. In fact the time to start their handling at the external terminal depends on how congested the external terminal is; however, in reality it is known from experience that they are not kept waiting longer than their estimated waiting time at the own terminal. Therefore, the operator is relieved from concerns regarding the externally allocated ships. In the case though, that the operator gets involved in the external terminal's situation regarding to how long ships allocated to the external terminal wait for handling and how fast they are served, then both own and external terminals have to jointly consider the comprehensive berth allocation. This berth allocation may be scheduled by the existing berth allocation studies as reviewed in the previous section. In fact, the operator of JCT plans berth allocation without taking into account wait and service time of ships (including moved ships from JCT to SAGT) at SAGT.

The terminal operator pays service charges to the external terminal based on the total handling time of ships served at the external terminal (we hereafter refer to the total handling time as the total service time at the external terminal since the waiting time is being ignored). The operator basically aims at maximizing the total workload allocated to own terminal during a specific time period by minimizing the total service time of ships, resulting in less usage of the external terminal. If the operator sets an extraordinary long time limit, then no ships are likely to be allocated to the external terminal; this however does not solve the problem because of the low service quality resulting from the long wait for idle berths at his own terminal. The terminal desires to see how the total service time, which is a metric of the satisfaction of the calling vessels, and the imposed service costs of allocating vessels to an external terminal fluctuate by varying the time limit.

Our final goal is to analyze the DBAPE; however, the DBAPE has a complicated formulation and is an extension of the SBAPE. Therefore, for easier understanding of the formulation of DBAPE, we first formulate

the SBAPE. Except for the assumptions regarding the external terminal usage, all others are the same as those described in Imai et al. (2001, 2005a), and are summarized as follows:

- (1) The berth allocation ignores the FCFS rule.
- (2) Each berth can handle one ship at a time.
- (3) The handling time of a ship depends on its allocated berth.
- (4) The handling of a ship continuously performs without any interruption.

3.2. SBAPE formulation

Assuming all the ships have arrived at a port before we start the berth allocation scheduling, i.e., $A_j \leq S_i$, the following may be the formulation of the SBAPE:

$$[S] \quad \text{Minimize} \quad H = \sum_{j \in V} \sum_{k \in U} C_{Qj} x_{Qjk} \quad (1)$$

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

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

$$\sum_{k \in U} \left\{ S_i - A_j + \sum_{l \in V} \sum_{m \in P_k} C_{il} x_{ilm} + (x_{ijk} - 1)TM \right\} \leq L_j \quad \forall i (\neq Q) \in B, \quad j \in V, \quad (4)$$

$$x_{ijk} \in \{0, 1\} \quad \forall i \in B, \quad j \in V, \quad k \in U, \quad (5)$$

where

$i(=1, \dots, Q) \in B$ set of berths where Q is the external berth (or terminal)

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

$k(=1, \dots, T) \in U$ set of service orders

TM a large value of the time enough to cover the planning horizon

L_j limit of waiting time for ship j

A_j arrival time of ship j

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

S_i time when berth location i becomes idle for the planning horizon

C_{ij} handling time spent by ship j at berth i

x_{ijk} =1 if ship j is served as the $(T - k + 1)$ th last ship at berth i , and =0 otherwise

x_{ijk} s are decision variables. As stated above, x_{ijk} defines the relationship between berth, ship and service order. Fig. 2 graphically illustrates this relationship for a specific berth. For any berth allocation strategy, each berth needs to have T service orders, while some berths may be unoccupied. In Fig. 2 with $T = 10$, ship 3 is scheduled to be served as the second ship (or the ninth last ship). This is ensured by putting $k = 2$ into $(T - k + 1)$, thus we obtain the “ninth” last ship with respect to the service order.

Objective (1) minimizes the total service time of ships that are allocated to the external terminal. Constraint set (2) ensures that every ship must be served at some berth in any order of service. Constraints (3) enforce that every berth serves up to one ship at any time. Constraints (4) assure that ships do not wait more than the time limit when they are served at the own terminal. The scheduling of berth allocation, i.e., in what order of service

k	1	2	3	4	5						$T (=10)$
j	4	3	7	2	1	-	-	-	-	-	

Fig. 2. Berth allocation representation.

k a specific ship j is handled at berth i , is determined by x_{ijk} . The time duration that the $(T - k + 1)$ th last ship has already been at the port (i.e., the waiting time for the assigned idle berth) is defined by the first to third terms in the parenthesis of the left-hand side of (4), $S_i - A_j + \sum_{l \in V} \sum_{m \in P_k} C_{il} x_{ilm}$. If ship j is served as the $(T - k + 1)$ th last ship, then the last term, $(x_{ijk} - 1)TM$, becomes zero because of $x_{ijk} = 1$ so that the left-hand side of (4) implies the waiting time of ship j . On the other hand, if ship j is not served as the $(T - k + 1)$ th last ship, $(x_{ijk} - 1)TM$ becomes negatively enormous with the resulting effect that a constraint (4) of ship j is always satisfied. For the derivation of the first to third terms of constraints (4), i.e., $S_i - A_j + \sum_{l \in V} \sum_{m \in P_k} C_{il} x_{ilm}$, see Imai et al. (2001, 2005a). Note that there are no constraints on the waiting time limit for ships allocated to the external terminal, because no wait is assumed at the external terminal.

