



The simultaneous berth and quay crane allocation problem

Imai, Akio

Chena, Hsieh Chia

Nishimura, Etsuko

Papadimitriou, Stratos

(Citation)

Transportation Research Part E: Logistics and Transportation Review, 44(5):900-920

(Issue Date)

2008-09

(Resource Type)

journal article

(Version)

Accepted Manuscript

(URL)

<https://hdl.handle.net/20.500.14094/90000737>



The simultaneous berth and quay crane allocation problem

Akio Imai

Faculty of Maritime Sciences

Kobe University

Fukae, Higashinada, Kobe 658-0022 Japan

&

World Maritime University, PO Box 500, S-201 24 Malmö, Sweden

Hsieh Chia Chen

Faculty of Maritime Sciences

Kobe University

Fukae, Higashinada, Kobe 658-0022 Japan

Etsuko Nishimura

Faculty of Maritime Sciences

Kobe University

Fukae, Higashinada, Kobe 658-0022 Japan

and

Stratos Papadimitriou

Department of Maritime Studies, University of Piraeus

80 Karaoli & Dimitriou Str., GR185 32 Piraeus, Greece

Abstract

This paper addresses efficient berth and crane allocation scheduling at a multi-user container terminal. First, we introduce a formulation for the simultaneous berth and crane allocation problem. Next, by employing genetic algorithm we develop a heuristic to find an approximate solution for the problem. The fitness value of a chromosome is obtained by crane transfer scheduling across berths, which is determined by a maximum flow problem-based algorithm based on a berth allocation problem solution defined by the chromosome. The results of numerical experiments show that the proposed heuristic is applicable to solve this difficult but essential terminal operation problem.

Keywords: Berth allocation; Crane scheduling; Terminal management;
Container transportation; Mathematical programming

1. Introduction

In Multi-User Container Terminals (MUTs) featuring a long quay, efficient berth allocation is a vital factor for successful terminal operations. In the existing literature it has been demonstrated that by employing flexible ship-berth-order allocation without taking into account the First-Come-First-Served rule, high productivity is obtained, where a ship's handling time depends on its berthing location.

Quay cranes are one of the key terminal equipment used for container movement. This in turn implies that inefficient quay crane employment could be the bottleneck for fast ship handling operations. For this reason, a terminal is obliged to be able to provide as many cranes as possible to serve a specific ship; however, when there is a fluctuation in ship calling frequency, this turns out to be a very expensive and certainly not a cost-effective investment, especially due to the high crane costs. Therefore, a competitive terminal needs to maximize the operating efficiency of a rather limited number of cranes.

As shown in Fig. 1, container quay cranes are mounted on the same track (or rail) and move from one berth to another on the track. This structural feature does not enable them to bypass each other and consequently they cannot move freely when requested at another berth. This study considers crane allocation (or scheduling), with respect to the decision to be made as to when and how many cranes are to be moved from a specific berth to another berth for fast overall ship handling in the system. From this point of view, the berth allocation and crane allocation are closely related.

Fig. 1

Envisage the following situation in which ships can start service only when a predetermined number of cranes have been allocated to them. There are three berths: 1, 2 and 3, located in that order. Berth 1 is serving a ship with the required number of cranes. Also this berth has some extra cranes which are currently not being used for ongoing

handling and remain idle. Berth 2, located between berths 1 and 3, is using a number of cranes in serving a ship with a long handling task. Berth 3 is idle with very few cranes available when a ship is arriving at the terminal. The arriving ship can be berthed at berth 3, but in most cases it does not start its handling if the available cranes are much fewer than the ones required. In such a case, if the surplus cranes at berth 1 were able to move to berth 3 through berth 2, the ship at berth 3 would start its service immediately; however this is not practical since the ship at berth 2 utilizes a number of cranes and it is almost impossible to suspend the ship's handling by stopping all the cranes engaged to the ship in order to let even one crane go through from berth 1 to berth 3. The other alternative for a faster turnaround would be making the incoming ship wait for berth 1 to become idle. This example signifies the close relationship between berth and crane allocation.

It is also noted that the trend towards increased container ship size, has led to a great diversity in the ship sizes that are calling at hubs. Consequently, the variation in the different number of quay cranes required for handling a ship will increase. Thus, the importance of crane allocation will become even more significant.

This paper proposes a model for simultaneous berth-crane allocation that minimizes the total service time (including wait and handling times) and develops a genetic algorithm-based heuristic.

The paper is organized as follows. The next section provides a literature review on the berth and crane allocation planning. The problem formulation is discussed in Section 3. Section 4 introduces a solution algorithm, followed by a number of computational analyses in Section 5. The final section concludes the paper.

2. Literature review on the berth and crane scheduling

As issues of efficient terminal operations have been constantly gaining importance, there have been a growing number of studies that deal with the berth allocation problem

(BAP) as well as the quay crane allocation problem (CAP) although very few of them have examined these issues simultaneously.

Lai and Shih (1992) proposed a heuristic algorithm for berth allocation, which was motivated by more efficient terminal (actually berth) usage in the HIT terminal of Hong Kong. Their problem considered a First-Come-First-Served (FCFS) allocation strategy, which is not the case in our problem. Brown et al. (1994, 1997) examined ship handling in naval ports. They identified the optimal set of ship-to-berth assignments that maximize the sum of benefits for ships while in port. Berth planning in naval ports has important differences from planning in commercial ports. In the former, a berth shift occurs when, for proper services, a newly arriving ship must be assigned to a berth where another ship is already being served. This treatment is unlikely in commercial ports. Berth shifting as well as other factors less relevant to commercial ports are taken into account in their paper, thus making their study inappropriate for commercial ports.

Imai et al. (1997) addressed a BAP for commercial ports. Most service queues are in general processed on an FCFS basis. They concluded that in order to achieve high port productivity, an optimal set of ship-to-berth assignments should have been 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 evaluation criteria, 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 assumed a static situation where ships to be served for a planning horizon had all arrived at a port before one plans the berth allocation. Thus, their study can apply 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 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, was considered. Due to the difficulty in finding an exact

solution, they developed a heuristic by using a subgradient method with 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 BAP for the multi-water depth configuration. They employed 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 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 BAP in 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. Imai et al. (2007) proposed the BAP with simultaneous berthing of multiple ships at the indented berth, which is potentially useful for fast turnaround of mega-containerships.

There is another type of the berth allocation problem, 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 berthing space unused. 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 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) 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. 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) studied 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 in terms of that factors assessed in the objective depend on the quay locations of ships. Kim and Moon (2003) addressed the same BAPC as one tackled by Park and Kim (2002), though the former employed the simulated annealing method while the latter applied the subgradient optimization method. Notice that all the above BAPC studies consider the ship handling time as independent from the berthing location. Imai et al. (2005b) addressed a BAPC, but with a major difference from the other BAPCs in that the handling time depends on the berthing location of ship. They developed a heuristic for that problem in cooperation with a heuristic for the dynamic BAP 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. 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. For the continuous location version, the solution quality was assessed by comparisons with solutions by the straightforward heuristic.

There are not so many existing studies on the CAP, especially ones that focus on quay crane scheduling, for both conventional and container berths. Quay cranes moving on

tracks at a quay are restricted in being relocated to another place along the berth as they cannot pass each other. Daganzo (1989) originally proposed a heuristic for efficient crane scheduling that optimally allocates a set of cranes to multiple ships in service with the minimization of the aggregate cost of delay in turnaround of the ships at the port. In a subsequent study by Peterkofsky and Daganzo (1990), an exact solution method was implemented by using the branch and bound technique. Kim and Park (2004) addressed a crane scheduling problem for a container ship which schedules cranes engaged in handling a specific single ship (not multiple ships under handling tasks) in order to minimize the ship's turnaround time. Therefore, their study was discussed in a different context from the two previous studies.

There are also a few studies, which deal with both berth and quay crane allocation to ships similar to this study. Park and Kim (2003) studied the BAP in a continuous location index with crane allocation with a similar objective to the one expressed by Park and Kim (2002), and Kim and Moon (2003). Park and Kim (2003) determined the optimal start times of ship service and associated mooring locations and at the same time they determined 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. Their solution procedure consists of two phases: one for berth allocation and the other for crane allocation. In a sense, they solved the berth-crane allocation problem simply by solving the berth allocation and crane allocation independently. Lee et al. (2005) also treated the berth and crane allocation problem. They employed genetic algorithm by decomposing the entire problem into BAP and CAP; however no detailed descriptions on the solution procedure were available in their paper.

3. Problem formulation

As described previously, the problem of berth and quay crane allocation to a ship (B&CAP) consists of two sub-problems: one is the BAP and the other the CAP. B&CAP assumes the discrete berth unit on which the allocation of ships and quay cranes to berthing places is based.

For berth allocation, we make the same assumptions as those for the DBAP in Imai et al. (2001). That is:

- (a) Each berth can serve one ship at a time
- (b) There are no physical or technical restrictions such as ship draft and water depth
- (c) Ship handling time is dependent on the berth where it is assigned
- (d) Ship is served after its arrival
- (e) Ship handling tasks must be finished without interruption once they get started.

For crane scheduling, the following assumptions are made:

- (f) Ship handling requires a specific number of cranes and it does not begin till that number of cranes are available
- (g) Cranes cannot move from one berth to another via other berths if the other berths are engaged in ship handling.
- (h) Cranes get through an idle berth having some cranes present by the pushing-in and pulling-out procedure

Regarding assumption (c), it is noted that the B&CAP does not imply that the handling time of a ship at a specific berth is constant regardless of the number of cranes engaged for its handling. As will be seen in the formulation of B&CAP, although we do not explicitly take into account the number of cranes as a factor influencing the handling time of a ship, we indirectly assume that the handling speed is dependent on the number of cranes programmed by planners in serving a particular ship at a specific berth, i.e., assumption (c), which is used as a parameter in constraints of the formulation.

