WORKING PAPER

ITLS-WP-16-12

Dynamic discrete berth allocation in container terminals under four performance measures

By
Collins Teye[1] and Michael Bell[1]

[1] Institute of Transport and Logistics Studies, The University of Sydney Business School, Sydney, Australia

June 2016

ISSN 1832-570X

INSTITUTE of TRANSPORT and LOGISTICS STUDIES
The Australian Key Centre in
Transport and Logistics Management

The University of Sydney

Established under the Australian Research Council's Key Centre Program.

NUMBER: Working Paper ITLS-WP-16-12

TITLE: **Dynamic discrete berth allocation in container terminals under four performance measures**

ABSTRACT: In this paper we develop new models for the dynamic discrete berth allocation problem under four performance measures (PM). The models allow for both dynamic berth availability and dynamic arrival of vessels within the planning time horizon. The new formulation allows the four models to be compared in terms of both model complexities and solutions. The models were implemented using CPLEX. The paper also proposed four heuristics under one framework for solving large instances of the problem. The study shows that the choice of PM to optimise is very crucial as different optimised PMs lead to different degrees of satisfactions or terminal efficiency.

KEY WORDS: ***Containers terminals, Discrete Dynamic Berth Allocation, Container vessels, Performance Measures***

AUTHORS: **Teye and Bell**

Acknowledgements: This paper has benefited from the financial support of Australian International Trade and Transport Industry Development Fund (AITTIDF). The authors thank David Hensher and John Rose for helping secure the funds for this research.

CONTACT: INSTITUTE OF TRANSPORT AND LOGISTICS STUDIES (C13)

The Australian Key Centre in Transport and Logistics Management

The University of Sydney NSW 2006 Australia

Telephone: +612 9114 1824

E-mail: business.itlsinfo@sydney.edu.au

Internet: http://sydney.edu.au/business/itls

DATE: June 2016

1 Introduction

The berth allocation problem (BAP) involves assigning vessels arriving at the port to berths within a terminal to optimise a given performance measure (Imai et al. 2007). Effective allocation of berths to arriving vessels, especially in a multi-user terminal, is critical for successful terminal operations (Imai et al 2007; Stahlbock and Voβ, 2007). This is due to the fact that the efficiency of the berthing process affects the work of other port resources such as quay cranes used for loading and unloading the vessels, yard cranes, labour and other important and expensive port resources (Stahlbock and Voβ, 2007). Thus poor berth allocation may lead to under utilisation of other port resources and hence an overall reduction in port or terminal performance. With increasing competition not just between ports but also between terminals within the same port, managers are keen to reduce costs by maximising the utilisation of resources (labour, berths, yards, quay cranes) and are constantly looking for ways to be more competitive. Several researchers have therefore been working on optimisation models, not just for improving the efficiency of berth utilisation, but also for improving the efficiency of all areas of port operations (Stahlbock and Voβ, 2007).

Models for berth allocation can be classified into two broad areas; the discrete berth allocation problem (DBAP) and the continuous berth allocation problem (CBAP). The DBAP models assume that the berthing area (wharf) is partitioned into discrete berths, where each berth can be occupied by at most one vessel at a time. Studies on this model type include (Brown et al. 1994; Imai et al. 2003; Cordeau et al. 2005). The models for continuous berth allocation problem (CBAP) treat the berth area (wharf) as one finite linear facility, where several vessels can be moored simultaneously. Under this continuous berthing approach, a vessel is allowed to berth wherever an empty space is available to physically accommodate it (Imai et al. 2007). This class of BAP, as noted in (Imai et al. 2007), could be considered as a form of cutting-stock problem (CSP) where a set of commodities is packed into boxes in an efficient manner. This class of models can be found in (Lim 1998; Park and Kim 2002; Cordeau et al. 2005; and Imai et al. 2007). Models under each of the above two classes can further be segmented into *static* or *dynamic* variants. The static variant assumes that the vessels to be berthed are available for berthing at the port at decision time or before the schedule is constructed. On the other hand, the dynamic variant allows the arrival of vessels at different times during the planning horizon or berthing process. A survey of berth allocation problems can be found in (Bierwirth and Meisel 2009).

The focus of this study is on the dynamic discrete berth allocation problem (DDBAP) under several performance measures. The performance measures considered are the *total* or *average turnaround time* for all vessels (*ATT*), the *maximum turnaround time* or makespan (*Cmax*) of any vessel in the schedule, the *number of late vessels* (*Tardy*) and the m*aximum lateness* (*Lmax*) of any vessel. Each of these performance measures or optimality criteria is discussed in the next section.

2 Optimality Criteria

The goal of any berth allocation problem is to generate the best schedule. However, what is best depends on the application and the chosen performance measure. Depending on the problem faced by the *port or terminal operator*, the goal may be to process vessels as quickly as possible. In other words minimise the total turnaround time of all vessels arriving during the scheduled period. Given the schedule $\mathbb{S}$, let C_j be the turnaround time for vessel $j \in \{1,2,\ldots,|V|\}$, namely the completion time of the vessel, then $\sum_{j\in V} C_j$ becomes the total turnaround time for all the vessels arriving at the port within the schedule period or planning time horizon. With this optimality criterion we care less about the turnaround time of the last vessel in the schedule as long as all the vessels on average receive good service. Dividing this total turnaround time by the number scheduled vessels gives the *average turnaround time*, and can be used as one of the performance indicators at the port or terminal. It is also possible that not all vessels are of equal importance and that the terminal operator might wish to consider this when measuring service quality provided to a vessel. This can be achieved by minimising the total weighted turnaround time $\sum_{j\in V} w_j C_j$ or the average weighted turnaround time $\frac{1}{|V|}\sum_{j\in V} w_j C_j$, where w_j is the weight or priority of vessel $j \in \{1,2,\ldots,|V|\}$.

A performance indicator in which carriers may be more interested is the maximum turnaround time of any vessel in the schedule. This performance measure is called the *makespan*, and is defined as $C_{max} = max_j\{C_j\}$ for a given schedule $\mathbb{S}$.