In this paper, we only focus on the DBAPE that follows the SBAPE; since in the SBAPE there may be some ships waiting for a long time at the moment of planning, resulting in an infeasible solution of the SBAPE in light of the limit on waiting time. Nevertheless, while this study discusses the genetic algorithm (GA) for a solution method as will be described later, it may be worth to consider the use of the subgradient optimization with the Lagrangian relaxation for solving the SBAPE, as proposed for the dynamic version of the BAP in Imai et al. (2001, 2005a), due to the similarity in model structures between the SBAPE and the dynamic version of the BAP.

3.3. DBAPE formulation

Assuming that all ships have not necessarily arrived at a port before scheduling the berth allocation, the DBAPE requires additional constraints, which ensure that ships are to be handled at allocated berths after their arrival.

$$[D] \quad \text{Minimize} \quad H = \sum_{j \in V} \sum_{k \in U} C_{Qj} x_{Qjk} \quad (6)$$

$$\text{subject to} \quad \sum_{i \in B} \sum_{k \in U} x_{ijk} = 1 \quad \forall j \in V, \quad (7)$$

$$\sum_{j \in V} x_{ijk} \leq 1 \quad \forall i \in B, \quad k \in U, \quad (8)$$

$$\sum_{k \in U} \left\{ S_i - A_j + \sum_{l \in V} \sum_{m \in P_k} (C_{il} x_{ilm} + y_{ilm}) + y_{ijk} + (x_{ijk} - 1)TM \right\} \leq L_j \quad (9)$$

$$\forall i (\neq Q) \in B, \quad j \in V,$$

$$\sum_{l \in V} \sum_{m \in P_k} (C_{il} x_{ilm} + y_{ilm}) + y_{ijk} - (A_j - S_i)x_{ijk} \geq 0 \quad \forall i \in B, \quad j \in W_i, \quad k \in U, \quad (10)$$

$$x_{ijk} \in \{0, 1\} \quad \forall i \in B, \quad j \in V, \quad k \in U, \quad (11)$$

$$y_{ijk} \geq 0 \quad \forall i \in B, \quad j \in V, \quad k \in U, \quad (12)$$

where

W_i subset of ships with $A_j \geq S_i$

y_{ijk} idle time of berth i between the departure of the $(T - k + 2)$ th last ship and the arrival of the $(T - k + 1)$ th last ship when ship j is served as the $(T - k + 1)$ th last ship

x_{ijk} s and y_{ijk} s are decision variables.

Similarly to constraints (4), constraint set (9) assures that ships do not wait more than the time limit when they are served at the own terminal. Note that this set is different from set (4), as the late arrival of a ship is taken into account. A ship that arrives at the port during the planning horizon may be scheduled to be moored at a berth in some hours after its immediately predecessor at the same berth has departed. This time gap, y_{ijk} , is considered in constraint set (9). Constraint set (10) enforces that no ships start their handling before their arrival. See Imai et al. (2001, 2005a) for the derivation of constraints (10). Constraints (9) can be easily derived from constraints (10), with the notion of constraints (4) of the SBAPE.

4. Solution procedure

We develop a heuristic for the DBAPE, since there is not likely an efficient exact solution procedure, which finds an optimal solution in polynomially-bounded computation time for the DBAPE. The procedure we employ for the heuristic is the GA. As GAs are widely applied for plenty of practical problems of mathematical programming, which are difficult to solve in terms of polynomially-bounded computation time, we do not explain the GAs in detail. Refer to [Reeves \(1993\)](#) for details.

4.1. Representation

Instead of using the classical binary bit string representation, the chromosomes are represented as character strings. [Fig. 3](#) states the berth allocation. [Fig. 3\(a\)](#) shows a typical chromosome representation for berth allocation with two berths. The length of the string of digits for the chromosome is the number of ships plus the number of berths minus one, where sets of ships allocated to berths are separated by zeros. Ships allocated to a specific berth are placed in order of berthing from left to right. In [Fig. 3\(a\)](#), ships 2, 1, 4 and 7 are served at berth 1 in order of service as they appear, whilst ships 3, 5, 6, 8 and 9 at berth 2. The associated waiting times as well as the assigned berths and service orders for those ships in computational progress are shown in [Fig. 3\(b\)](#). Given the limit of waiting time as 5 h, the result is given as [Fig. 3\(a\)](#), but interpretation with respect to allocated berths is given in [Fig. 3\(c\)](#), where ships 4 and 9, which were supposed to be served at the own terminal, are allocated to the external terminal.

4.2. Fitness

A selection criterion is used for picking the two parents to apply the crossover operator. The appropriateness of a selection criterion for a GA depends on the other GA operators chosen. A typical selection criterion gives a higher priority to fitter individuals and this leads to a faster convergence of the GA. The DBAPE is a minimization problem; thus, the smaller the objective function value is, the higher the fitness value must be. For this, the fitness function can be defined by the reciprocal of objective function as done in [Kim and Kim \(1996\)](#). Another alternative is a sigmoid function as used in [Nishimura et al. \(2001\)](#) for a GA heuristic so as to find a near optimal solution to a BAP, where multiple vessels may be served at a specific berth. The former was selected for the DBAPE, based on the result of a preliminary experiment.

a	Cell #	1	2	3	4	5	6	7	8	9	10
	Representation	2	1	4	7	0	3	5	6	8	9

b	Berth #	1	1	1	1	-	2	2	2	2	2
	Order of berthing	1	2	3	4	-	1	2	3	4	5
	Waiting time	1	4	6	0	-	0	3	2	4	7

c	Berth #	1	1	Ex	1	-	2	2	2	2	Ex
	Order of berthing	1	2	-	3	-	1	2	3	4	-

Ex: External terminal

Fig. 3. Chromosome representation (a), waiting for ships in progress (b) and resulting berth # and order of berthing for ships (c).