Notice that assumption (f) may not be pragmatic, because in practice the ship handling can start with some available cranes, even if fewer than needed, in order to

minimize the delay in turnaround at port. This assumption is made to facilitate the solution process of the problem. Letting ships start their handling with fewer cranes and then add some more cranes in the course of handling, asks the formulation to define the ship's handling time depending not only on the berth assigned to the ship and but also on the number of cranes engaged to the ship. In addition it asks to consider a handling speed varying with time as cranes are added or withdrawn from the handling of the ship under consideration. This type of crane usage, although it may be realistic indeed, transforms the formulation in a complex form (maybe nonlinear) and considerably complicates the solution procedure. It is considered that the problem solution of the B&CAP in this study provides the upper bound of the solution to the B&CAP with flexible crane usage that is discussed above.

With respect to assumption (g), in container terminals quay cranes are basically shared by multiple berths and ships; therefore it is common practice to move some cranes from one berth (ship) to another, while handling tasks of the ships in concern are underway. This is also related to assumption (f). In fact moving cranes between two adjacent berths is not physically difficult even when those berths are serving ships. Though the authors do not know the details of quay crane transfer in container terminals, it is easily envisaged that crane movements via some berths (under handling task) interrupt the smooth handling of ships at those intermediate berths. Consequently, in a large container terminal with a lot of berths, e.g., 5 and more berths, frequent crane movements are substantially limited. If an intermediate berth serves a large ship by using many cranes (say, 7), all cranes must concurrently stop working in order to let even one crane move from one berth to another. This results in disruption of service and subsequently causes a long delay in handling the large ship. In light of this discussion, assumptions (e), (f) and (g) may not necessarily be applied in practice in all cases, but are considered acceptable for a huge container terminal. Note that in this study a crane movement is allowed via intermediate berth in idle state.

We employ genetic algorithm (GA) for a heuristic that solves the B&CAP. Park and Kim (2003) dealt with B&CAP as two independent problems: BAP and CAP. In contrast,

this study determines these two individual decisions at the same time, while the two decision making procedures are constructed one by one in an iteration of the entire GA solution procedure. In the BAP phase, the ship-to-berth allocation is determined. Given this allocation, cranes are scheduled by assigning them to ship handling tasks in sequence across all berths. In the problem setting, every ship handling task at a berth does not start if fewer cranes than the required number of cranes for that task are available at that berth; therefore the task will be postponed till it gets the sufficient number of cranes. This implies that throughout a specific computational iteration, the BAP solution remains unchanged.

From the above discussion, the formulation of B&CAP can be partitioned into the BAP and CAP. Before composing the B&CAP formulation, we first formulate the BAP.

3.1. BAP formulation

The BAP, which is discussed in this paper, is equivalent to the DBAP defined in Imai et al. (2001). The formulation in their study is an extended form of the three-dimensional assignment problem with two types of decision variables: one implying the ship-berth-order assignment and the other the time gap existing between two adjacent ships served at a berth. Their formulation does not explicitly need to specify the start and completion times of ship handling. This study, however, needs these times as decision variables. As the problem size increases with more variables, the BAP in this study is formulated without the time gap variables used in Imai et al. (2001).

The BAP formulation may be as follows:

$$[\text{BAP}] \text{ Minimize } \sum_{j \in V} \left\{ \sum_{i \in B} \sum_{k \in U} f_{ijk} - A_j \right\} \quad (1)$$

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

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

$$\sum_{i \in B} \sum_{k \in U} b_{ijk} \geq A_j \quad \forall j \in V, \quad (4)$$

$$\sum_{j \in V} b_{ijk} \geq \sum_{j \in V} f_{ijk-1} \quad \forall i \in B, k \in U, \quad (5)$$

$$b_{ijk} \leq Mx_{ijk} \quad \forall i \in B, j \in V, k \in U, \quad (6)$$

$$b_{ijk} + C_{ij}x_{ijk} = f_{ijk} \quad i \in B, j \in V, k \in U, \quad (7)$$

$$\sum_{j \in V} f_{ij0} = S_i \quad i \in B, \quad (8)$$

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

$$b_{ijk}, f_{ijk} \geq 0 \quad \forall i \in B, j \in V, k \in U, \quad (10)$$

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

M : large constant

S_i : start time of the usage of berth i in the planning horizon

A_j : arrival time of ship j

C_{ij} : handling time of ship j at berth i

b_{ijk} : start time of serving ship j at berth i as the k th ship

f_{ijk} : completion time of serving ship j at berth i as the k th ship

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

b_{ijk} s, f_{ijk} s and x_{ijk} s are decision variables.

The objective function (1) is the minimization of the total service time. Constraints (2) ensure that each ship is served in one of the berths. Constraints (3) enforce that every berth serves up to one ship at any time. Constraint set (4) assures that a ship starts being handled after its arrival at port. Constraint set (5) guarantees that the handling of a ship starts after the completion of handling of its immediate predecessor at the same berth. Inequalities (6) set $b_{ijk} = 0$ if $x_{ijk} = 0$ and $b_{ijk} \geq 0$ otherwise. Equalities (7) define the relationship of b_{ijk} and f_{ijk} . Equalities (8) imply that the ship prior to the first ship served at berth i finishes its handling at S_i . Along with constraints (5) this actually guarantees

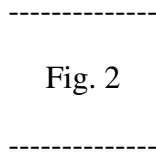
that the first ship at berth i does not start being handled before berth i is available in the planning horizon.

3.2. B&CAP formulation

We add constraints of the CAP to the BAP in order to make up the B&CAP formulation. We first describe the outline of the CAP constraints as follows:

Quay cranes move from a berth to another in a quay. However, as they are mounted on rails and run on the same track, they cannot pass each other. Thus, if we would like to move a quay crane from berth 1 to berth 3 via berth 2 where some cranes are engaged in handling a ship, first we must suspend ship handling at berth 2, and then we may move a crane from berth 1 to berth 2 while we move another crane, which has already been working at berth 2, to berth 3. Nevertheless, such a crane transfer is not preferable because of the suspended ship handling at berth 2. This leads to the assumption that a crane transfer cannot take place from an origin berth to a destination berth via berths in between under loading and discharging.

Fig. 2 illustrates crane movements across berths in a chronological time axis. Cranes are transferred from one berth to another when the ship handling at the first berth is finished. In Fig. 2, ship 1 at berth 1 finishes its handling while ship 4 is still under handling at berth 2. Some cranes are to be moved from berths 1 to 2 for usage in handling ship 5. Physically cranes to be transferred can be moved to berth 2, but berth 2 is busy; therefore they stay idle at berth 2 and are practically used after ship 4 has finished its handling. Thus, the arrow in Fig. 2 for moving cranes from berths 1 to 2 starts at the end of ship 1 and arrives at the end of ship 4 (or the start of ship 5).



For the B&CAP formulation, we treat the movement of cranes in more detail. Fig. 3 shows detailed crane movements among ships 1, 2, 4, 5, 7 and 8. For representation of crane

movements, we define the following variables:

- e_{ik} : number of cranes at berth i immediately after serving the k th ship, including cranes coming from neighboring berths
- $m_{i'ik'}$: number of cranes transferred from berth i' after serving the k' th ship to berth i (the next to berth i') after serving the k th ship
- z_{ik} : number of cranes allocated to berth i for the k th ship, including cranes in idle state

e_{ik} can be interpreted as the hypothetical node existing at berth i as shown in Fig. 3, where it is surrounded by related ships 1, 2, 4, 5, 7 and 8. After finishing ship 4's handling as the k th ship at berth i ($=2$), cranes (i.e., z_{ik} s) used at berth 2 stay there and those employed at adjacent berths (1 and 3) come up to berth 2, as flow $m_{i'ik'}$ s, to form the inventory of cranes, e_{ik} . From e_{ik} , cranes are delivered to those ships to be served next at the neighbor berths (and for berth 2 also).

Fig. 3

The crane allocation problem may be formulated as follows:

$$[\text{B\&CAP}] \text{ Minimize } \sum_{j \in V} \left\{ \sum_{i \in B} \sum_{k \in U} f_{ijk} - A_j \right\} \quad (1)$$

subject to (2) - (10)

$$\sum_{j \in V} F_j x_{ijk} \leq z_{ik} \quad \forall i \in B, k \in U, \quad (11)$$

$$e_{ik} = z_{ik} + \sum_{i' (=i-1, i+1) \in B} \sum_{k' \in U} m_{i'ik'} \quad \forall i (\neq 1, I) \in B, k \in U, \quad (12)$$

$$e_{ik} = z_{ik+1} + \sum_{i' (=i-1, i+1) \in B} \sum_{k' \in U} m_{i'ik'} \quad \forall i (\neq 1, I) \in B, k \in U, \quad (13)$$

$$e_{ik} = z_{ik} + \sum_{k' \in U} m_{i+1, i, k'k} \quad \forall i (=1) \in B, k \in U, \quad (14)$$

$$e_{ik} = z_{ik+1} + \sum_{k' \in U} m_{i,i+1,kk'} \quad \forall i(=1) \in B, k \in U, \quad (15)$$

$$e_{ik} = z_{ik} + \sum_{k' \in U} m_{i-1,i,k'k} \quad \forall i(=I) \in B, k \in U, \quad (16)$$

$$e_{ik} = z_{ik+1} + \sum_{k' \in U} m_{i,i-1,kk'} \quad \forall i(=I) \in B, k \in U, \quad (17)$$

$$\sum_{j'(\neq j) \in V} f_{i'j'k'} - (\delta_{ii'kk'} - 1)M \geq \sum_{j \in V} f_{ijk} \quad \forall i, i' \in B, k, k' \in U, \quad (18)$$