There are problems or situations where each vessel may have an associated departure time (due date) by which it must depart from the port or by which any operations on it must be completed. This gives rise to two different optimality criteria. The first optimality criterion is the *Maximum Lateness* (L_{max}) of any vessel in the schedule $\mathbb{S}$ and is defined as $L_{max} = max_j\{L_j\}$, where $L_j = max\{0, C_j\text{-}d_j\}$ is the lateness of vessel $j \in \{1,2,\ldots,|V|\}$ and d_j is the due date of vessel *j*. Alternatively, the port or terminal operator may be interested in constructing a schedule that maximizes the number of vessels that can be served before their due dates (not late or tardy).

This condition can be captured in the objective function by defining a dummy variable U_j which equals 0 if vessel j is handled before its departure time or due date d_j and 1 otherwise. Thus we can define the minimisation of $\sum_{j\in V} U_j$ or more generally $\sum_{j\in V} w_j U_j$. If U_j equals 1 then it means the vessel is late or *tardy* and the objective is to minimise the total number or total weighted number of *tardiness* in the schedule.

In this paper we investigated the implications of using the above four optimality criteria described in berth scheduling. Minimising one optimality criterion may not necessarily imply minimising the others. Thus each optimality criterion is expected to yield a different schedule and hence a different port or terminal performance indicator or efficiency measure.

3 Literature Review

Early studies on the berth allocation problem include (Brown et al. 1994; Imai et al. 1997; Lim 1998 & Li et al. 1998) and a detailed review of the subject can be found in (Stahlbock and Voβ 2007). The dynamic continuous version was studied by (Lim 1998). He exploits the graphical representation of the problem to develop a heuristic algorithm to solve the problem. (Li et al. 1998) also considered the continuous BAP with the objective of minimising the makespan (the completion time of the last vessel in the schedule). In (Guan et al. 2002), a heuristic for the continuous BAP with an objective that minimises the total weighted completion time of vessel services is developed. The problem was also studied by (Park and Kim 2002) with an objective function that minimises the cost of delayed departures of vessels or total tardiness and employed a subgradient algorithm to solve the problem. The problem was tackled by (Kim and Moon 2003) using a simulated annealing method.

The static discrete version of the berth allocation problem was studied by (Brown et al. 1994). They formulated the problem as an integer-programming model for assigning a vessel to a berth with several constraints. Here the performance measure considered was the total service time of all vessels. (Dai et al. 2004) formulated the static berth allocation as a rectangle problem and developed a heuristic algorithm to solve it. (Cordeau et al. 2005) considered both the discrete and continuous versions of the BAP with the objective of minimising the total weighted service time of all vessels. They provided mathematical formulations of both problems and solved small instances to optimality. They also employed the *Tabu Search* algorithm for the discrete case and developed a heuristic for solving large instances of the continuous problem. In (Imai et al. 1997) the static discrete BAP for commercial ports is addressed with a multi-criteria objective function that combines the minimisation of service order dissatisfaction and the maximization of berth performance.

As (Imai et al. 2007) noted, solutions to the static BAP are more useful for busy ports. But competition is increasing between ports and between terminals operators of the same port, which means carriers are less likely to tolerate long delays at the port. (Imai et al. 2001) then extended the static version of the BAP to a dynamic treatment that is similar to the static treatment, but with the difference that some vessels arrive while work is in progress with the objective of maximizing berth performance. They developed a Lagrangian heuristic by to solve the problem as the problem is known to be NP-hard. (Nishimura et al. 2001) also considered a similar dynamic version of the BAP and employed genetic algorithms to solve that problem with the results shown to be comparable to that of Imai et al. (2001). However, there were no computational comparisons between two proposed heuristics of the problem. The model in (Imai et al. 2001) was extended in (Imai et al. 2001) by considering vessels with different priorities with the objective function being the total weighted service time of all vessels.

Reformulation of the DDBAP was presented by (Cordeau et al. 2005) in a form of a Multi-Depot Vehicle Routing Problem with Time Windows (MDVRPTW) and had the objective of minimising the total service time of all vessels. Here, vessels were considered as customers and the berths as depots. The re-formulation does not obviate the need for a heuristic as the MDVRPTW is also NP-Hard, so a tabu search algorithm was developed for solving large instances of the problem. (Imai et al. 2007) proposed the BAP with simultaneous berthing of multiple vessels at the indented berth, which is potentially useful for fast turnaround of mega-containerships. (Christensen and Holst 2008) presented another re-formulation of the problem by considering it as a generalised set-partitioning problem. In their formulation time was discretised and for each berth and time interval a packing constraint ensures that at most one vessel can be at the berth at any given time. (Buhrkal et al. 2011) considered the computational advantage of the three main formulations through extensive numerical tests and suggested that the generalized set-partitioning model (first proposed in Christensen and Holst 2008) outperforms all other existing models.

To the best of our knowledge, the dynamic berth allocation problem under the range of performance measures proposed in this paper has yet to be studied. Most of the existing papers focus mainly on the static berth allocation problem, where the central issue is to obtain a good plan to pack the vessels waiting and arriving within the scheduling window, and the few dynamic berth allocation papers considered only one performance measure without any regard to the performances of other PMs.

In this paper proposed new mathematical models for the dynamic discrete berth allocation problem (DDBAP) under four performance measures. The new formulation allows the four proposed models (one for each PM) to be compared in terms of both model complexities and solutions. The models were each solved to optimality using the CPLEX software for a small-sized instance of the problem. We also proposed four (one for each PM) heuristics under one algorithm framework for large instances of the problem. We then demonstrated through extensive numerical examples of 48,000 test instances that the choice of performance measure (PM) to optimise is crucial as different optimised PMs lead to different service orders and different degrees of customer satisfaction as optimising one PM leads to the deterioration or worsening other PMs. As will be demonstrated in this paper it is important for terminal managers to be aware of and familiar with all the key performance measures considered here and to know the most appropriate one to use for any given situation.