4.3. Crossover

As the reproduced chromosomes constitute a new population, crossover is performed to introduce new chromosomes (or children) by recombining current genes. The crossover operator we employed is the so-called *two-point crossover*. A crossover may generate infeasible children in terms of constraint set (2), i.e., a child chromosome may have no ships served or ships served twice. In order to keep the feasibility, the crossover operation is performed in the following manner. Fig. 4 shows an example where new children are created by crossover. First, substrings to be interchanged are given by two crossover points that are randomly selected. Then, we obtain chromosomes C' and D' as temporary children after interchanging the substrings between chromosomes A and B. They are infeasible, because for instance, chromosome C' has ships 1, 2, and 7 to be serviced twice while ships 0, 4 and 9 are not served. Next, additional interchanges described below are carried out to make them feasible. Letting the *interchanged string* be the substring that was interchanged so far in each chromosome, we examine genes from left to right in the interchanged string of C' as demonstrated as follows. Note that we never interchange any genes in the interchanged string again in the following process.

- (i) As the outermost left gene is ship 1 which is scheduled to be served twice in C' , obtain the ship number (which is 5) in D' at the corresponding position of the gene (hereafter referred to as a *cell*). There are two possibilities: a ship 5 in C' , and no ship 5 in C' . Since ship 5 exists in C' for this example, find a cell which has it. The corresponding cell of D' has ship 0 (which is a berth separator). Thus, switch ships between the cell of C' which has ship 1 and the one of D' which holds ship 0 not in the interchanged string.
- (ii) For D' ship 9 is located in the cell corresponding to ship 2 in the interchanged string of C' . Therefore, interchange ship 2 of C' and ship 9 of D' , both not in the interchanged string.
- (iii) Ship 5 was already examined. Ship 4 in D' corresponds to ship 7 in C' . As ship 4 does not exist in C' , exchange ship 7 in C' and ship 4 in D' , both are not in the interchanged string.

A more formal description of the procedure is as follows, where k denotes the cell number ($k = 1, \dots, LK$) in the exchanged string, and $d(k)$ is a digit in cell k .

Step 1: Let $k = 1$.

Step 2: If $k > LK$, then go to Step 3, otherwise go to Step 4.

Step 3: If chromosomes C' and D' are feasible, then STOP. Otherwise, treat chromosomes C' as D' and D' as C' in the following steps, and let $k = 1$.

Step 4: If another cell in C' has the value of $d(k)$, let $k' = k$ and go to Step 5. Otherwise, go to Step 6.

Step 5: If there are two cells in D' having the value of $d(k')$, store the value to the cell of C' not in the interchanged string and go to Step 6. Otherwise, find a cell in C' having the value of $d(k')$ of D' , let k' be the cell number, and go to Step 5.

Step 6: Let $k = k+1$ and go to Step 2.

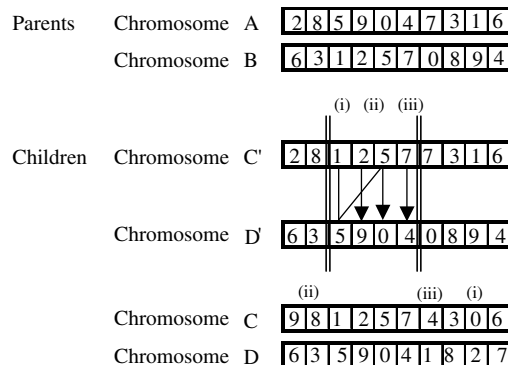


Fig. 4. Crossover operation.

4.4. Tournament process

We apply the tournament process, which [Ahuja et al. \(2000\)](#) proposed for a better solution. One can apply a GA many times starting with different populations and choose the best individual obtained among all the runs. In order to save substantial run time, as an alternative they take the final population of two different runs, keep the best 50% of the individuals from these two runs, and apply the GA again with this mixed population as the initial population.

4.5. Waiting time computation

As an obtained chromosome, shown in [Fig. 3\(a\)](#), does not take into consideration the waiting time violation, we check the violation based on the chromosome, then proceed with recalculating start and completion times of ships that remain at the own terminal.

4.6. Parameter setting

The number of individuals in a generation is set to 40 and the mutation rate is set to 0.08. The GA computation performs with the limit of 6000 generations.

5. Numerical experiments

Before referring to the primary experiments, we discuss key data on arrival and handling of 56 ships calling at JCT of the port of Colombo for 10 days in June 2003. [Fig. 5](#) shows the distribution of observed interval of arriving ships, which follows an exponential function, as generally observed in many practices. The distribution of observed handling time of ship, as portrayed in [Fig. 6](#), follows a 7-Erlangian function. Note that the handling time may include the waiting time as the waiting time itself was not able to be identified in the data. Therefore, we analyze the data, assuming that no ships wait for idle berths at the JCT in reality; however, this assumption does not seem to lose generality since the resulting distribution follows the Erlangian function as often observed in service practices.