$$\sum_{j \in V} f_{ijk} - (\delta_{ii'kk'} - 1)M \geq \sum_{j'(\neq j) \in V} b_{i'j'k'} \quad \forall i, i' \in B, k, k' \in U, \quad (19)$$

$$m_{ii'kk'} - (\delta_{ii'kk'} - 1)M \geq 0 \quad \forall i, i' \in B, k, k' \in U, \quad (20)$$

$$\sum_{j \in V} f_{ijk} - (\phi_{ii'kk'} - 1)M \geq \sum_{j'(\neq j) \in V} f_{i'j'k'} \quad \forall i, i' \in B, k, k' \in U, \quad (21)$$

$$\sum_{j'(\neq j) \in V} b_{i'j'j'(k'+1)} - (\phi_{ii'kk'} - 1)M \geq \sum_{j \in V} f_{ijk} \quad \forall i, i' \in B, k, k' \in U, \quad (22)$$

$$m_{ii'kk'} - (\phi_{ii'kk'} - 1)M \geq 0 \quad \forall i, i' \in B, k, k' \in U, \quad (23)$$

$$m_{ii'kk'} - (1 - \gamma_{ii'kk'})M \leq 0 \quad \forall i, i' \in B, k, k' \in U, \quad (24)$$

$$\delta_{ii'kk'} + \phi_{ii'kk'} + \gamma_{ii'kk'} = 1 \quad \forall i, i' \in B, k, k' \in U, \quad (25)$$

$$\sum_{i' \in B} \sum_{k' \in U} m_{ii'kk'} \leq M \sum_{j \in V} x_{ijk} \quad \forall i \in B, k \in U, \quad (26)$$

$$z_{i1} = Q_i \quad \forall i \in B, \quad (27)$$

$$TQ = \sum_{i \in B} Q_i \quad (28)$$

$$m_{ii'kk'} \leq z_{ik} \quad \forall i, i' \in B, k, k' \in U, \quad (29)$$

$$m_{i'ik'k} \leq e_{ik} \quad \forall i, i' \in B, k, k' \in U, \quad (30)$$

$$\sum_{i \in B} z_{ik} = \sum_{i \in B} Q_i \quad \forall k \in U, \quad (31)$$

$$\delta_{ii'kk'} \in \{0, 1\} \quad \forall i, i' \in B, k, k' \in U, \quad (32)$$

$$\phi_{ii'kk'} \in \{0, 1\} \quad \forall i, i' \in B, k, k' \in U, \quad (33)$$

$$\gamma_{ii'kk'} \in \{0, 1\} \quad \forall i, i' \in B, k, k' \in U, \quad (34)$$

$$z_{ik} \geq 0 \quad \forall i \in B, k \in U, \quad (35)$$

$$e_{ik} \geq 0 \quad \forall i \in B, k \in U, \quad (36)$$

$$m_{ii'kk'} \geq 0 \quad \forall i, i' \in B, k, k' \in U, \quad (37)$$

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

F_j : number of cranes needed by ship j

Q_i : number of cranes initially assigned to berth i in the planning horizon

TQ : total number of cranes deployed in the system

b_{ijk} s, e_{ik} s, f_{ijk} s, x_{ijk} s, z_{ik} s and $m_{ii'kk'}$ s are decision variables, while $\delta_{ii'kk'}$, $\phi_{ii'kk'}$ and $\gamma_{ii'kk'}$ are auxiliary variables.

The objective function (1) is the minimization of the total service time. Constraints (11) ensure that there must be an adequate number of cranes to be used at a berth for handling a ship. Constraint sets (12) – (17) are flow conservation criteria for movements of cranes (i.e., cranes must leave a hypothetical node to which they moved) related to the completion of the service for the k th ship at berth i . These constraints ensure crane transfers along arrows around interchange node e_{ik} in Fig. 3. In more detail, (12) enforces that a specific berth i receives cranes from the berth and neighboring ones before the time in concern. Conversely, (13) ensures that berth i delivers cranes to itself and the neighboring ones after the time in concern. Eqs. (14) and (15) serve as same as (12) and (13) but for berth 1. Similarly (16) and (17) are for berth I .

Constraints (18) – (25) are a set of flow conservation formulations in terms of crane transfer between two specific adjacent berths. Between two ships, one served as the k th ship at berth i and the other as the k' th at the next berth i' , the crane transfer is possible in two cases as shown in Fig. 4: case (i) is the case (i.e., $\delta_{ii'kk'} = 1$) when the ship at berth i completes its handling before the ship at berth i' , and the latter ship of berth i' starts the handling no later than the former ship of berth i completes; and case (ii) is the case (i.e., $\phi_{ii'kk'} = 1$) when the ship at berth i completes its handling after the one at berth i' and before the next ship at berth i' . Constraints (18) – (20) are those for case (i) while

constraints (21) – (23) for case (ii). Constraints (24) ensure no crane transfer between two ships in any case except for cases (i) and (ii). Equality set (25) guarantees that there are only two cases for crane transfer flow ≥ 0 . Constraint set (26) enforces that if a specific ship j is to be served at another berth and not at berth i , then no flow at berth i is generated with respect to ship j . Constraints (27) associate the initial number of cranes at each berth with the cranes assigned to the first ship at that berth. Equality (28) ensures that all the cranes available in the system are deployed to all the berths. Constraints (29) and (30) define, for a specific berth, the relationship between crane moves, a hypothetical node of crane move and the number of cranes located at that berth immediately before the node. Equalities (31) guarantee that at each phase of the planning horizon, the total number of cranes across the berths remains the same as the total number of cranes in the system.

Fig. 4

4. Solution procedure

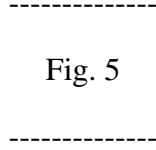
Due to the complexity of formulation [B&CAP], it is unsuited to be solved by commercial mathematical programming solvers such as LINGO. For this reason, we develop a GA-based heuristic to find an approximate solution for the B&CAP. As stated before, the B&CAP has a mixed feature of BAP and CAP. From this point of view, a chromosome should represent berth-ship-order-crane assignment; however, it results in a very complicated form of chromosome. For simplicity in the solution procedure, we develop the following GA procedure:

At each iteration (or generation), the entire procedure performs two computations: one is the GA procedure (i.e., crossover and mutation) for ship-berth-order allocation without consideration of crane scheduling and the other is the heuristic for scheduling

cranes given a ship-berth-order allocation alternative defined by a chromosome.

4.1. Genetic Algorithm

The chromosome we use is shown in Fig. 5. In this chromosome, ships 2, 8, 5 and 9 are served at berth 1 in this order, while ships 4, 7, 3, 1 and 6 are handled at berth 2, where there are only two berths in the terminal.



As GA is widely applied for a number of combinatorial optimization problems, we do not discuss it in detail; however, its major characteristics are the following:

(a) Fitness

The B&CAP is the minimization problem; thus its objective function value is transformed to the fitness value, which is evaluated for reproduction (see (b)), by the sigmoid function as below:

$$f(x) = \frac{1}{1 + \exp(y(x)/10000)} \quad (38)$$

where $f(x)$ has a value ranging from 0 to 0.5.

Note that the objective function $y(x)$ used for the fitness value is the one of the B&CAP. As mentioned, a chromosome simply defines only a ship-berth-order assignment without any information on the crane schedule. Consequently, before the reproduction process, crane scheduling is performed given a chromosome, so that the objective function value is produced to be utilized for the fitness value. The crane scheduling will be discussed later in detail.

(b) GA operators

Reproduction is a process where individual chromosomes are copied according to

their scaled fitness function value. We provide the tournament strategy. For the crossover procedure, the so-called *2-point crossover* is employed. See Nishimura et al. (2001) for detail.

4.2. Crane scheduling

In the crane scheduling phase, we perform the following two tasks:

- (a) Feasibility check of ship service schedule regarding the number of cranes available
- (b) Rescheduling the crane task

The chromosome determines the ship-berth-order assignment. If there are a lot of cranes, ships are served as soon as they arrive at the port and assigned berths become idle, according to the service order of ships at the respective berths which is scheduled by a chromosome generated (such a ship service schedule is hereafter referred to as *initial schedule*). If there are not enough cranes, some ships may delay their services even with a well organized crane schedule. The feasibility check of (a) examines if the given number of cranes to the system as an input parameter is large enough for ships to be served as initially scheduled. Unless it is identified that the service can be executed as initially scheduled, the delay of service is minimized by optimally scheduling crane tasks. This process is (b).

In the following section, both processes are described in detail.

4.2.1. Feasibility check for the ship service schedule

4.2.1.1 Outline of the checking procedure

First, we determine the start and completion times for ship service (without consideration of cranes). This computation is easily made, since they are calculated one by one in their order of service for each berth such that the start time of a specific ship is the earliest time between the ship arrival time and the completion time of the immediately previous ship served at the berth.

Next, given the ship service schedule determined as above, we construct a crane movement network as shown in Fig. 6, in which dotted boxes represent ship services in a

chronological scale. This network is comprised of two types of nodes and two types of arcs. One type of node is the special node (one is source and the other is sink) and the other type is the node between two adjacent ships at a berth, where the former is the white node and the latter is the black node, respectively in Fig. 6. One type of arc is the arc representing a ship service, while the other is the crane move between berths, where the former arc is solid and the latter is dotted in Fig. 6. Crane flows are distributed from the source node, which associates all the cranes available in the system (i.e., TQ), to all the berths. At last, cranes in all the berths are destined for the sink node.