4 Problem Formulation

4.1 Introduction

This section presents the mathematical models for the discrete dynamic berth allocation problem under the four objective functions (performance measures). In the formulation we consider a set of discrete berths B at the port indexed by $i \in \{1,2,\dots,|B|\}$, where $|B|$ is the number of berths, a set of vessels V to be handled at the berths with index $j \in \{1,2,\dots,|V|\}$, with arrival time a_j, deadline or due date d_j and handling time each vessel at each berth h_{ij}. In this paper $N = |V|$ is the number of positions in the schedule expressing the order in which the vessels are handled at each berth. For example, Figure 1 shows a feasible schedule $\mathbb{S} = \{V_3, V_1, V_4, V_2\}$, which means that the first vessel to be handled is vessel 3 followed by vessel 1, with vessel 2 being the last to be handled. The schedule means that if for example vessel 1 is scheduled to be handled at berth 1 (assume berth 1is idle) but is not available or yet to arrive, berth 1 would be forced to wait until it arrives as shown in Figure 1. Also, vessel 2 has to wait in the queue until berth 2 finishes handling vessel 3 before it is berthed. It is therefore feasible that an optimal schedule may result in forced berth idle time and/or vessel waiting time.

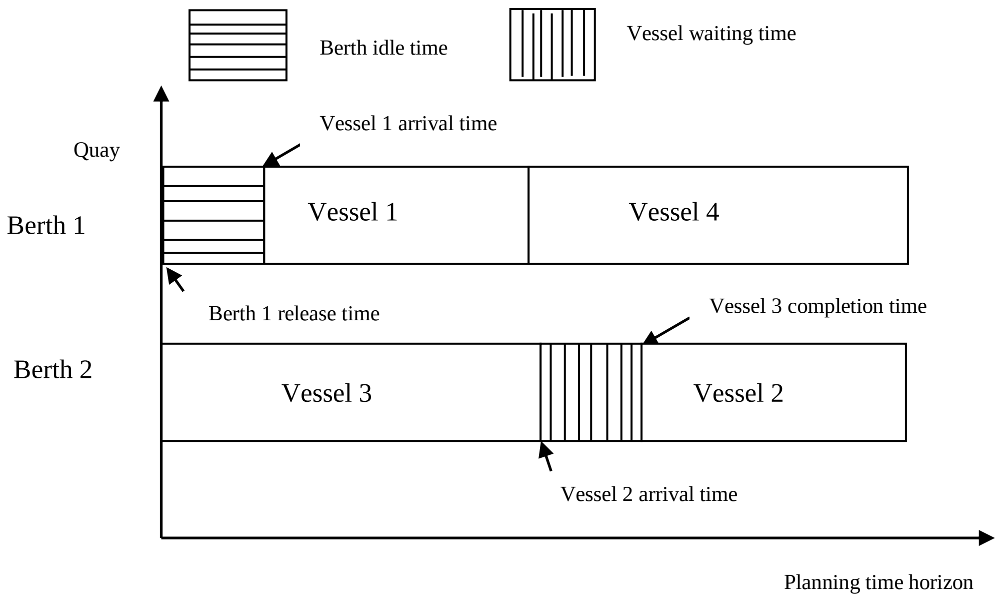

Figure 1: Feasible discrete dynamic berth allocation solution

A summary of the key assumptions underlying the formulated models and variable definitions are:

Main assumptions

1. The quay is partitioned into discrete sections (berths).
2. Each vessel can be handled at any berth.
3. Only one vessel can be served at each berth at a time.
4. There are no physical or technical restrictions such as vessel draft and water depth that affect the schedule.
5. Vessel handling time depends on the assigned berth.
6. Vessels can only be served after they arrive.
7. Each vessel has expected arrival and departure times.
8. Vessel handling at a berth cannot be interrupted once started.
9. All the models apply to a given planning time horizon.

Definition of input variables

V The set of vessels to be handled at the port indexed by $j \in \{1,2, \ldots, |V|\}$.

B The set of berths at the port or terminal indexed by $i \in \{1,2, \ldots, |B|\}$.

N The number of possible vessel positions in a schedule indexed by $p = 1,2, \ldots., N$.

h_{ij} The handling time of vessel $j \in \{1,2, \ldots, |V|\}$ at berth $i \in \{1,2, \ldots, |B|\}$.

a_j The arrival time of vessel $j \in \{1,2, \ldots, |V|\}$.

d_j The expected departure time or date of vessel $j \in \{1,2,\dots,|V|\}$ from the berth

w_j The weight or importance of vessel $j \in \{1,2,\dots,|V|\}$

M Large positive constant

Output variables

S_{ip} The setup time of the pth vessel in the schedule at berth by $i \in \{1,2,\dots,|B|\}$.

R_{ip} The release time of berth $i \in \{1,2,\dots,|B|\}$ to handle the pth vessel in the schedule

C_j The completion time of vessel $j \in \{1,2,\dots,|V|\}$

C_{max} The completion time of the last vessel in the schedule

L_{max} The maximum lateness of any vessel in the schedule

Note that some variables, like S_{ip}, R_{ip} and C_{ip}, are points in time to be decided by the problem (and hence upper case), while h_{ij} is a duration. The arrival and expected departure times, a_j and d_j, are given points in time and therefore lower case. Also the variable S_{ijp} equals S_{ip} if vessel $j \in \{1,2,\dots,|V|\}$ is the pth vessel in the schedule and is handled at berth $i \in \{1,2,\dots,|B|\}$, and 0 otherwise.

Key decision variables

$X_{ijp} = 1$ if the vessel $j \in \{1,2,\dots,|V|\}$ is the pth vessel in the schedule and is handled at berth $i \in \{1,2,\dots,|B|\}$, and 0 otherwise.

$U_j = 1$ if the vessel in the schedule is late and 0 otherwise.

Based on the above dentitions and assumptions we provide as simplified and intuitive formulation of the DDBAP under each of the four performance measure.