The problem assumes that each berth handles a single ship at any berth at a time. In fact, the 10-day ship service data at Colombo contain multiple ships being served at the same berth. There are four cases of multiple-ship berthing that involve only eight ships in total out of a total of 56 calling ships at the port. Therefore, the assumption is substantially justified by the data.

The authors have already addressed another type of the BAP, which is the dynamic version of the BAP (referred to as the DBAP and described in [Appendix A](#)) where all the ships in the system are served at the own terminal (or berths). While its objective function is different from the one of the DBAPE, the DBAP may work to generate the solution to the DBAPE. This is because the DBAP aims to minimize the total service

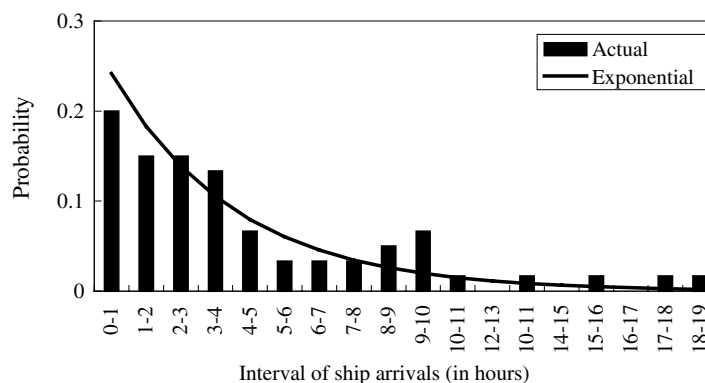


Fig. 5. Distribution of ship arrival at Port of Colombo.

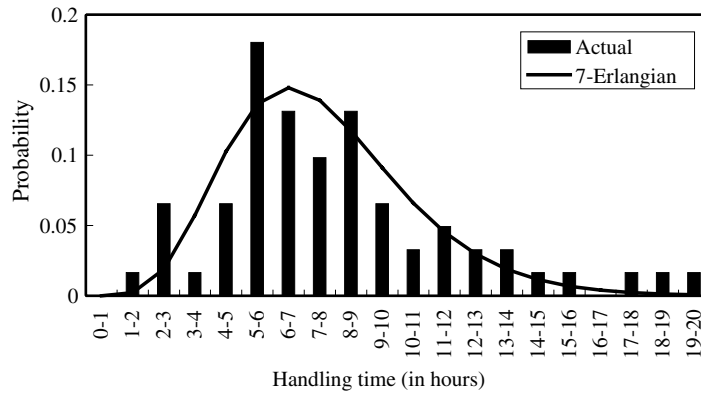


Fig. 6. Distribution of ship handling time at Port of Colombo.

time and therefore implicitly the waiting time of the ships. To look into such a possible application of the DBAP, we compare the solutions of the DBAP with those of the DBAPE. Note however that as the DBAP does not let any ships be served at the external terminal, the solution has to be modified to cope with the problem setting of the DBAPE as follows:

[Modification of the solution to the DBAP]

- Step 1: Pick up the first ship, in terms of the start of the service, in the solution to the DBAP. If no ship is left in the solution, then STOP.
- Step 2: If the ship is expected to wait for over its L_j , then assign the ship to the external terminal. Remove the ship from the solution to the DBAP.
- Step 3: Start service of the subsequent ship if it exists, as early as possible, in the berth to which the previous ship was originally allocated in the DBAP solution. Go to Step 1.

A better solution may be identified when using the DBAP if in the above modification, a DBAP computation is performed every time a ship has been assigned to the external terminal. However, this procedure was not applied in this study for the following reasons: (i) if a number of ships are removed as they are sent to the external terminal, the DBAP is computed many times with considerable computational burdens, (ii) otherwise, the DBAP computation is performed fewer times and no significant improvement in the solution is expected.

The DBAPE and DBAP are coded in “C” and FORTRAN77 languages on Sun SPARC64 GP (275 MHz) workstation, respectively. We generate a data set for computation, by using the 10 day-statistics from the port of Colombo. As, for computational experiments, we need to generate the handling times of a ship, which would have been supposed to be consumed when the ship had been served at the other berths, we assume longer handling times (three hours longer at most) at those berths than at the actually-used berth based on experiences at the port of Colombo. This in turn implies that the handling time of each ship observed in the statistics is the minimum one among all the berths.

As pointed out in Nishimura et al. (2001), the GA is a heuristic and therefore the solution quality may depend on the problem size (the numbers of ships and berths for the BAP). Consequently, we solve the problem, both by the DBAPE and DBAP, with four different schemes: solving it as a whole with a 10-day span, solving it as two sub-problems with a 5-day span for each, as three sub-problems with a 3-day span, and as five sub-problems with a 2-day span.

The limit of waiting time at the own terminal is defined as the following equation: $L_j = \alpha \times \{\min_i C_{ij}\}$, where α is a parameter. This equation is applied as a sensible measure, based on interviews with JCT.