Fig. 6

Solid arcs are all directed arcs, while dotted arcs are directed or bidirected. A bidirected dotted arc is the representation of two directed dotted arcs. A bidirected arc lies between two berths in the case that the time gap between two ships at a berth overlap (partly or fully) with the other time gap at the adjacent berth. In Fig. 6, a bidirected arc exists between the ships 4-5 node at berth 2 and the ships 7-8 node at berth 3, since cranes can move towards either direction. On the other hand, a single-directed arc exists when those time gaps do not overlap each other at all, like the arc between nodes of ships 1-2 at berth 1 and ships 4-5 at berth 2. This is interpreted as that cranes cannot move from berths 2 to 1 after ship 4, in order to be used for ship 2; whereas cranes are able to move, for ship 5, in the opposite way after ship 1. It is noteworthy that cranes can move from one berth to a remote berth via some other berths along dotted arcs; for instance from the node of ships 2-3 at berth 1 to that of ships 11-12 at berth 4, via berths 2 and 3. Therefore, the nodes serve as crane switching points.

Given the crane movement network $G = (V, A)$, where V is a node set representing crane switching points as described before and A is an arc set, we solve the following flow problem:

$$[\text{CS}] \text{ Minimize } q \quad (39)$$

$$\text{subject to} \quad \sum_{(i,j) \in A} y_{ij} = q \quad i = s, \quad (40)$$

$$\sum_{(i,j) \in A} y_{ij} - \sum_{(i,j) \in A} y_{ji} = 0 \quad \forall i \in V, \quad (41)$$

$$\sum_{(i,j) \in A} y_{ji} = q \quad i = t, \quad (42)$$

$$y_{ij} \geq F_{ij} \quad \forall (i,j) \in A, \quad (43)$$

where,

s : source node

t : sink node

F_{ij} : minimum required flow of arc (i, j)

y_{ij} : flow on arc (i, j)

Flow y_{ij} is the number of cranes used for a ship service or for a crane transfer across adjacent berths. The objective function value q of the optimal solution to [CS] is the minimum number of cranes that serve ships as scheduled initially. Equalities (40) and (42) are the total flow originating from the source node and heading to the sink node, respectively. Equalities (41) are flow conservation criteria at intermediate nodes. Inequalities (43) guarantee that the number of cranes (or a flow) through arc (i, j) , which corresponds to a ship handling task, has to be greater than or equal to the required number of cranes for the arc. For the arc representing ship service, F_{ij} corresponds to F_j in the formulation [B&CAP], which is the required number of cranes; while $F_{ij} = 0$ for the arc of crane transfer. q is a criterion to judge if the number of cranes given to the system as an input parameter TQ in [B&CAP] is large enough to serve the ships as scheduled; if $q \leq TQ$, the service schedule remains unchanged, otherwise it must be changed with the resulting delay in handling ships. Notice that no efficient exact algorithm is found for [CS].

We next transform [CS] to solve it. Define $\psi_{ij} = -y_{ij}$, $\Theta_{ij} = -F_{ij}$ and $\rho = -q$. Putting these new variables and parameters into [CS], we have the following problem:

[CS'] Minimize $-\rho$

$$\begin{aligned}
\text{subject to} \quad & - \sum_{(i,j) \in A} \psi_{ij} = -\rho & i = s \\
& - \sum_{(i,j) \in A} \psi_{ij} + \sum_{(i,j) \in A} \psi_{ji} = 0 & \forall i \in V \\
& - \sum_{(i,j) \in A} \psi_{ji} = -\rho & i = t \\
& -\psi_{ij} \geq -\Theta_{ij} & \forall (i,j) \in A
\end{aligned}$$

Furthermore, [CS'] is transformed as below:

[CS''] Maximize ρ (44)

$$\text{subject to} \quad \sum_{(i,j) \in A} \psi_{ij} = \rho \quad i = s, \quad (45)$$

$$\sum_{(i,j) \in A} \psi_{ij} - \sum_{(i,j) \in A} \psi_{ji} = 0 \quad \forall i \in V, \quad (46)$$

$$\sum_{(i,j) \in A} \psi_{ji} = \rho \quad i = t, \quad (47)$$

$$\psi_{ij} \leq \Theta_{ij} \quad \forall (i,j) \in A, \quad (48)$$

The resulting formulation [CS''] is the well-known maximum flow problem, except for constraint (48). In the maximum flow problem, the capacity constraints have a lower bound of zero, i.e., $0 \leq \psi_{ij} \leq \Theta_{ij}$. In contrast, [CS''] enables ψ_{ij} to be negative due to $\psi_{ij} = -y_{ij} \leq 0$ and $\Theta_{ij} = -F_{ij} \leq 0$. For this reason, we cannot employ efficient exact solution methods for the maximum flow problem such as the labeling algorithm.

Instead of solving [CS] directly, we treat this problem in the following way. First, with a sufficiently large number of cranes in the system, we determine the crane flow in the network so as to meet constraints (43), by the modified labeling algorithm (described in Section 4.2.1.2). Then, letting σ be the given number of cranes, we have $q = \sigma - \delta$. Minimizing q corresponds to maximizing δ , where δ is the total flow from t to s traversing reversely along arcs in the network. This backward flow decreases the forward

flow from s to t by the concept of the residual network (Ahuja et al., 1993), on which the labeling algorithm is based. In the network of concern, every arc $(i, j) \in A$ associates $(j, i) \in A$ with it. The residual capacity r_{ij} is defined as $r_{ij} = u_{ij} - w_{ij} + w_{ji}$, where u_{ij} and w_{ij} are the capacity of arc (i, j) and the flow on it, respectively. By the definition of the residual capacity, the flow on backward arc (j, i) cancels to increase the flow from node i to node j . In order to find the backward flow from t to s , we solve the maximum flow problem in the network only with backward arcs, as follows:

$$[\text{BMAX}] \quad \text{Maximize } \delta \quad (49)$$

$$\text{subject to} \quad \sum_{(j,i) \in A} w_{ji} = \rho \quad i = t, \quad (50)$$

$$\sum_{(j,i) \in A} w_{ji} - \sum_{(j,i) \in A} w_{ij} = 0 \quad \forall i \in V, \quad (51)$$

$$\sum_{(j,i) \in A} w_{ij} = \rho \quad i = s, \quad (52)$$

$$0 \leq w_{ji} \leq u_{ji} \quad \forall (j, i) \in A, \quad (53)$$

where $u_{ji} = y'_{ij} - F_{ij}$ and y'_{ij} is the initial crane flow on arc (i, j) as mentioned above.

The definition of u_{ji} is justified by the following lemma:

Lemma 1. *The definition of u_{ji} guarantees that $y_{ij} \geq F_{ij}$.*

Proof. As $u_{ji} = y'_{ij} - F_{ij}$, constraints (53) are reduced to:

$$w_{ji} \leq y'_{ij} - F_{ij}$$

and further

$$F_{ij} \leq y'_{ij} - w_{ji} \quad (54)$$

The crane flow on arc (i, j) is:

$$y_{ij} = y'_{ij} - w_{ji} \quad (55)$$

By substituting (55) for (54)

$$F_{ij} \leq y_{ij}. \quad \square$$

Notice that [BMAX] also assumes the property of the residual capacity.

4.2.1.2 Flow construction procedure

In order to construct the flow from s to t that satisfies the required number of cranes by each ship, F_{ij} , we modify the labeling algorithm (see Appendix A) to develop the following procedure where r_{ij} is defined as $r_{ij} = F_{ij} - y_{ij} + y_{ji}$ in this algorithm.

algorithm *modified-labeling*;

begin

while all $F_{ij} \leq y_{ij}$ are satisfied **do**

begin

unlabel all nodes;

set $\text{pred}(j) := 0$ for $j \in N$;

label node s and set $\text{LIST} := \{s\}$;

while $\text{LIST} \neq 0$ or t is unlabeled **do**

begin

remove a node i from LIST;

for each arc (i, j) in residual network emanating from node i **do**

if $r_{ij} > 0$ and node j is unlabeled **then** set $\text{pred}(j) := i$, label node j ,

and add j to LIST

end;

if t is labeled **then** *augment*

end;

end;

4.2.2. Rescheduling the crane job

After the feasibility check regarding the given number of cranes as described previously, ship services may be rescheduled differently from the initial schedule which was

made directly from a GA chromosome. That is, if adequate quantities of cranes are not available at the initially scheduled start time of a ship service at a berth, then the service is postponed till the quantities are satisfied by the transfer of cranes from other berths.

The procedure that determines the service start time for ships, checks the ships in ascending order of ship service defined by a chromosome. It is noticed that the defined service orders at a berth are not arranged in ascending order of ship arrival time. Thus, even if a ship to be processed at a berth according to the service order arrives earlier than already processed ships at the same berth, the ship is to be scheduled to be served later than the processed ships.

Basically the ship checking procedure examines the ships one by one in ascending order of service by the chromosome; however, the resulting total service time is not necessarily better than by the other process rules. For this reason, we process the ships in the following way:

We examine two ships at a time. Note that the two ships are those scheduled to be served at different berths. Those ships start service as initially scheduled if an adequate number of cranes can be moved over from the previously-served ships at their respective berths. Otherwise, crane transfers should be performed. In this case, we examine which ship should start service with higher priority. Assuming two ships 1 and 2, first we make the crane transfer for ship 1 first and then for ship 2. Then we perform the same process for ship 2 first and for ship 1 second. The crane transfer procedure results in some delay in services of ships 1 and 2 (and those ships that follow them as well). For these two process alternatives, we compute the total estimated delay of service of ships that follow the two ships at both berths. We schedule the two ships with a better alternative in terms of the estimated delay. Then, we examine the next two ships that are scheduled to be served at different berths, and so on till all the ships are processed.