4.2 Minimising the Total Weighted Turnaround Time (TTT)

$$TTT = Min\left\{\sum_{j\in V} w_j C_j\right\}$$

Subject to:

$$\sum_{i\in B}\sum_{j\in V} X_{ijp} = 1;\ \ p = 1,2,\dots N \qquad (1)$$

$$\sum_{i\in B}\sum_{p=1}^{N} X_{ijp} = 1;\quad \forall j \in V \qquad (2)$$

$$R_{i1} = \tau_i;\ \ \forall i \in B \qquad (3)$$

$$R_{ip} \geq S_{ip-1} + \sum_{j\in V} h_{ij} X_{ijp-1};\quad p = 2,\dots N, \forall i \in B \qquad (4)$$

$$S_{ip} \geq \sum_{j \in V} a_j X_{ijp} \; ; \quad \forall i \in B \, , p = 1, \dots N \qquad (5)$$

$$S_{ip} \geq \; R_{ip;} \; \forall i \in B \, , p = 1, \dots N \qquad (6)$$

$$S_{ijp} \leq M X_{ijp}; \quad \forall i \in B, j \in V, p = 1,2, \dots, N \qquad (7)$$

$$S_{ijp} \leq S_{ip}; \quad \forall i \in B, j \in V, p = 1,2, \dots . N \qquad (8)$$

$$S_{ijp} \geq S_{ip} - M(1 - X_{ijp}); \quad \forall i \in B, j \in V, p = 1,2, \dots . N \qquad (9)$$

$$C_j \geq \sum_{i \in B} \sum_{p=1}^{N} S_{ijp} + \sum_{i \in B} \sum_{p=1}^{N} h_{ij} X_{ijp} \; ; \; \forall j \in V \qquad (10)$$

$$X_{ijp} = \{0,1\}; \;\; \forall i \in B, j \in V, p = 1,2, \dots . N \qquad (11)$$

$$S_{ip} \geq 0 \; ; \; S_{ijp}; R_{ip} \geq 0; \; C_j \geq 0; \; \forall i \in B, j \in V, p = 1,2, \dots . N \quad (12)$$

The objective function minimises the total completion times of the all the vessels in the schedule. Constraint (1) ensures that each position in the schedule is occupied by exactly one vessel and served at one berth. Constraint (2) ensures that each vessel occupies exactly one position in the schedule and it served at one berth. Constraint (3) gives the time each berth is available to start handling vessels in the schedule. The constraint allows for the dynamic availability of berths as well as the dynamic arrival of vessels. This flexibility could be very useful in practice since the berth can be busy handling vessels from the previous schedule or undergoing repair or maintenance works. Constraint (4) ensures that no vessel is assigned to a berth when it is busy. Constraint (5) ensures that a vessel is not served unless it arrives at the port. Constraint (6) ensures that a berth cannot serve a vessel unless it is available and the vessel arrives at the port. Constraint (7), (8) & (9) satisfy the definition of variable S_{ijp} which equals S_{ip} if vessel $j \in \{1,2, \dots, |V|\}$ is the pth vessel in the schedule and is handled at berth $i \in \{1,2, \dots, |B|\}$, and 0 otherwise. Constraints (10) compute the completion time of each vessel. Constraint (11) guarantees integer solutions for the corresponding decision variables. Constraint (12) ensures that the corresponding decision variables take on non-negative values.

4.3 Minimising the Makespan

$MS = Min \; C_{max}$

Subject to constraints (1) to (12) and

$$C_j \leq C_{max} \; ; \; \forall j \in V \qquad (13)$$

$$C_{max} \geq 0 \qquad (14)$$

The objective function minimises the makespan or maximum completion time of any vessel in the schedule (C_{max}). Constraint (13) ensures that the completion time of any vessel at any berth is not greater than the makespan. Constraint (14) ensures that the makespan takes on a non-negative value.

4.4 Minimising the Total Weighted Tardiness

$$Tardy = Min \left\{ \sum_{j \in V} w_j U_j \right\}$$

Subject to constraints (1) to (12) and

$$C_j - MU_j + M > d_j \; ; \quad \forall j \in V \tag{15}$$

$$C_j - MU_j < d_j ; \quad \forall j \in V \tag{16}$$

$$U_j = \{0,1\} \; ; \; \forall j \in V \tag{17}$$

The objective function minimises the (weighted) number of vessels that are late or tardy in the schedule. Constraints (15) and (16) ensure that a vessel is only tardy if its completion time is greater than its due date. Constraint (17) ensures that the corresponding decision variable takes on only two values with the value 0 meaning the vessel is either completed on or before its due date and 1 indicating the vessel is late. If the weight of each vessel $w_j = 1, \forall j \in V$, then the problem reduces to minimising total tardiness.

4.5 Minimising Maximum Lateness

$$ML = Min \;\; L_{max}$$

Subject to constraints (1) to (12) and

$$C_j - d_j \leq \;\; L_{max} ; \; \forall j \in V \tag{18}$$

$$L_{max} \geq 0 \tag{19}$$

The objective function minimises the maximum lateness of any vessel in the schedule. Constraint (18) ensures that maximum lateness must not be less than the difference between completion and due date of any vessel. Constraint (14) ensures that the maximum lateness takes on a non-negative value.

5 Illustrative Example and Analysis

5.1 Introduction

In this section we demonstrated the importance of choosing the right performance measure for the problem at hand, as different performance measures result in different schedules and hence may result in less efficient use of resources at the port. The DDBAP with different objective functions considered in section 4 are compared in this section. For simplicity only one instance of the problem was generated and was solved to optimality under the four objective functions or performance measures discussed earlier using the CPLEX software. The four objection functions considered are:

1. The average turnaround time (ATT) which the total turnaround time (TTT) divided by the number of vessels.
2. The makespan (Cmax)
3. The total tardiness (Tardy)
4. The maximum lateness (Lmax)

5.2 Instance Generation

The number of berths and the number of vessels considered in this experiment is 2 and 10 respectively. The characteristics of the berths and the vessels such as vessel handling time at each berth, expected arrival and departure times (due dates) were generated randomly taking into account the number of cranes assigned to each berth. The data are presented in Table 1. For example vessel number 1 is expected to arrive in 8 hours time with 14 hours handling time at berth 1 and 8 hours at berth 2 and is expected to depart in 101 hours time. The times generated are for illustrative purposes only. The weight or importance of each vessel was set to 1.