The total service time of ships at the own and external terminals and the total waiting time of ships at the own terminal by the four different solving schemes are shown in Fig. 7, while the average service and waiting times per ship at the own terminal and the average service time per ship at the external terminal are portrayed in Fig. 8. The average times at the own and external terminals are calculated as the total of those values

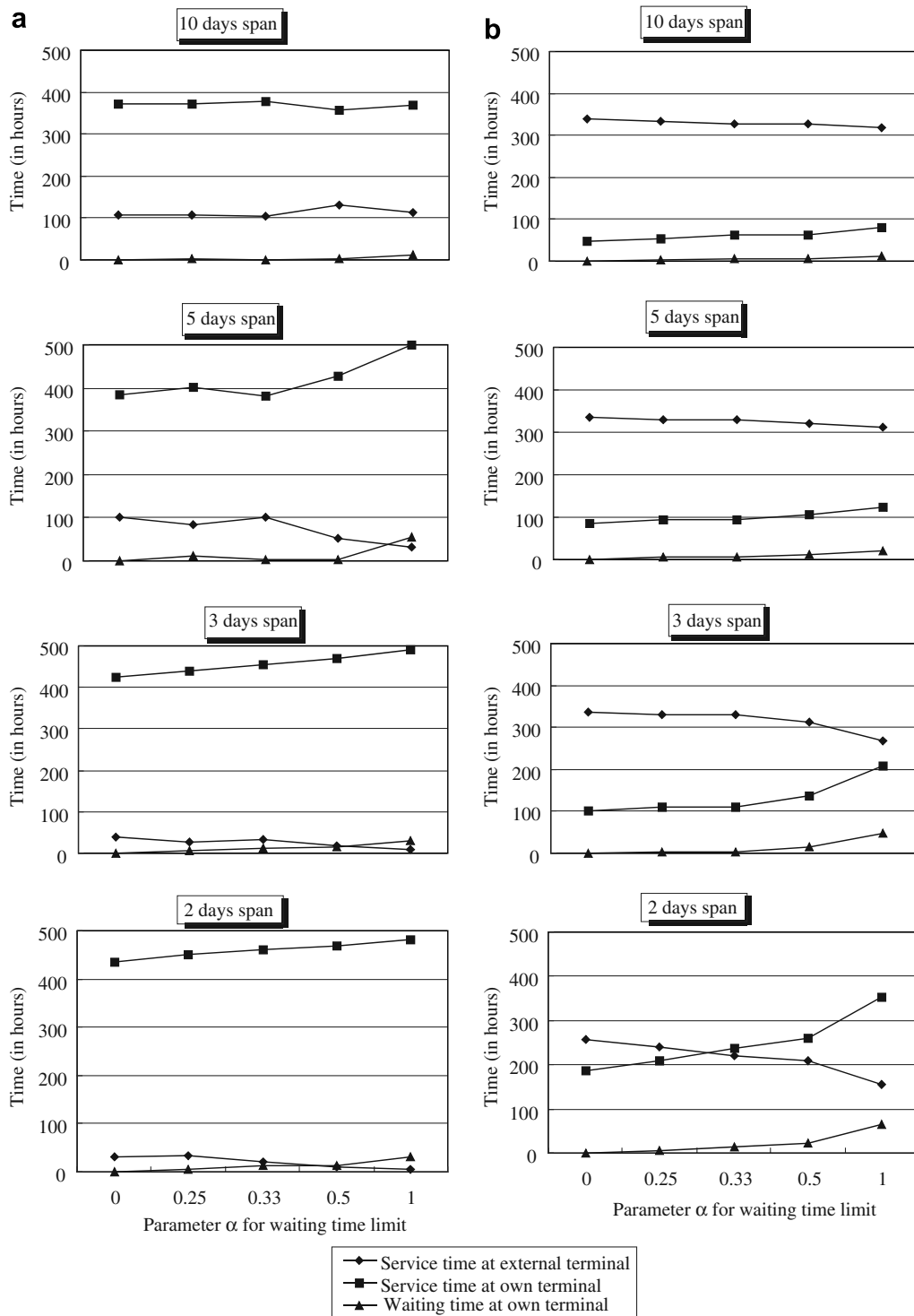


Fig. 7. Total service and waiting times: (a) solutions by the DBAPE and (b) solutions by the DBAP.

divided by the number of ships at the respective terminals. Note that normally the total and average values should have an identical trend because the average value is derived from the division of the total service (or waiting) time by the number of ships. However, this may not be the case in the problem of our concern