An effort should be made to transfer some cranes from neighbor berths in the case that with presently available cranes at respective berths, either one or both ships are not served as initially scheduled. If the number of cranes is still not satisfactory for the service

of the ship after the crane transfer from the neighbor berths, additional cranes are moved from neighboring berths of the neighbors. The number of transferable cranes from a berth is the number of cranes available at the present time minus the number of cranes used by the next ship at that berth, which has been already fixed its service schedule. For instance, in Fig. 7, the number of transferable cranes from berth 3 to berth 4 is the number of available cranes after ship 4 minus the number of cranes used by ship 5.

Fig. 7

As shown in Fig. 7, this process is repeated till it is determined whether or not sufficient cranes are provided for the service as long as the service begins at the scheduled time. If there is a crane shortage for the service, the service is postponed till the satisfactory number of cranes is obtained. Note that in the crane transfer process, there may be ships which cannot provide cranes to other berths after their service due to the completion time of the ships such as ship 1 at berth 1 in Fig. 7. Hereafter, those ships that can provide cranes to the ship at the berth of concern after their service, are referred to as *source ships*. In Fig. 7, ships 2 and 4 are source ships for ship 7. Also it should be noted that in examining the possibility of crane transfer we only consider those ships whose service schedule has already been determined. There may be some ships that have not yet determined their schedule after ships 1, 3 and 5 in the system; but those unscheduled ships are not taken into consideration.

As mentioned before, the crane transfer procedure may cause some delay in service for the ship under consideration. In Fig. 7, if there are not enough cranes available to ship 7 after the crane transfer, it is postponed till the time when another crane transfer is possible, i.e., the completion time of ship 5.

The estimation of the delay in ship service after the two presently examined ships, is made as follows: If the start time of the ship immediately following the ship under examination is no earlier than the delayed (or new) completion time of the latter ship, there

is no change in the start time of the former ship. Otherwise, the start time of the former is adjusted so that its service begins immediately after the new completion time of the latter without consideration of crane. This rescheduling is a temporary one in order to obtain the estimated delay in service. Service to those ships that follow the ship temporarily rescheduled are put off in the same manner.

The entire procedure is as follows, where each berth i has a ship set $SHIP_i$ that contains ships to be served at the berth in order of service defined by a chromosome, and REQ and CRANE are the number of cranes required by the ship and the number of available cranes at the berth of the ship.

algorithm *rescheduling*;

begin

while all ships are not processed **do**

begin

pick up the next two ships with the earliest start time from $SHIP_i$ s for all the berths where both ships must be chosen from different $SHIP_i$ s;

delete the two ships from respective $SHIP_i$ s;

if there are not enough number of cranes at either or both of their berths **then do**

Let SHIP1=ship with the earliest start time;

Let SHIP2=ship with the second earliest start time;

Postponing;

compute DELAY1 as the total estimated delay of the ships after both ships;

Let SHIP2=ship with the earliest start time;

Let SHIP1=ship with the second earliest start time;

postponing;

compute DELAY2 as the total estimated delay of the ships after both ships;

choose the alternative with the minimum among DELAY1 and DELAY2 as the new temporary schedule;

```

    treat the temporary start and completion times of the two ships as determined ones;

end;

end;

procedure postponing;

begin

    while SHIP1 has not enough cranes do

        begin

            let SHIP=SHIP1 and BERTH=berth # of SHIP1;

            let F/B=1 and transfer;

            if ANS = 0 then let F/B = 2 and transfer

            if ANS = 0 then postpone of the service of SHIP1

        end;

        while SHIP2 has not enough cranes do

            begin

                let SHIP=SHIP2 and BERTH=berth # of SHIP2;

                let F/B=1 and transfer;

                if ANS = 0 then let F/B = 2 and transfer

                if ANS = 0 then postpone of the service of SHIP2

            end;

        end;

    end;

procedure transfer;

begin

    if F/B=1 then do;

        while CRANE  $\leq$  REQ do

            begin

                for each berth from BERTH to berth 1 do

```

```

        let SURPLUS=Max {the # of cranes after the service of the source ship –
        the required # of cranes for the next ship to the source ship at the berth, 0};
        let CRANE:=CRANE+SURPLUS;
    if CRANE≤REQ then ANS=0 else ANS=1;
if F/B=2 then do
    while CRANE≤REQ do
        begin
            for each berth from berth 1 to the last berth do
                let SURPLUS=Max {the # of cranes after the service of the source ship –
                the required # of cranes for the next ship to the source ship at the berth, 0};
                let CRANE:=CRANE+SURPLUS;
                if CRANE≤REQ then ANS=0 else ANS=1;
        end;
end;

```

5. Computational experiments

5.1. Settings

The solution procedure is coded in “C” language on a Fujitsu SPARC-64V workstation. By preliminary experiments, we identified parameters for GAs as population size=30, mutation rate=0.8 and the number of generations=2000. We consider two sizes of terminals: one with 4 berths and another with 5 berths. The ship handling time follows the 3-Erlangian distribution with two average times of 6 and 8 h and we produce two data sets with two different seeds of random number for generating the ship handling time for each average time. The number of cranes to be engaged to ship j , F_j , is defined randomly by using a uniform distribution. The planning horizon is one week and ships arrive at the terminal with an exponential distribution having four average intervals of 2, 3, 4 and 5 h. The numbers of calling ships generated during the one week planning horizon with arrival

intervals of 2, 3, 4 and 5 h are 88, 64, 45 and 34, respectively. In total, we generate 32 problem cases. For each problem case, the total number of quay cranes in the system, TQ , ranges from 8 to 18 with an interval of 2.

5.2. Results

Figs. 8 and 9 show the total service time of all ships involved in the system. Although different seeds result in different curves of the total service time in the figures, the overall trends are similar. As a general trend, the service time decreases with an increasing TQ because with more cranes ships are able to start service as soon as the assigned berth becomes idle. On the contrary, it is apparent that longer service times are associated with fewer cranes, the reason being that the ships are kept waiting for service at the assigned berth till the necessary number of cranes becomes available. Though generally there is a declining trend of the service time with TQ , a rising trend can be partly observed. This is caused by the limit of the heuristic's performance. Roughly speaking, the service time has a monotonous decreasing trend with a large arrival interval of ship. The reason envisaged is that the problem size in terms of the number of calling ships within the predetermined planning horizon increases with more frequent calling and subsequently poorly optimized solutions are likely identified with large problems. If the proposed algorithm was able to identify an exact solution, the service time would have to monotonously decrease as TQ increases.

Figs. 8-9

Service time is increasing with increased handling time, since ships are likely to be forced to wait for an idle berth. This, in turn, implies that the service time is decreasing with shorter handling time, which results in more idle time of berths and more crane availability. As a general trend, service time is decreasing with many more berths; however, it is not always the case with the arrival interval of 2 h. The reason is the following: the total service

time definitely decreases with an increasing number of berths in the BAP, as discussed in existing studies, using different number of berths in the terminal since an infinite number of cranes is implicitly assumed to be available while no quay cranes are taken into consideration explicitly in the BAP. This is the situation where as soon as ships are moored to the assigned berths, they start being handled immediately without any wait. On the other hand, in the B&CAP discussed here, ships cannot start their handling if the necessary number of cranes is not available at the assigned berth. In other words, ships can be moored sooner with more berths being assumed; however they cannot actually start their handling without the required number of cranes even after being moored at the dedicated berths. In short, the total service time can be shortened only if there are both enough berths and cranes. From this point of view, it seems logical that the total service time does not decrease as the number of berths increases in the case of fewer cranes being employed. Typically, there are the cases with $TQ=8$ and 10 that the total service time is larger for the 5-berth terminal than for the 4-berth terminal, since crane transfer across berths is not efficiently scheduled in the 5-berth terminal.

Next, by using the results shown in Figs. 8 and 9, we look into how the ship arrival interval affects the total service time. Figs. 10 and 11 show the average of service time per ship by the number of quay cranes. Note that the number of ships in the problem vary with a different ship arrival time: 88, 64, 45 and 34 ships by an interval of 2, 3, 4 and 5 h. Thus, the average of service time is obtained by dividing the total service time by the corresponding number of ships. These graphs show that when the terminal is congested with a lot of calling ships in a specific time duration, the average service time is long since the quay cranes are spread over many ships, their handling slows down and calling ships are forced to wait for idle berths. This trend is typical for the case of fewer cranes, i.e., $TQ=8$ and 10 in most of problem instances.

Figs. 10-11

In fact the problem instances with ship arrival interval of 2 h show very long service times regardless of TQ and therefore unrealistic scenario settings. For the instances of more than 2 h interval, even with some exceptions TQ more than 14 does not affect the service time. This implies that 14 cranes are enough to serve calling ships efficiently with a 4 or 5 berth terminal.

We conducted the experiments with two different terminal capacities: 4 and 5 berths. As the number of calling ships is common between the two terminals, it is envisaged that the 4-berth terminal results in longer total service time than the 5-berth one. However, with the scenario settings in the experiments, there is no significant difference between the terminals. This may be due to the fact that the crane scheduling complexity limits the expected terminal performance with a larger quay capacity.

6. Summary and Conclusions

Berth space and quay cranes are expensive assets at maritime container terminals. In the past decade, efficient berth scheduling has been investigated by many researchers. While quay crane is one of the key equipment for efficient ship handling, little attention has been paid to it. To achieve a quick ship turnaround, a maritime container terminal provides as many cranes as possible to a specific ship; however, it may turn to be an excessive investment especially due to the high crane costs, when a fluctuation in ship calling frequency is observed in the planning horizon of the terminal system. Therefore, a competitive terminal needs to efficiently operate cranes with a reasonably limited number of them.

Within the above mentioned context this paper addressed efficient scheduling of simultaneous berth and crane at a container terminal. In previous studies of the berth and crane scheduling problem, both assets were separately optimized. This paper proposed a

genetic algorithm-based heuristic, which iterates the procedure of determination of berth scheduling and crane scheduling at the same time. Though the solution quality by the algorithm was not examined, e.g., by the optimal solution or lower bound of the problem objective function, this paper proposed a logical procedure of scheduling berth and crane simultaneously.

