Journal Pre-proofs

The tactical berth allocation problem with time-variant specific quay crane assignments

Emmanouil Thanos, Tulio Toffolo, Haroldo Gambini Santos, Wim Vancroonenburg, Greet Vanden Berghe

PII: S0360-8352(21)00072-3
DOI: <https://doi.org/10.1016/j.cie.2021.107168>
Reference: CAIE 107168

To appear in: *Computers & Industrial Engineering*

Received Date: 15 April 2020
Accepted Date: 27 January 2021

Please cite this article as: Thanos, E., Toffolo, T., Santos, H.G., Vancroonenburg, W., Berghe, G.V., The tactical berth allocation problem with time-variant specific quay crane assignments, *Computers & Industrial Engineering* (2021), doi: <https://doi.org/10.1016/j.cie.2021.107168>

This is a PDF file of an article that has undergone enhancements after acceptance, such as the addition of a cover page and metadata, and formatting for readability, but it is not yet the definitive version of record. This version will undergo additional copyediting, typesetting and review before it is published in its final form, but we are providing this version to give early visibility of the article. Please note that, during the production process, errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

© 2021 Published by Elsevier Ltd.

The tactical berth allocation problem with time-variant specific quay crane assignments

Emmanouil Thanos^a, Tulio Toffolo^b, Haroldo Gambini Santos^b, Wim Vancroonenburg^a, Greet Vanden Berghe^a

^a *Department of Computer Science, CODeS, KU Leuven, Belgium*

^b *Department of Computing, Federal University of Ouro Preto, Brazil*

Abstract

Decision support systems and automated planning have become essential for the effective management of seaport container terminals. Due to ever-increasing levels of maritime traffic and freight transportation, terminals are expected to facilitate their intra-yard operations while respecting tighter vessel schedules. Employed models must therefore be capable of providing berth allocation decisions which are not only efficient, but also physically realizable. The present paper addresses the continuous Berth Allocation problem with Specific Quay Crane Assignments, with the aim being to minimize container transshipment distances within the terminal yard. An integrated mathematical formulation is presented which considers various real-world properties such as limited crane ranges and quay discontinuity. Based on this formulation, we also introduce a fast local search-based heuristic, capable of accommodating both fixed and flexible departure time settings. The proposed heuristic is evaluated upon a set of instances which cover a wide range of diverse layout specifications and the solutions are compared against those obtained by a state-of-the-art commercial solver. Experimental results confirm the effectiveness of the heuristic and demonstrate how significantly improved schedules can be obtained when flexible departures are permitted.

Acknowledgments

This research was partly supported by Data-driven logistics (FWO-S007318N). Editorial consultation provided by Luke Connolly (KU Leuven).

CONTACT: EMMANOUIL THANOS, Gebroeders De Smetstraat 1, 9000 Gent, Belgium, emmanouil.thanos@kuleuven.be

The tactical berth allocation problem with time-variant specific quay crane assignments

Abstract

Decision support systems and automated planning have become essential for the effective management of seaport container terminals. Due to ever-increasing levels of maritime traffic and freight transportation, terminals are expected to facilitate their intra-yard operations while respecting tighter vessel schedules. Employed models must therefore be capable of providing berth allocation decisions which are not only efficient, but also physically realizable. The present paper addresses the continuous Berth Allocation problem with Specific Quay Crane Assignments, with the aim being to minimize container transshipment distances within the terminal yard. An integrated mathematical formulation is presented which considers various real-world properties such as limited crane ranges and quay discontinuity. Based on this formulation, we also introduce a fast local search-based heuristic, capable of accommodating both fixed and flexible departure time settings. The proposed heuristic is evaluated upon a set of instances which cover a wide range of diverse layout specifications and the solutions are compared against those obtained by a state-of-the-art commercial solver. Experimental results confirm the effectiveness of the heuristic and demonstrate how significantly improved schedules can be obtained when flexible departures are permitted.

Keywords: Berth allocation, Quay crane assignment, Container transshipment distances, Flexible departure times

1. Introduction

Container handling operations constitute a critical component of maritime freight transportation. Efficient berthing decisions can help reduce container transshipments and yard moves, thereby improving in-yard traffic and rendering terminals highly competitive. However, increased vessel

traffic, restricted time windows and service resource limitations impose great difficulties in providing high-quality berth allocation schedules.

When it comes to designing efficient berthing schedules, a significant limitation faced by terminal managers concerns how best to deploy their quay cranes (QCs). QCs handle the container loading and unloading operations of the vessels. The allocation of mooring positions must therefore allow for the assignment of suitable sets of QCs and for sufficient service times for each vessel, such that their container handling requirements are fully met during their stay.

Motivated by the rapid development of management techniques for port logistics, seaside operations and maritime transportation (Bierwirth and Meisel, 2015), this paper investigates the integration of the continuous Berth Allocation (BAP) and specific Quay Crane Assignment (QCAP) problems at a tactical level, with the aim being to minimize container transfer distances within the terminal yard. This objective is directly related to various operational features of a transshipment terminal - yard congestion, driving costs and container relocation requirements - and thus reflects the terminal's total costs in a highly realistic manner.