Table 1: Instance characteristics

Vessel Number	Arrival Time	Due Date	Handling times (Hrs) Berth 1	Berth 2
1	8	101	14	8
2	17	79	156	56
3	11	48	69	29
4	34	136	227	78
5	55	150	167	56
6	20	108	144	45
7	32	90	101	38
8	83	168	246	84
9	78	163	225	78
10	65	151	211	75

5.3 Results and Analysis

Here we compare the performances of the optimisation problem under the four objective functions with the simple priority rule of First Come Fist Serve (FCFS), which is what port mangers often use (Imai et al. 2001). The results presented in Table 2 include the resulting order of service and the computed values of the performance measures (Average turnaround time (ATT), makespan (Cmax), total tardiness (Tardy) and maximum lateness (Lmax). Due to significant differences in the magnitude of the performance measures, the results are expressed in percentage terms. That is the percentage improvements in the four PMs relative to those of the FCFS values. Both Table 2 and Figure 2 show that significant gains in berth efficiency and utilisations can be achieved when berth allocation is optimised instead of the using the FCFS rule. For example, minimising TTT (or ATT) can result in a savings of more 9% of average vessel turnaround time compared with FCFS rule. Minimising the makespan can reduce the maximum turnaround time of a vessel by 31%; the number of late vessels can be reduced by over 25% under total tardiness minimisation and the maximum lateness of a vessel can also be reduced by 43% under *Lmax* minimisation, all compared with the FCFS rule.

Figure 2 shows the deterioration or worsening of each PM under the optimisation of other PMs. For example, the average turnaround time of a vessel deteriorated from over 9% improvement relative to the FCFS rule under ATT minimisation to -15% under total tardiness minimisation, making the FCFS rule appear better if the wrong PM is optimised. The *Cmax* (makespan) and the *Lmax* (maximum lateness) performance measures also recorded their worst values under total tardiness minimisation. The improvement in maximum turnaround time of any vessel in the schedule over the FCFS rule reduced from 31% under *Cmax* minimisation to 6% under *Lmax* minimisation whilst that of maximum lateness of a vessel reduced from 43% under *Lmax* minimisation to just 1% under total tardiness minimisation. These results clearly demonstrate the importance of the choice of PM to optimise to meet customers' requirements or achieve required port efficiency targets as different choice of PMs leads to different service orders and different outcomes.

Table 2: Optimisation models vs the FCFS rule

	Service order ------------------------------------>												
Objective Function	1	2	3	4	5	6	7	8	9	10	Optimal Value	FCFS	% Improvement
ATT	1	3	7	6	5	4	9	8	2	10	202	223	9%
Cmax	1	3	2	6	9	5	8	4	10	7	412	594	31%
Tardy	1	3	7	9	5	6	8	2	10	4	6	8	25%
Lmax	1	2	4	3	7	5	6	10	8	9	244	427	43%

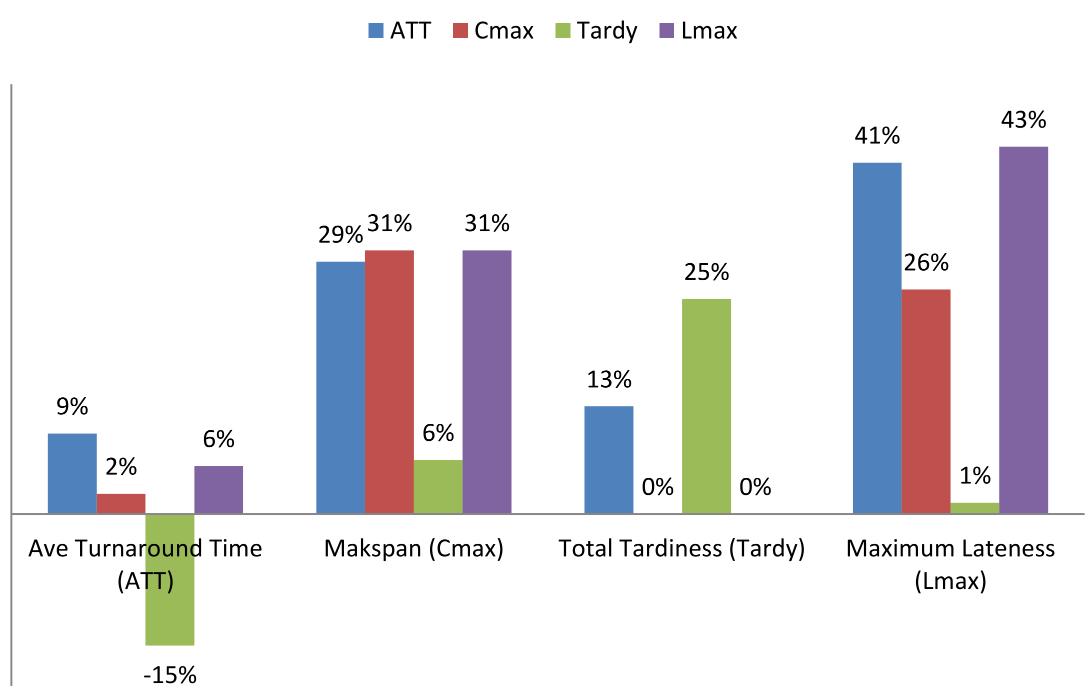

Figure 2: Deterioration of PM under other PM optimization

6 Heuristic Algorithms

The complexity (NP hard) of the problem means that large problem instances are unlikely to be solved by existing commercial solvers such as CPLEX or LINDO. We therefore developed four heuristics algorithm, one for each PM motivated by the insight gained from the formulations above. The performances of the heuristics were compared with the First Come First Served (FCFS) rule using 48,000 generated instances of the problem. The four heuristics consist of three main stages; build, repair and optimise (BRO). We therefore refer to them as *BRO-Z,* where Z is the PM to be optimised. For example, *BRO-Cmax*, is the proposed heuristic for minimising the makespan. The algorithms first construct a feasible solution, then spread the load cross or move vessels across berths and then finally perform local optimisation on each berth. Thus the proposed algorithm can be seen as a general framework that can be adapted to solve all berth allocation problems.