In addition, some problem constraints may not hold true to some extent in practice, i.e., no crane transfer performs when the transfer must get through ships under handling and no ship handling gets started without the predetermined number of cranes to be needed. These constraints are interrelated. For instance, there is an increase or decrease in the number of cranes that are engaged in a ship's handling if some crane transfer takes place in the course of its handling. Nevertheless, in a large scale of container terminal with a number of berths such as HIT of Hong Kong, Port of Singapore and ECT of Rotterdam, it could be envisaged that frequent crane transfer while ships are under handling operation is troublesome for the handling task of the ship between the origin and destination berths for the crane transfer. From this point of view, the proposed problem does not lose the generality of crane scheduling in container terminal operations. Even if some issues (hypothetical or impractical) are embedded in the problem, the algorithm provides the upper bound of a scheduling solution, which is useful for terminal planning.

Finally, this study did not consider the relationship between the handling time and the number of cranes. That is, the B&CAP does not explicitly take into account the handling time as a function of the number of cranes, where a ship starts its handling before the predetermined number of cranes become available. The problem with the addition of this parameter will be a topic for future research.

Acknowledgements

This research is partially supported by the Grant-in-Aid for Scientific Research of

Appendix A: Labeling algorithm for the maximum flow problem

The labeling algorithm for the maximum flow problem that determines y_{ij} (the flow on each arc (i, j)) for the maximum flow from a source node s to a sink node t is as follows, where $\text{pred}(j)$ is the predecessor label to trace back from t to s in procedure *augment*, u_{ij} is the capacity of arc (i, j) and $r_{ij}(=u_{ij}-y_{ij}+y_{ji})$ is the residual capacity of arc (i, j) .

algorithm *labeling*;

begin

while t is labeled **do**

begin

 unlabel all nodes;

 set $\text{pred}(j) := 0$ for $j \in N$;

 label node s and set $\text{LIST} := \{s\}$;

while $\text{LIST} \neq \emptyset$ or t is unlabeled **do**

begin

 remove a node i from LIST;

for each arc (i, j) in residual network emanating from node i **do**

if $r_{ij} > 0$ and node j is unlabeled **then** set $\text{pred}(j) := i$, label node j ,

 and add j to LIST

end;

if t is labeled **then** *augment*

end;

end;

procedure *augment*;

begin

use the predecessors labels to trace back from t to s to obtain an augmenting path

P from s to t ;

$\delta := \min\{r_{ij} : (i, j) \in P\}$;

Augment δ units of flow along P and update the residual capacities r_{ij} ;

end;

References

- Ahuja, R.K., Magnanti, T.L., Orlin, J.B. 1993. Network Flows, Prentice-Hall, Englewood Cliffs.
- 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.
- Daganzo, C.F., 1989. The crane scheduling problem. Transportation Research Part B 23, 159-175.
- 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., 2007. Berth allocation at indented berths for mega-containerships. *European Journal of Operational Research* 179, 579-593.
- Kim, K.H., Moon, K.C., 2003. Berth scheduling by simulated annealing. *Transportation Research Part B* 37, 541-560.
- Kim, K.H., Park, Y.-M., 2004. A crane scheduling method for port container terminals. *European Journal of Operational Research* 156, 752-768.
- Lai, K.K., Shih, K., 1992. A study of container berth allocation. *Journal of Advanced Transportation* 26, 45-60.
- Lee, D.-H., Song, L., Wang, H., 2005. A bi-level programming model and solutions of berth allocation and quay crane scheduling. *Proceedings of the 1st International Conference on Transportation Logistics*, available in CD.
- 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.

Peterkofsky, R.I., Daganzo, C.F., 1990. A branch and bound solution method for the crane scheduling problem. *Transportation Research Part B* 24, 159-172.