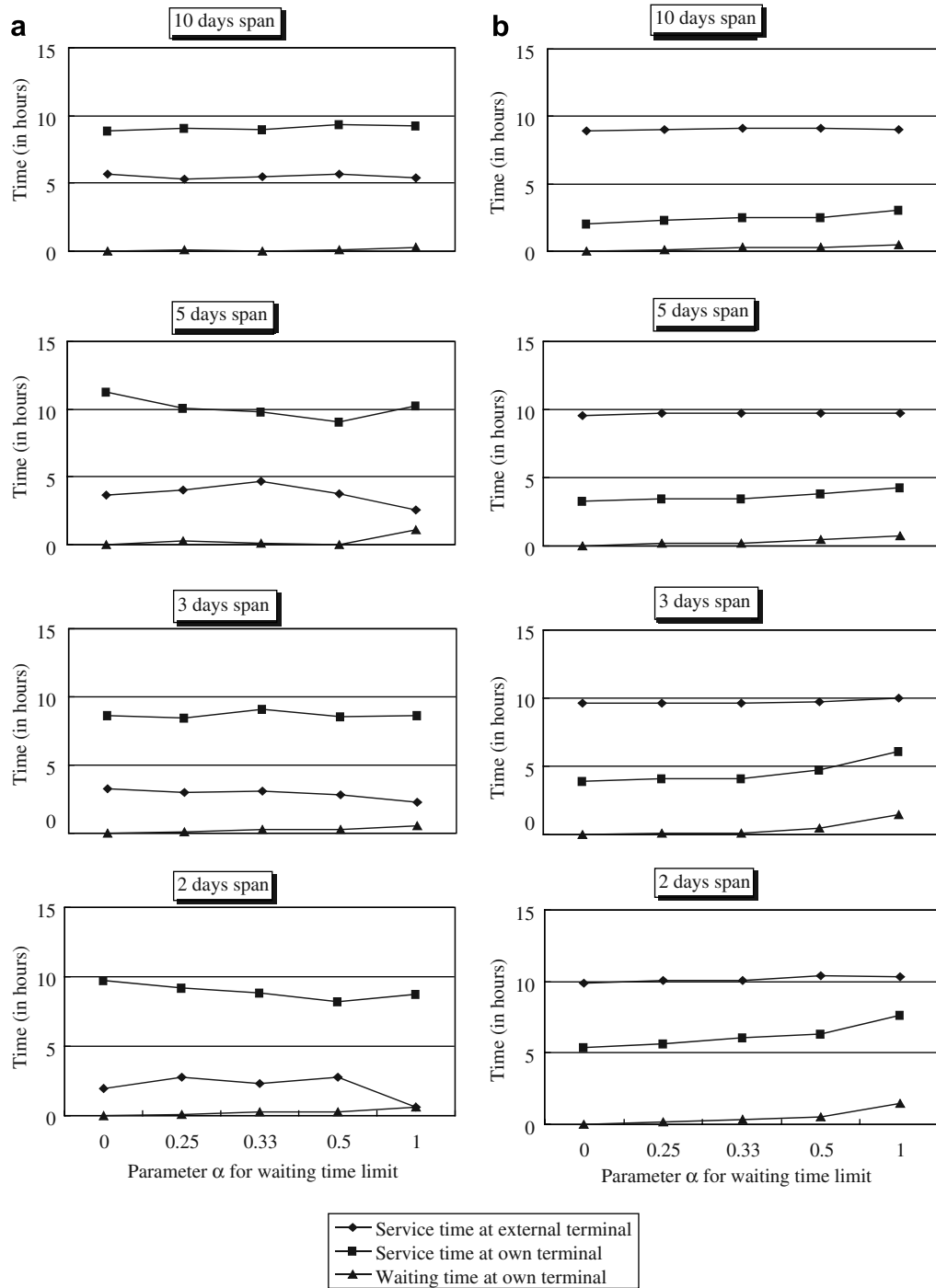


Fig. 8. Average service and waiting times per ship: (a) solutions by the DBAPE and (b) solutions by the DBAP.

since we may have different numbers of ships at the own and external terminals; therefore, as observed in Figs. 7 and 8, the trends of the values are not really identical between the figures.

Comparing the solutions of the DBAPE and DBAP in Fig. 7, the trends are completely different except for the waiting time, contrary to our expectations. Basically it is logical that both procedures should have a similar trend in the waiting time since ships are assigned to the external terminal, so that they should not wait for over