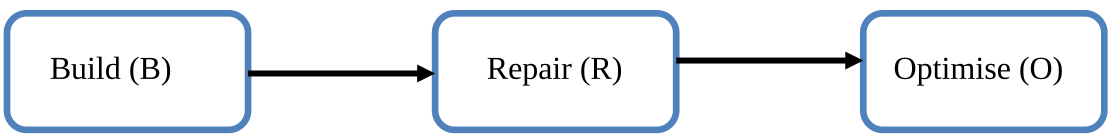

Figure 3: Heuristic framework for DDBAP

Build (B) Algorithm.

Assign the vessels to the berths using the FCFS rule.

The solutions from the *Build (B)* algorithm then go into the *Repair (R)* algorithm for repairs, where it is very likely all the vessels are assigned to a subset of the available berths. The objective of the *R-algorithm* is to spread the load for better utilisation of all berths.

Repair (R) Algorithm

1. *Compute the PM (objective function) for each berth assuming a 1-berth system by doing one of the following :*

   a. If PM is Cmax, schedule the vessels in increasing order of $\frac{a_j}{h_{*j}}$ *(* h_{*j}*. is the handling time on the berth which vessel* j *is assigned to)*

   b. If the PM is ATT, schedule the vessels in increasing order of $a_j + h_{*j}$

   c. If the PM is Tardy schedule the vessels in increasing order of due dates (EDD)

   d. If the PM is Lmax, schedule the vessels in increasing order of due dates
2. *Find the over-utilised berth (*B_o*). That is the berth with the worse evaluated PM*
3. *For the set of vessels assigned to berth* B_o*, find vessel* V_**, which:*
   1. *For Cmax minimisation is the vessel with the earliest arrival time (EAT)*

b. For ATT minimisation is the vessel with largest $a_j + h_{*j}$

c. For Tardy minimisation is the first tardy vessel and

d. For Lmax minimisation is the vessel with largest lateness

4. *Assign* V_* *to the berth* B_* *where* $B_* \neq B_o$ *is the berth where* V_* *has the smallest* $a_j + h_{*j}$ *(arrival + handling time)*
5. *Update the objective function.*

   a. If it improves keep the V_* *on berth* B_* *else*

   a. Mark vessel V_* *as tabu for berth* B_* *(i.e.,* B_* *will not be selected again for* V_**)*
6. *Repeat steps 3-4 until vessel* V_* *is permanently assigned or is tabu on all berth, in which case is left on berth* B_0*.*
7. *Repeats steps 2-6 until any for any selected* B_o *, all* $V_* \in B_o$ *is marked as tabu*

Once the *R-algorithm* is applied the resulting solution is optimised across berths and on each berth using the Swap algorithm described below. The algorithm is a two-staged swap algorithm, where the first stage swaps vessels across berths followed by swapping of vessels on the same berth. A swap is only made permanent if it results in improved PM of interest, else the swap is reversed. In the algorithm below B_i represents the set of vessels assign to berth $i \in B$, $\mathcal{B} = \{\bigcup B_i, \forall i \in B\}$ and V_u represent a vessel in position u for any given berth.

Local optimisation (O)-Swap algorithm

1. For $i \coloneqq 1,2, \dots |\mathcal{B}|$ do:

   1.1 For $j \coloneqq i + 1, 1 + 2, \dots |\mathcal{B}|$ do:

   1.1.1 For $u \coloneqq 1,2, \dots, |B_i|$ do:

   1.1.1.1 $v \coloneqq 1, 2, \dots |B_j|$ do:

   1.1.1.2 Swap the vessel at position u (V_u) on berth B_i with vessel at position v (V_v) on berth B_j

   1.1.1.3 If the swapping improves the PM of interest, make the swap permanent, else reverse the swap.
2. For $i \coloneqq 1,2, \dots |\mathcal{B}|$ do:

   2.1 For $u \coloneqq 1,2, \dots |B_i|$ do:

   2.1.1 $v \coloneqq u + 1, u + 2, \dots |B_i|$ do:

   2.1.1.1 Swap the vessel at position u (V_u) with vessel at position v (V_v)

   2.1.1.2 If the swapping improves the PM of interest, make the swap permanent, else reverse the swap.

6.1 Experimental Study

As noted in Rardin and Uzsoy (2001) conducting experiments to test the quality of the heuristic algorithms is more scientific and also gives an indication of how much the results can be trusted. Thus analysis of using averages (measure of central tendencies) and standard deviations (measure of variability) for a given set of instances, gives a better picture of how the algorithms may perform in practice. So in the study we conducted extensive experiment involving over 48,000 instances of the problems under consideration to ascertain the strengths and weaknesses of the proposed heuristics.