The combined Berth Allocation and Quay Crane Assignment Problem (BACAP) must assign both a berthing position and a number of cranes to each vessel. When specific cranes must also be assigned to each vessel, then the resulting problem is referred to as the BACASP. Solution quality is quantified using vessel handling or departure times, which are frequently impacted by crane assignment decisions. The deviation of assigned positions from desired berthing positions may also be considered when terminal management objectives are involved.

Numerous studies have addressed a wide range of variants of the aforementioned problem components. Research focusing on integrated solution approaches and container transfer-related objectives will be reviewed in Section 2. For general reviews concerning independent developments and classifications for both the BAP and QCAP, readers are referred to Bierwirth and Meisel (2010, 2015).

Given a specific quay layout and a set of scheduled weekly vessel arrivals along with the required information concerning their container loads and transshipments, the Tactical Berth Allocation and Specific Quay Crane Assignment Problem (TBACASP) studied throughout this paper can be characterized by way of three primary decisions:

- (i) assigning each vessel to a berthing position at the quay,
- (ii) assigning to each vessel a set of cranes compatible with its berthing position,
- (iii) determining the time interval during which each assigned crane will be serving its assigned vessel.

Various mooring and operational restrictions are integrated into the developed model, such as crane crossing constraints and range limitations. The continuous quay layout considered implies that vessels may berth at any position along the quay and offers more flexibility than discrete quay assignments. QC assignments are time-variant, meaning that cranes can be assigned to serve vessels for either their entire stay or a fraction of that stay. A detailed setting description and a mathematical formulation of the TBACASP are provided in Section 3.

Although transshipment-related objectives typically consider lengthy horizons, terminal managers must frequently respond to unexpected schedule modifications which require quick decisions. Solutions are thus required within short computation times. The proposed heuristic, detailed throughout Section 4, works by considering both fixed and flexible departure time settings. It employs a remove & reinsert strategy utilizing spatio-temporal criteria, which have demonstrated significant potential for improving berthing schedules (Li et al., 2015). Feasible solutions are evaluated in terms of their total container transfer distance, which is a result of the transshipment flows between vessels' mooring positions.

Experimental results are presented in Section 5, with the proposed approach evaluated over a set of benchmark instances which covers a wide range of quay layouts, vessel arrivals, QC features and container loads. Heuristic solutions for small-sized instances are compared against those obtained by a commercial solver, while the added value of considering flexible departure times is analysed with respect to different instance features. The paper concludes with Section 6, which documents the most significant contributions while also outlining future research directions.

2. Literature review

The integration of the BAP and QCAP was first introduced by Park and Kim (2005), who utilized a Lagrangean relaxation-based heuristic to determine the berthing position, berthing time and number of cranes assigned to

each vessel. A dynamic programming procedure was subsequently employed for the specific crane-to-vessel assignments, thereby providing a BACASP solution at a later stage.

The BAP has been shown to be an NP-hard problem (Lim, 1998; Hansen and Oguz, 2003). The majority of studies which address it and its variants employ heuristic approaches. Table 2 summarizes the problem and method features considered by each of the various BACAP and BACASP studies discussed throughout this section.

2.1. Time-invariant versions and accommodating real-world limitations

A considerable number of studies have addressed the time-invariant version of both the BACAP and BACASP, where the cranes assigned to each vessel remain the same throughout its stay at the terminal. Türkoğullari et al. (2014) provide a new position assignment formulation and propose an efficient cutting plane algorithm for solving large BACASP instances to optimality. Correcher and Alvarez-Valdes (2017) address the continuous, time-invariant version of both the BACAP and BACASP with a Genetic Algorithm (GA), whereas Correcher et al. (2019) developed a Mixed Integer Programming (MIP) model to solve both variants in an exact manner.

Zhang et al. (2010), Raa et al. (2011) and Xiaolong et al. (2015) all proposed MIP formulations which take into account a limited coverage range for each crane and demonstrate how such restrictions may be of significant practical relevance for integrated planning approaches. Zhang et al. (2010) further assumed that the number of cranes assigned to each vessel may only be changed once prior to departure. Yang et al. (2012) introduced crane non-crossing constraints for the BACAP and analysed the interactions between the BAP and QCAP. Finally, Ma et al. (2019) considered the BACAP with multiple continuous quays to address discontinuity issues in both berth and storage yard layouts, showing how such limitations can significantly affect the resulting container transfer distances.