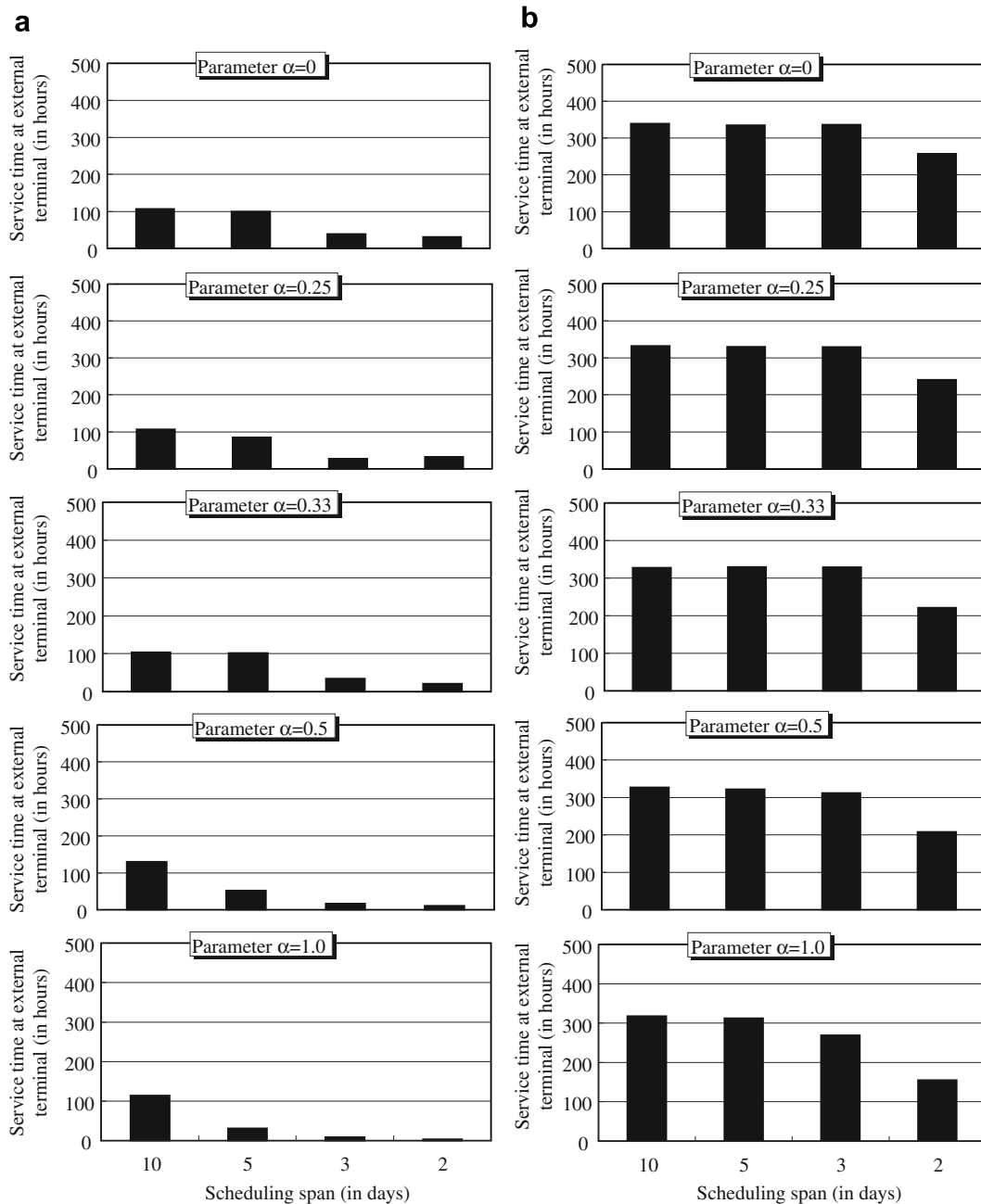


Fig. 9. Total service time at the external terminal by scheduling span: (a) solutions by the DBAPE and (b) solutions by the DBAP.

L_j . However, both procedures have different trends in the service times at the two terminals. The DBAPE produces smaller service times at the external and larger times at the own terminal than the DBAP. This observation suggests that the problem should be solved by the DBAPE, which outperforms the alternative method based on the heuristic for the DBAP. A similar trend can be seen in the average values shown in Fig. 8. Surprisingly, on average a ship spends 5 h less at the external terminal by the DBAPE than by the DBAP. In addition, it waits for less time by the DBAPE than by the DBAP. The superiority of the DBAPE is typically shown in the case of $\alpha = 1$ with a 2-day span, where the service time at the external terminal and the waiting

time at the own terminal are smaller by the DBAPE whilst the service time at the own terminal is nearly the same.

Regarding the influence of L_j (actually α), the total service time at the external, except for the case of a 10-day span, decreases by increasing α , for both the DBAPE and DBAP. With respect to the charge for the external terminal usage, one has to assess the berth scheduling by the total service time. According to Fig. 7(a), the best scheduling is identified by the problem with 2-day span sub-problems among others. If the operator is ready to pay the charge to the external terminal regardless of its amount with the assumption of the charge being proportional to the service time, α should be zero with the 2-day span because $\alpha = 0$ results in the minimum total service time at the own terminal whilst there is no difference in the service time at the external terminal between $\alpha = 0$ and 0.25. On the other hand, if the operator desires to minimize the charge, α should be 0.5 or 1, which give null or nearly null value of the total service time at the external terminal in Fig. 7(a). From the ships' point of view, there is no superior parameter α since the total of service times at the own and external terminals are nearly constant by different α s in the case of the 2-day span of Fig. 7(a). However, if the schedule for the ships is evaluated by equity in average service times per ship at own and external terminals, $\alpha = 0.5$ is the best in the 2-day span of Fig. 8(a) because of the minimum deference between the two terminals.

Next, we look into the solution quality (in terms of the total service time of the external terminal) of the DBAPE by different scheduling spans. Fig. 9 summarizes the total service times of the external terminal by both DBAPE and DBAP with different scheduling spans which are already shown in Fig. 7. The solution with a longer scheduling span is worse than the one with a smaller span. Nevertheless, improvement of the solution quality is converged at a 3-day span. As expected, the solution quality decreases with an increasing size of the problem due to the heuristic being applied. However, if the scheduling is made with very small interval such as every few hours, it corresponds to the berth allocation with the FCFS principle with poor scheduling consequences. Thus, the quality trend likely forms a U-shape curve along with the span axis. Such a trend is apparent in the case of $\alpha = 0.25$. For the DBAP, while not so clear, there is the same trend in the solution quality, in terms of the service time at the external, at 2- and 3-day spans.

6. Conclusions

In this study, we examined a berth allocation problem under exceptional circumstances, which are almost likely to be observed in busy container ports in developing countries. The situation that surrounds the problem we addressed in this paper, is that a terminal operator manages the berth allocation efficiently and allocates ships, whose expected waiting time for service at his own terminal will exceed a specific time, to an external terminal. As additional costs are incurred to the terminal for utilization of the external terminal, the berth allocation should be planned as to serve as many ships as possible in his own terminal with their waiting time less than the set time limit. We developed a GA-based heuristic for the problem (called the DBAPE). It is possible to modify the solution of the DBAP, which the authors have already implemented for a different problem setting, in order to solve the DBAPE. However, according to the numerical experiments using the statistics of the port of Colombo, as expected, the heuristic that was elaborated for the DBAPE outperformed the procedure using the DBAP. The solution procedure we developed in this study is helpful in efficient management of extremely busy container terminals in developing countries.