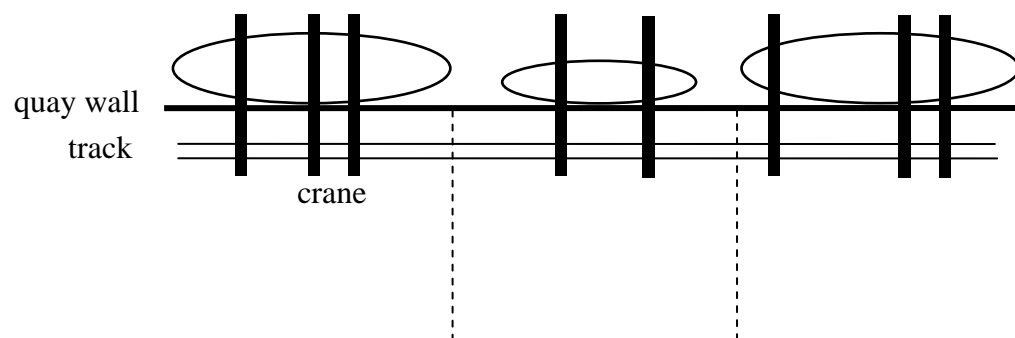


Figure 1. Crane allocation

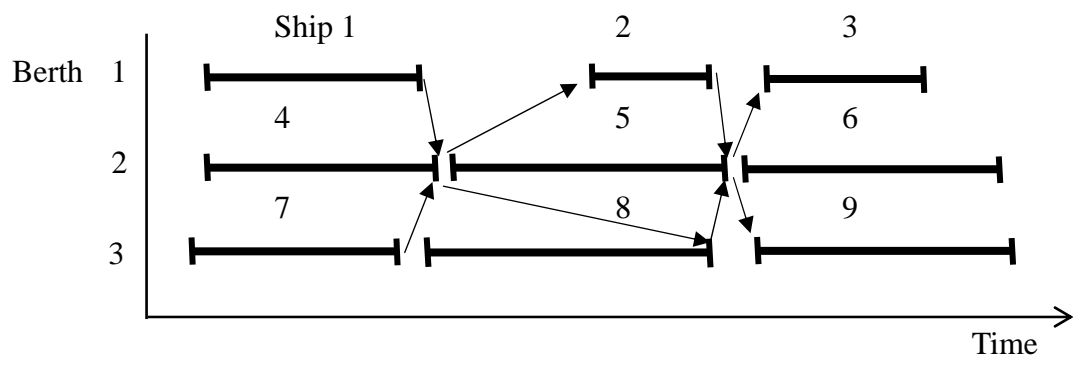


Figure 2. Crane transfer

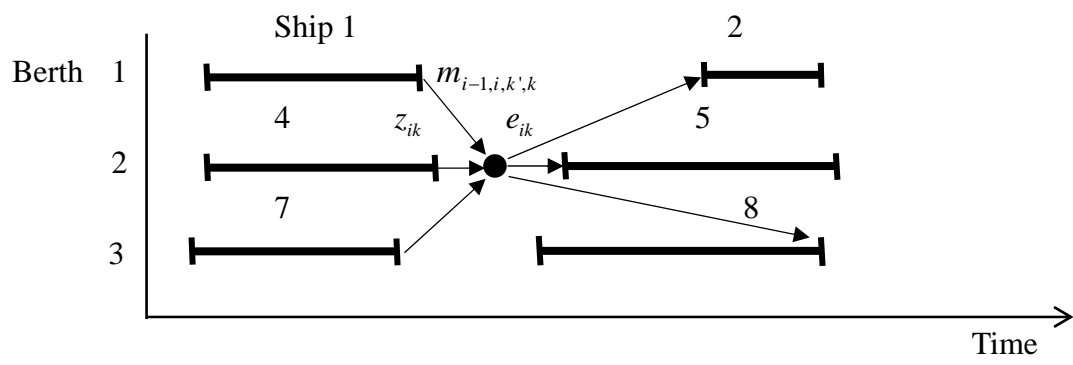


Figure 3. Detailed movement of crane

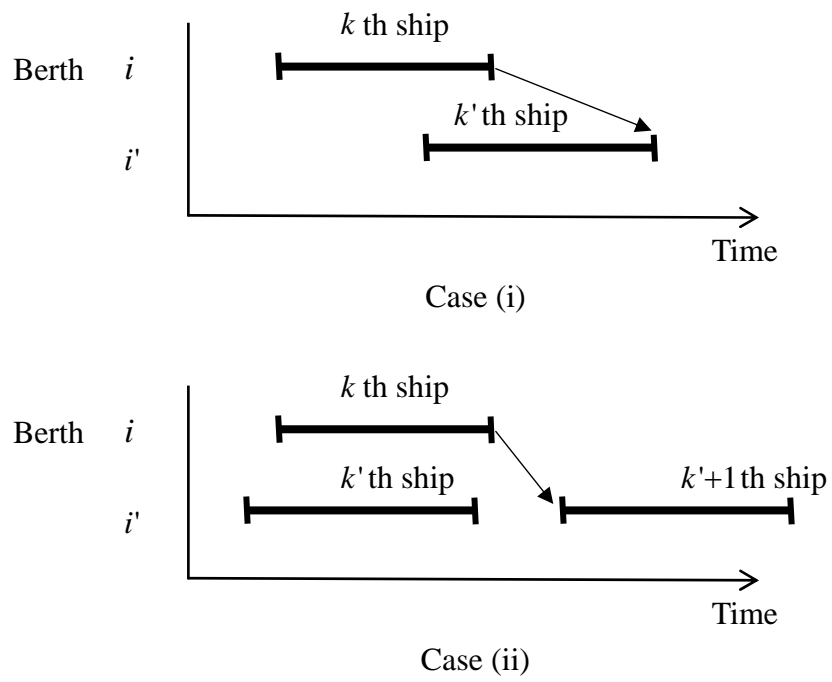


Figure 4. Possible transfer cases

Chromosome	2	8	5	9	0	4	7	3	1	6
i	1	1	1	1		2	2	2	2	2
k	1	2	3	4		1	2	3	4	5

Figure 5. Chromosome representation

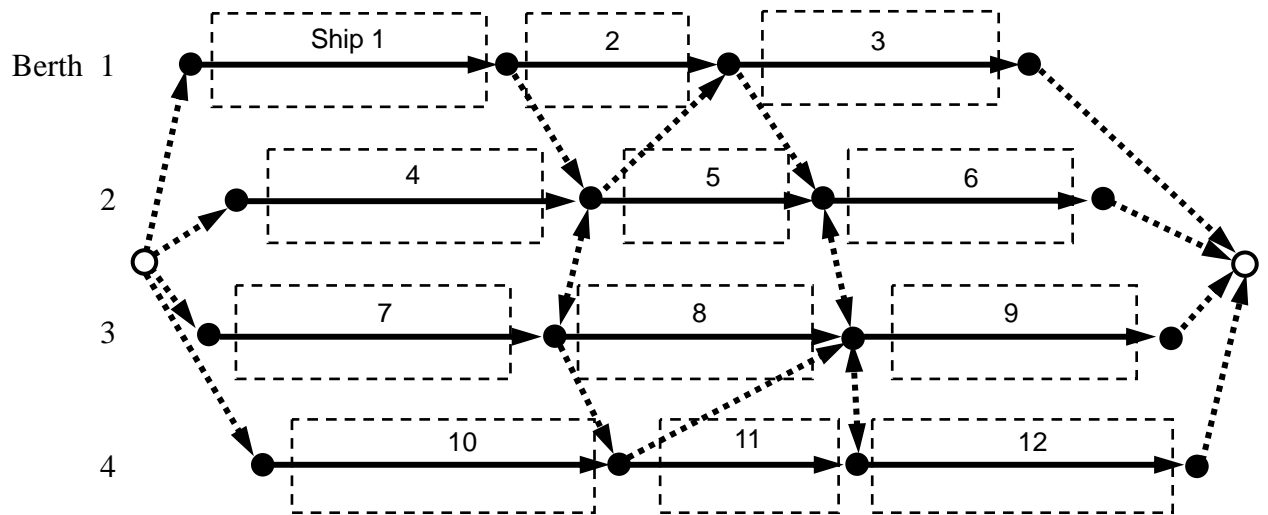


Figure 6. Crane movement network

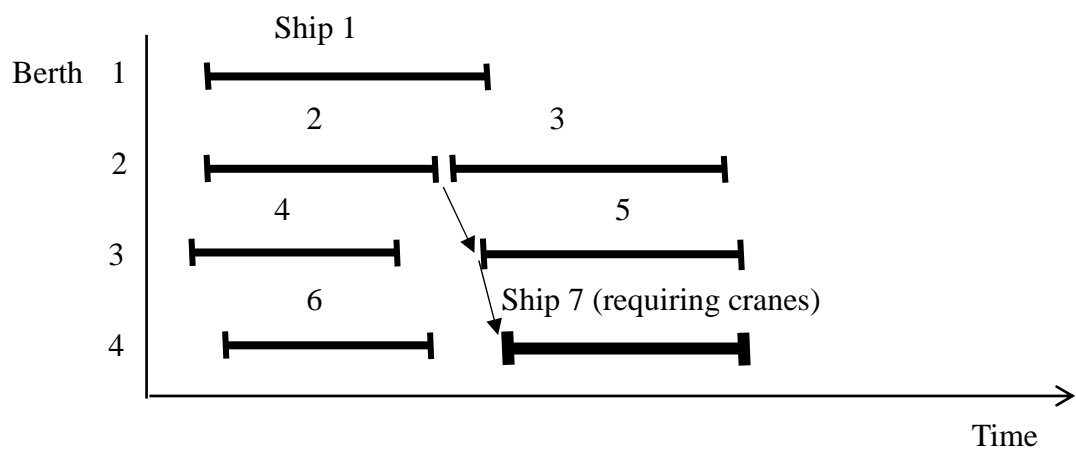


Figure 7. Repeated crane transfer

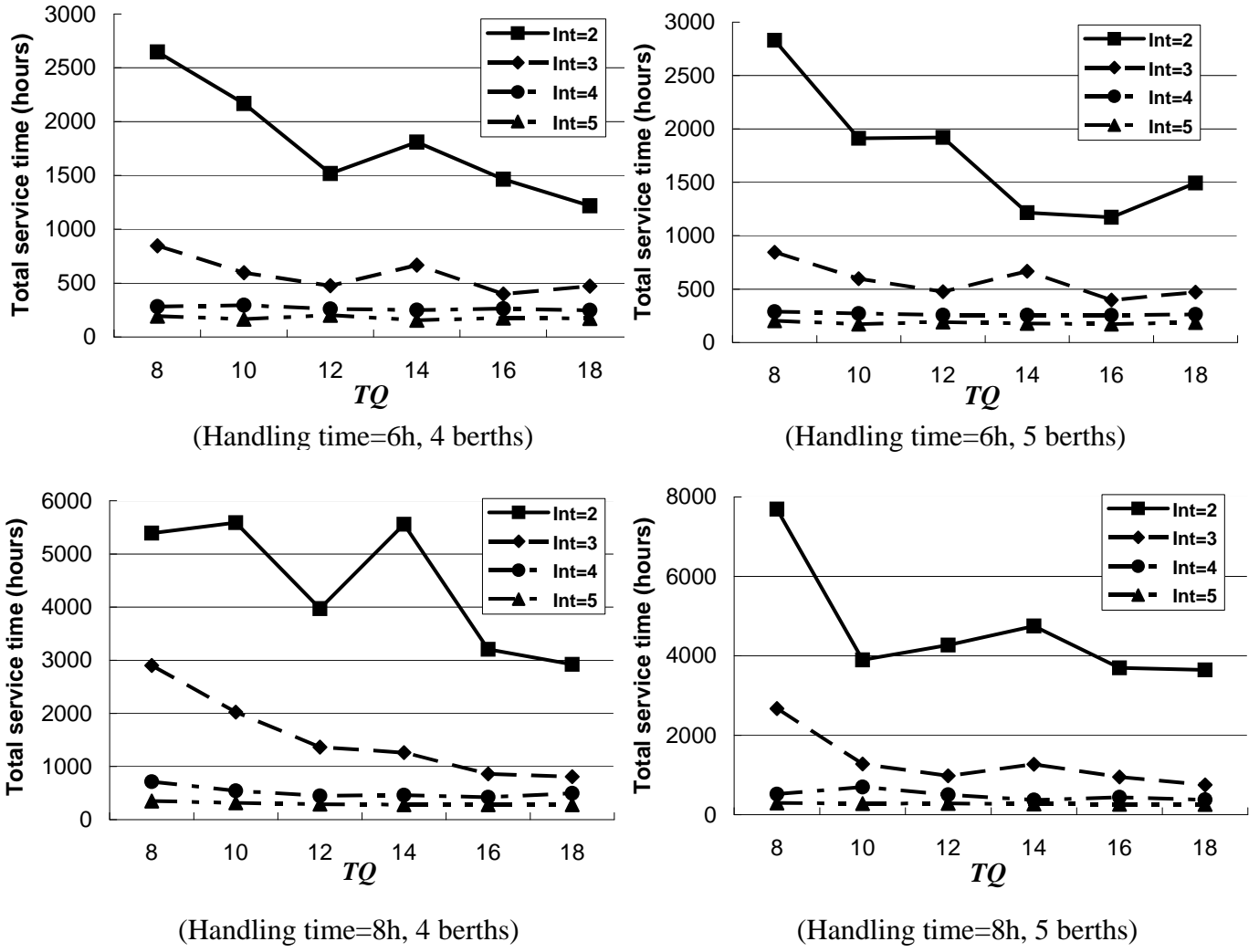


Figure 8. Total service time (seed=1)