6.2 Instances Generation

Since there is essentially no reference benchmark available, we developed our own instance generator for this study guided by generating instances with features that resembles those in real-life conditions. In this experiment we assumed a planning horizon (T) of one week, which is equivalent to 168 hours assuming a 24/7 operation; 24 different number of berths were drawn from the set $\mathfrak{B} = \{2,3, \ldots .,25\}$ and the number of cranes (N_i) at each berth were randomly generated to lie between 1 and 5 cranes with the average productively of each crane assumed to be 35 (TEU/hr). For each selected berth number $b \in \mathfrak{B}$ at the multi-user container terminal, 10 different number of vessels ($\#V$) were generated as a function of the number of berths; $\#V = b + (v * b), v = 1, \ldots .10; b \in \mathfrak{B}$. The load (L_j) carried by each arriving vessel was randomly drawn from 250 and 8000 (TEUs/vessel) and combined with the number of cranes at each berth the handling time (hours) of a vessel at each berth were computed as:

$$h_{ij} = \frac{L_j}{35 * N_i}; \ j \in V; \ \ i \in B$$

The arrival time of vessels (a_j) were assumed to be a function of the planning horizon (T): $a_j = u * (\alpha * T), u \sim U(0,1)$. 10 different values of α were chosen in set {0, 0.1, 0.2… 0.9}, where $\alpha = 0$ reduces the DDBAP to its static version where all vessels are at the port before the start of the schedule and $\alpha = 0.1$ implies the arrival time of any vessel in schedule must not exceed $0.1 * T$ (or 1008minutes) from the start of the schedule. Finally, to test the robustness of the algorithms we randomly generated 20 instances (*replicates*) of each combination of the above problem characteristics by changing the seed of the random number generator. So in all we generated a total of 48,000 (24*10*10*20) feasible instances for each of the four proposed algorithms plus the FCFS rule. The due date for each vessel was computed as a function of its arrival time, maximum possible handling time on any berth and the length of the planning horizon:

$d_j = a_j + \max\{h_{ij}\} + |T - a_j - \max\{h_{ij}\}| * u; u \sim U(0,1)$

The departure time or due date of some vessels could be higher than the length of the planning horizon, a situation expected even in practice when large vessels call at the terminal. In summary, the size of the instances considered ranges from 2 berths with 4 vessels terminal to 25 berths with 275 vessels terminal.

6.3 Analysis of experimental results

For each generated instance we sort for the percentage improvement in the proposed algorithms (BRO-Z) with respect to the FCFS rule. Thus for each PM, we evaluated the improvements (in percentage) of *BRO-Z* over the FCFS solution for the 48,000 instances and computed the average and standard deviation. The figures below show the comparison results of the four proposed heuristics. Figure 4 shows on average the proposed heuristic (*BRO-ATT*) can reduce the ATT (average vessel turnaround time) of a vessel by 18% (with a standard deviation of about 5%) with respect to the FCFS rule based on the 48,000 generated instances. The figure also shows that on average the *BRO-Cmax* heuristic can reduce the makespan (the maximum time a vessel stays at the terminal) by 23% (with a standard deviation of about 12%); and the *BRO-Tardy* can also on average reduce the number of tardy vessels by 34% with a standard deviation of 15%. Finally, algorithm *BRO-Lmax* makes an average savings of 36% on the maximum lateness of a vessel. The standard deviations show the variations in the performance of the algorithms on the different test instances with algorithm *BRO-Lmax* showing the highest variations and algorithm *BRO-ATT* showing the least. This could suggest that the FCFS rule can provide a reliable performance guarantee under ATT minimisation. This result is generally consistent with the results in section 5 where the FCFS rule had it strongest performance under ATT minimisation and also worse under *Lmax* minimisation.

Figure 5 shows the deterioration or the worsening of each PM under the optimisation of other PMs. For example, only savings of 2%, 5%, and 2% with respect to the FCFS rule were recorded for the average turnaround time PM under the minimisations of *Cmax* (makespan), *Tardy* (number of late vessels), and *Lmax* (maximum lateness) respectively. The savings in the number of late vessels dropped from 34% under *Tardiness* minimisation to 4% under *Cmax* minimisation. Similar conclusions can be drawn from the PMs. The results support the conclusion drawn from the analysis in section 5 where optimising one PM has the potential of worsening other PMs.