Acknowledgements

The authors would like to thank the referee for insightful suggestions, which improved this paper to a great extent. The authors also express a sincere gratitude to Mr. Kurihara, a former undergraduate student of Kobe University, for implementation of the algorithms, and to Mr. Kumara of Jaya Container Terminal at the port of Colombo, a graduate of the World Maritime University, for ship calling statistics of that terminal. This research is partially supported by the Grant-in-Aid for Scientific Research of Japan under Scientific Research-C Grant 17510123.

Appendix A. Formulation of the DBAP

$$[\text{DBAP}] \quad \text{Minimize} \quad \sum_{i \in B} \sum_{j \in V} \sum_{k \in U} \{(T - k + 1)C_{ij} + S_i - A_j\}x_{ijk} + \sum_{i \in B} \sum_{j \in W_i} \sum_{k \in U} (T - k + 1)y_{ijk} \quad (\text{A.1})$$

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

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

$$\sum_{l \in V} \sum_{m \in P_k} (C_{il}x_{ilm} + y_{ilm}) + y_{ijk} - (A_j - S_i)x_{ijk} \geq 0 \quad \forall i \in B, \quad j \in W_i, \quad k \in U, \quad (\text{A.4})$$

$$x_{ijk} \in \{0, 1\} \quad \forall i \in B, \quad j \in V, \quad k \in U, \quad (\text{A.5})$$

$$y_{ijk} \geq 0 \quad \forall i \in B, \quad j \in V, \quad k \in U. \quad (\text{A.6})$$

The decision variables are x_{ijk} s and y_{ijk} s; where as defined in SBAPE and DBAPE x_{ijk} is 1 if ship j is served as the $(T - k + 1)$ th last ship at berth i and is 0 otherwise, and y_{ijk} is an idle time of berth i between services of the $(T - k + 2)$ th last ship and the next ship. Subsets P_k and W_i are also defined in the SBAPE and DBAPE. Objective (A.1) minimizes the total service time. Constraint set (A.2) ensures that every ship must be served at some berth in any order of service. Constraints (A.3) enforce that every berth serves up to one ship at any time. Constraints (A.4) assure that ships are served after their arrival. For the detailed derivation of the formulation, see Imai et al. (2001, 2005a).

References

- Ahuja, R.K., Orlin, J.B., Tiwari, A., 2000. A greedy genetic algorithm for the quadratic assignment problem. *Computers and Operations Research* 27, 917–934.
- Brown, G.G., Lawphongpanich, S., Thurman, K.P., 1994. Optimizing ship berthing. *Naval Research Logistics* 41, 1–15.
- Brown, G.G., Cormican, K.J., Lawphongpanich, S., Widdis, D.B., 1997. Optimizing submarine berthing with a persistence incentive. *Naval Research Logistics* 44, 301–318.
- Cordeau, J.-F., Laporte, G., Legato, P., Moccia, L., 2005. Models and tabu search heuristics for the berth-allocation problem. *Transportation Science* 39, 526–538.
- Guan, Y., Xiao, W.-Q., Chueng, R.K., Li, C.-L., 2002. A multiprocessor task scheduling model for berth allocation: Heuristic and worst case analysis. *Operations Research Letter* 30, 343–350.
- Imai, A., Nagaiwa, K., 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., Papadimitriou, S., 2001. The dynamic berth allocation problem for a container port. *Transportation Research Part B* 35, 401–417.
- Imai, A., Nishimura, E., Papadimitriou, S., 2003. Berth allocation with service priority. *Transportation Research Part B* 37, 437–457.
- Imai, A., Nishimura, E., 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., Nishimura, E., 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., Papadimitriou, S., in press. Berth allocation at indented berths for mega-containerships. *European Journal of Operational Research*.
- Kim, J.-U., Kim, Y.-D., 1996. Simulated annealing and genetic algorithms for scheduling products with multi-level product structure. *Computers and Operations Research* 23, 857–868.
- Kim, K.H., Moon, K.C., 2003. Berth scheduling by simulated annealing. *Transportation Research Part B* 37, 541–560.
- Lai, K.K., Shih, K., 1992. A study of container berth allocation. *Journal of Advanced Transportation* 26, 45–60.
- Li, C.-L., Cai, X., Lee, C.-Y., 1998. Scheduling with multiple-job-on-one-processor pattern. *IEEE Transactions* 30, 433–445.
- Lim, A., 1998. The berth planning problem. *Operations Research Letters* 22, 105–110.
- Nishimura, E., Imai, A., 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., Kim, K.H., 2002. Berth scheduling for container terminals by using a sub-gradient optimization technique. *Journal of the Operational Research Society* 53, 1054–1062.
- Park, Y.-M., Kim, K.H., 2003. A scheduling method for berth and quay cranes. *OR Spectrum* 25, 1–23.
- Reeves, C.R., 1993. *Modern Heuristic Techniques for Combinatorial Problems*. John Wiley & Sons, New York, NY.