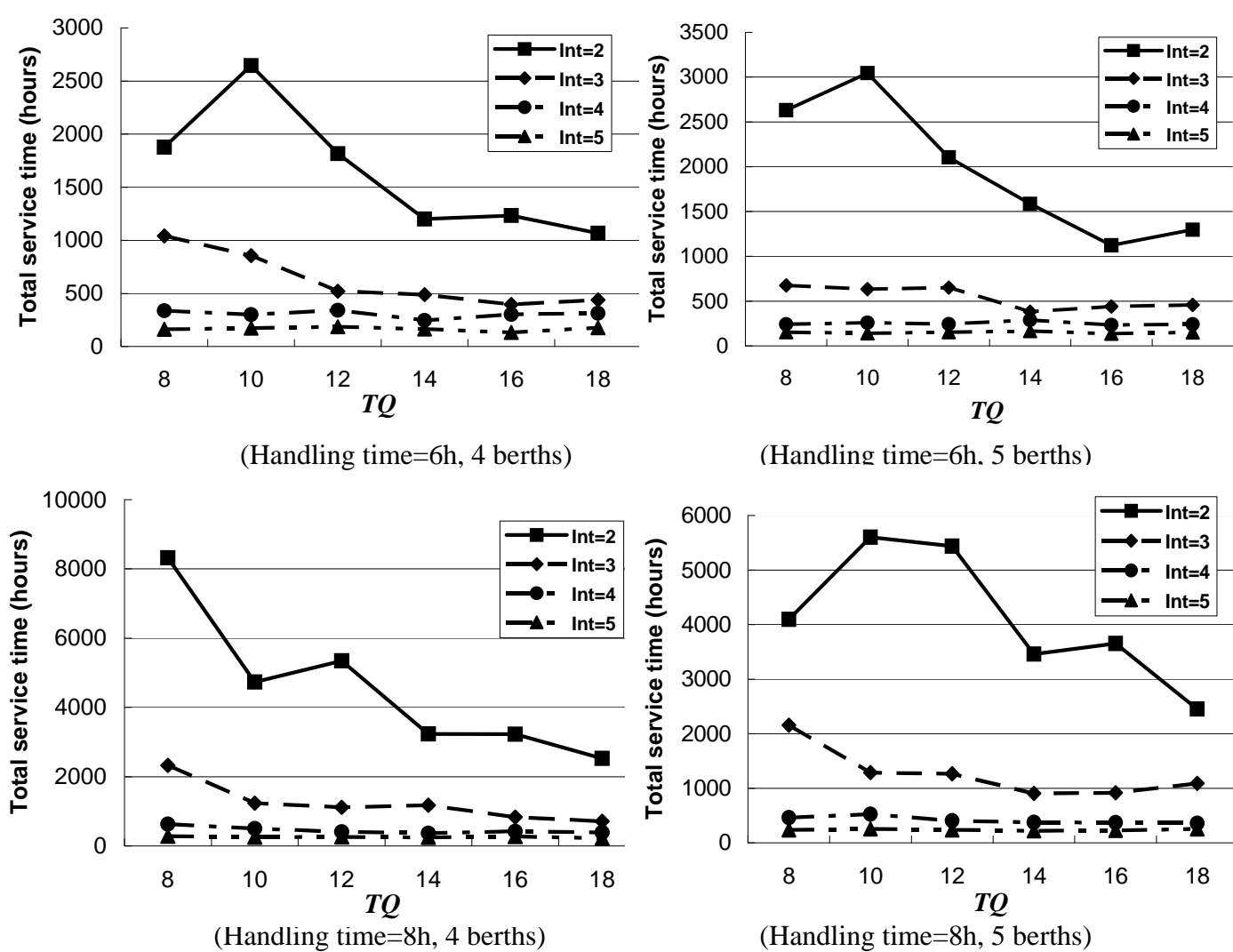
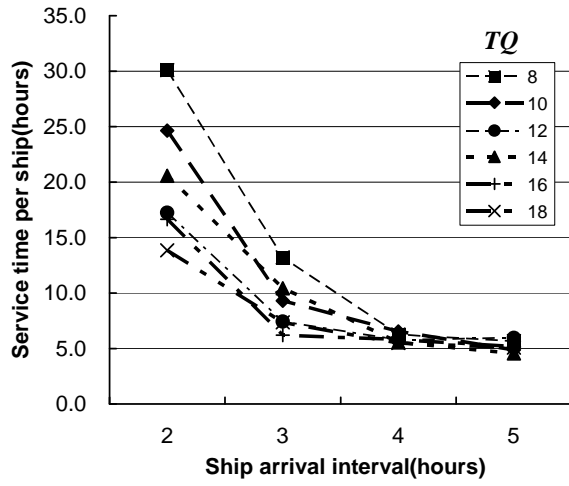
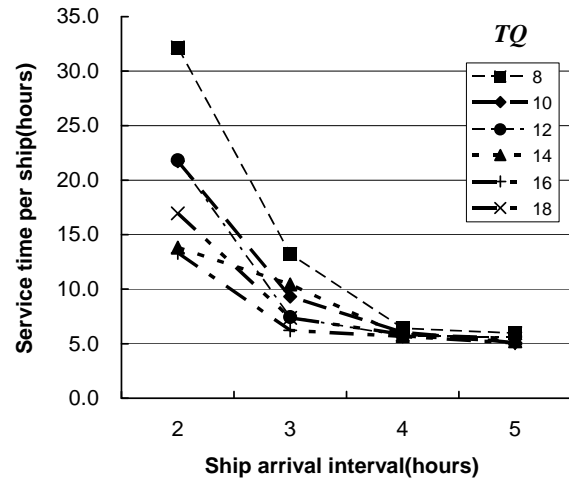


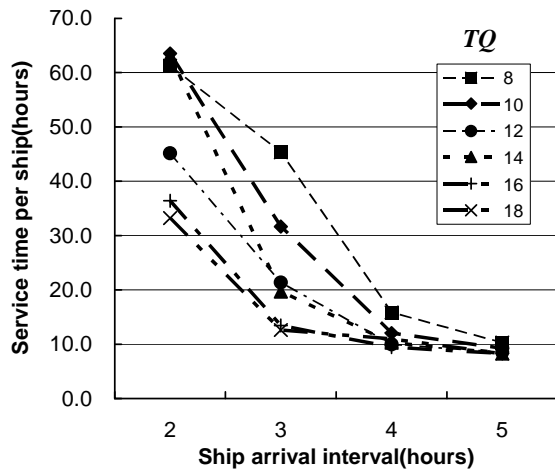
Figure 9. Total service time (seed=2)



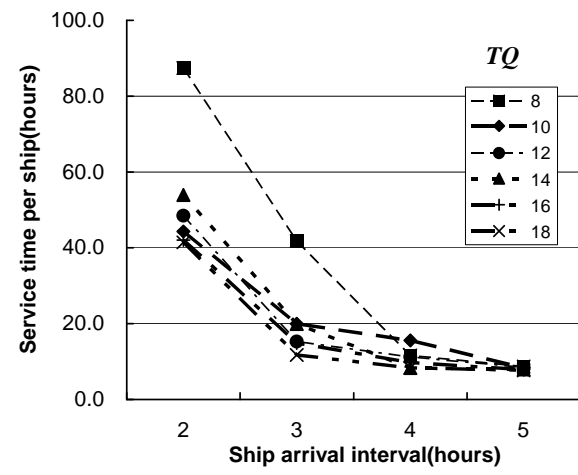
(Handling time=6h, 4 berths)



(Handling time=6h, 5 berths)

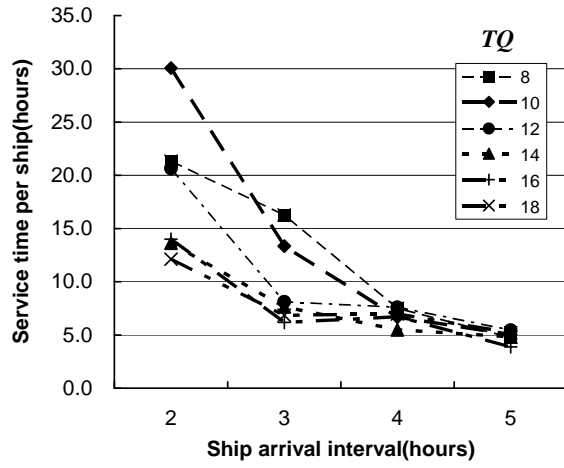


(Handling time=8h, 4 berths)

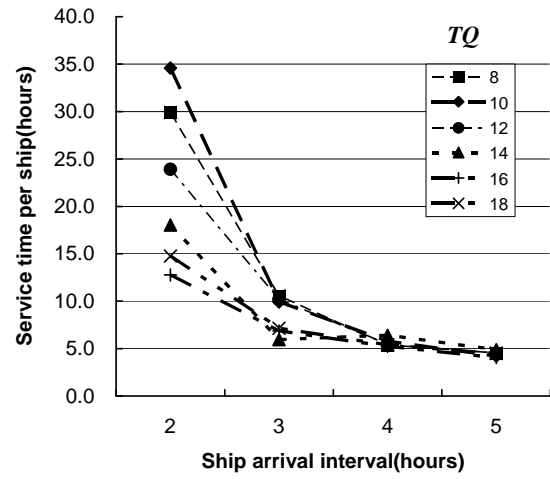


(Handling time=8h, 5 berths)

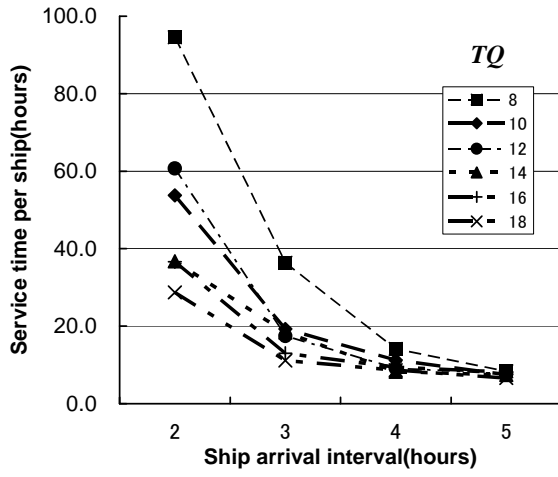
Figure 10. Average service time per ship (seed=1)



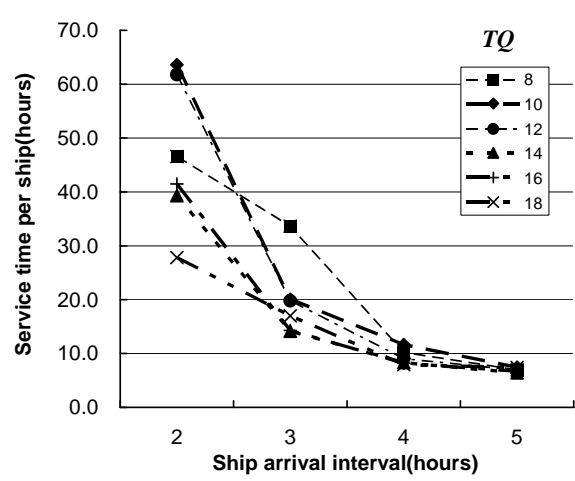
(Handling time=6h, 4 berths)



(Handling time=6h, 5 berths)



(Handling time=8h, 4 berths)



(Handling time=8h, 5 berths)

Figure 11. Average service time per ship (seed=2)