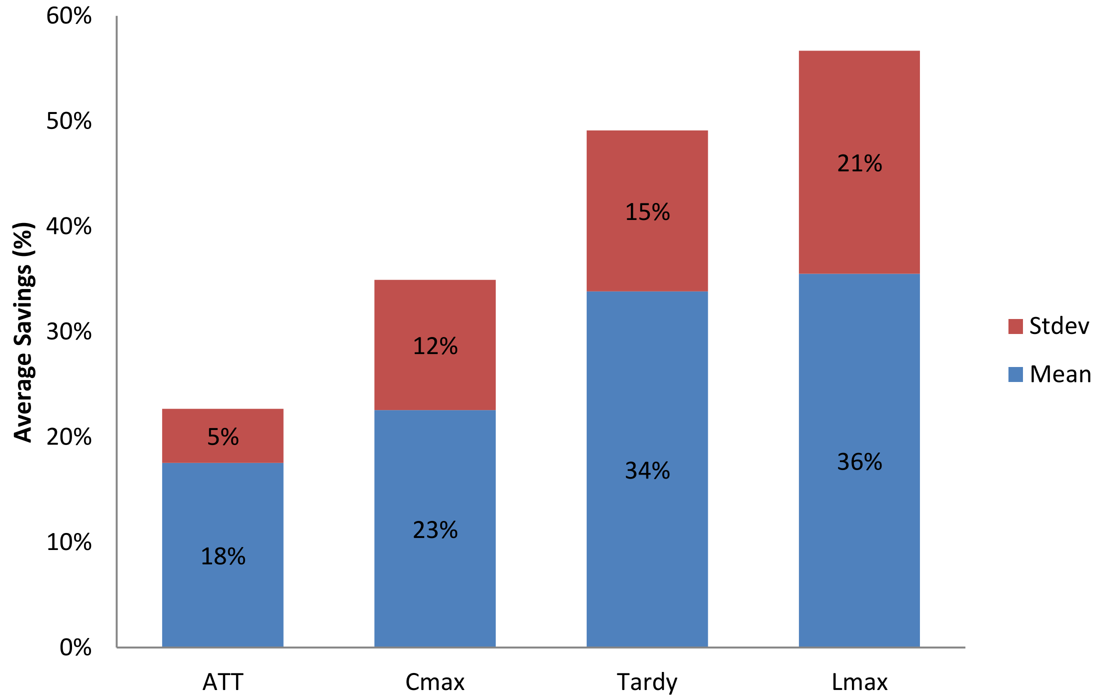

Figure 4: Summary statistics of the model runs

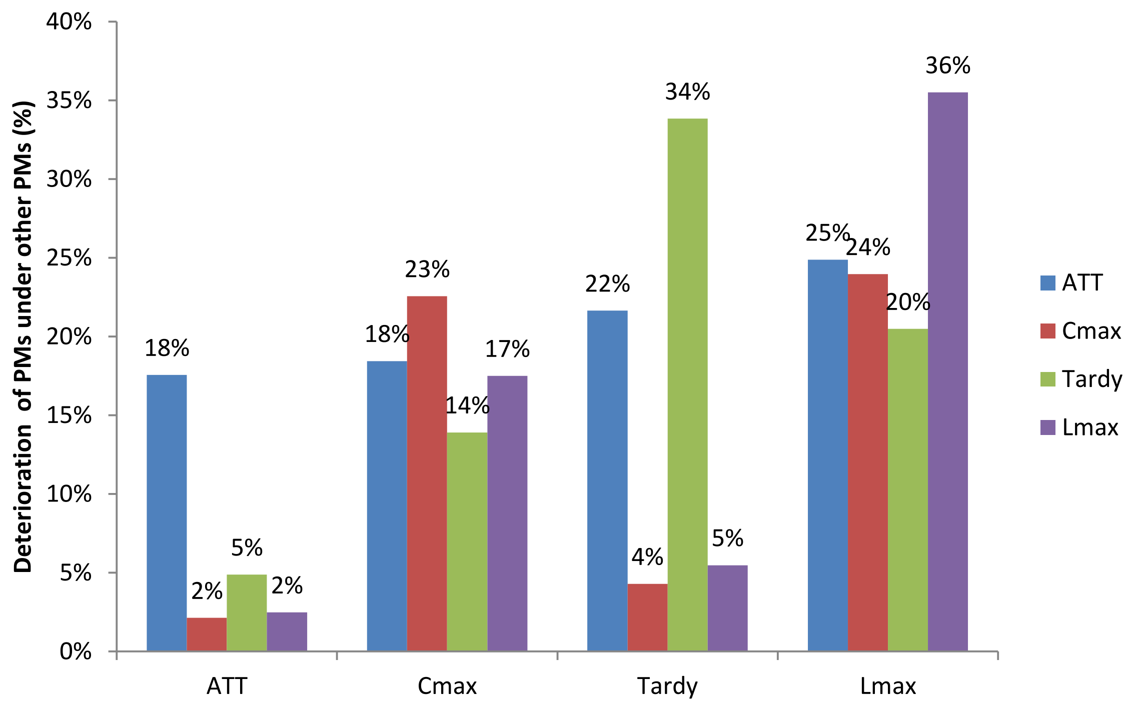

Figure 5: The deterioration of PMs under the optimisation of other PMs

7 Conclusion

This paper presented four new mathematical models for the dynamic discrete berth allocation problem (DDBAP) under four performance measures. The models were successfully implemented using CPLEX for a small-sized instance of the problem and the results were compared with the corresponding results from the FCFS scheduling approach. This paper has also shown that the choice of performance measure to optimise is crucial in berth allocation and may influence customer satisfaction or dissatisfaction and port efficiency measures. For example, the average turnaround time of a vessel deteriorated from over 9% improvement relative to the FCFS rule under ATT minimisation to -15% under total tardiness minimisation, making the FCFS rule appear better if the wrong PM is optimised. The *Cmax* (makespan) and the *Lmax* (maximum lateness) performance measures also recorded their worst values under total tardiness minimisation. The improvement in maximum turnaround time of any vessel in the schedule over the FCFS rule reduced from 31% under *Cmax* minimisation to 6% under *Lmax* minimisation whilst that of maximum lateness of a vessel reduced from 43% under *Lmax* minimisation to just 1% under total tardiness minimisation. Similar conclusions can be drawn for the other performance measures. Thus it is very important for port managers to be familiar with all the performance measures and their implications for the service order and customer satisfaction when planning berth allocations.

This paper has also demonstrated, albeit based on the test instances that significant gains in berth efficiency and utilisation can be made when optimisation models are applied to solve berth allocation problems at ports compared to the First Come First Serve (FCFS) approach. The paper also proposed four (one for each PM) heuristics under one algorithm framework for large instances of the problem and their solution quality were demonstrated through extensive numerical experiment using 48,000 test instances allowing the quality of the solutions (compared with FCFS rule) to measured in terms of statistical average and standard deviations. The results from the experiment support the conclusion drawn from the analysis when the proposed models were solved in CPLEX to optimality.

Further studies are required to test these models on real world data. Extending these models to solve the continuous dynamic berth allocation problem would also be very useful and enrich the literature on berth allocation problems.

References

Brown GG, Lawphongpanich S, Thurman KP (1994) Optimizing ship berthing. Naval Research Logistics 41, 1–15.

Buhrkal, K., Zuglian, S., Larsen, J., Lusby, R (2011) Models for the discrete berth allocation problem: A computational comparison, Transportation Research Part E 47 (2011) 461-472

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.

Christensen CG, Holst CT (2008) Berth allocation in container terminals. Master's thesis, Department of Informatics and Mathematical Modelling, Technical university of Denmark, April 2008.

Guan Y, Xiao WQ, Chueng RK, Li CL (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 WT (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, Hattori M, Papadimitriou S (2007) Berth allocation at indented berths for mega-containerships. European Journal of Operational Research 179, 579–593.

Kim, KH, Moon, KC (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.

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.

Bierwirth C, Meisel F (2009) A survey of berth allocation and quay cranes scheduling problems in container terminals. European Journal of Operational Research 202 (2010) 615–627

Park KT, Kim KH, (2002) Berth scheduling for container terminals by using a sub-gradient optimization technique. Journal of the Operational Research Society 53, 1054–1062.

Park YM, Kim KH (2003) A scheduling method for berth and quay cranes. OR Spectrum 25, 1–23.

Steenken D, Voß S, Stahlbock R (2008) Container terminal operations and operations research a classification and literature review, OR Spectrum 26:3–49.