2.2. Time-variant BACAP and BACASP

In the time-variant BACAP and BACASP, the number of assigned cranes may vary over time, allowing for the more efficient utilization of cranes. Meisel and Bierwirth (2009) developed a MIP formulation for the continuous BACAP with time-variant crane assignments and proposed two effective metaheuristic approaches. Meisel and Bierwirth (2013) extended their model to consider specific crane assignments in a functional integration approach,

Reference	Spatial	Temp	QCAP	OF	Method
Park and Kim (2005)	C	TV, TI	S	Dev, Lat, QC	H
Imai et al. (2008)	D	TV	S	ST	H
Meisel and Bierwirth (2009)	C	TV	N	ST	H
Zhang et al. (2010)	C	TV	S	Dev, Lat	H
Giallombardo et al. (2010)	D	TV	G	QCP, HC	H (E)
Raa et al. (2011)	C	TV	N	Dev, Lat	H
Yang et al. (2012)	C	TI	N	ST, QC	H
Meisel and Bierwirth (2013)	C	TV	S	QCP	H (E)
Vacca et al. (2013)	D	TV	G	QCP, HC	E
Lalla-Ruiz et al. (2014)	D	TV	G	QC, HC	H (E)
Rodriguez-Molins et al. (2014)	C	TV	N	ST	H
Türkoğullari et al. (2014)	C	TI	S	Dev, Lat	E
Xiaolong et al. (2015)	D	TV	N	ST, Lat	H (E)
Hu (2015)	C	TV	N	QC, Lat	H
Karam and Eltawil (2016)	C	TV	S	Dev, Lat	H
Liu et al. (2016)	C	TV	G	HC, Lat	H (E)
Türkoğullari et al. (2016)	C	TV	S	Dev, Lat	E
Correcher and Alvarez-Valdes (2017)	C	TI	S	Dev, Lat	H (E)
Xiang et al. (2018)	D	TV	S	Dev, ST	H (E)
Agra and Oliveira (2018)	C	TV	S	ST	H (E)
Correcher et al. (2019)	C	TI	S	Dev, Lat	E
Ma et al. (2019)	C	TI	G	HC, Lat	E
This paper	C	TV	S	HC	H (E)

Table 1: Integrated BACAP attributes.

BAP spatial features (**Spatial**): [C: Continuous, D: Discrete];

QCAP temporal features (**Temp**): [TV: Time-variant, TI: Time-invariant];

QCAP integration features (**QCAP**): [N: number, G: Groups/Profiles, S: Specific];

Objective function features (**OF**): [HC: transshipment-related housekeeping cost, ST: service/handling times, QCP: QC Preference, QC: QC productivity, Dev: deviation from the desired berth section, Lat: the cost of berthing/departing later than the estimated time];

Developed methods (**Method**): [H: Heuristic, E: Exact, (E): MIP solver utilizing a mathematical formulation to evaluate or complement a proposed heuristic]

which also includes the Quay Crane Scheduling Problem. Rodriguez-Molins et al. (2014) showed that their GRASP heuristic outperforms traditional heuristics for BACAP. Hu (2015) addressed the BACAP as a composition of separate submodels and employed different heuristics to solve each submodel. Karam and Eltawil (2016) considered the functional integration of the BAP and Specific QCAP, which are iteratively solved until the complete solution cannot be further improved. More recently, Agra and Oliveira (2018) presented a mathematical model for the integration of the two problems and proposed heuristic methods to solve the model sequentially.

When berth allocation and time-variant specific crane assignments are addressed simultaneously, the underlying settings typically consider either a discrete quay layout or a limited number of candidate berthing positions. Imai et al. (2008) proposed a GA for the discrete BAP with time-variant assignment of specific QCs subject to detailed QC movement constraints. Xiang et al. (2018) proposed a reactive strategy to address various types of disruptions present in the same variant of the integrated problem. Türkoğullari et al. (2016) addressed the deep integration of BAP and specific QCAP and were able to solve instances containing up to 60 vessels to optimality. The quay setting considered was continuous but highly discretized, with 24 berth slots and 12 cranes. Their approach employs a MIP model for solving the BACAP and acquires the optimum crane schedule with a minimum cost non-crossing flow formulation.

2.3. Tactical Berth Allocation

The tactical version of the integrated problem (TBAP) takes into account the terminal's container yard and assumes cyclic arrival schedules. The objective involves minimizing the housekeeping costs generated by transshipment flows, typically for a weekly time horizon.

Giallombardo et al. (2010) were the first to model the TBAP for discrete quay layouts. They employed the concept of QC profiles to encode the number of QCs per working shift along with real-world limitations. They propose two MIP formulations and a matheuristic. Vacca et al. (2013) implemented a branch-and-price algorithm to produce optimal integer solutions for instances with up to 20 vessels. Lalla-Ruiz et al. (2014) propose a GA for solving larger instances which include realistic scenarios based on the model by Giallombardo et al. (2010). More recently, Liu et al. (2016) considered transshipment-related quadratic yard costs within the continuous BACAP. They designed a bi-objective model to simultaneously optimize berth and

yard allocation decisions and proposed three heuristic approaches for solving the problem. Their methods, however, also utilize QC profiles instead of individual crane assignments and thus remain theoretical. In other words, they do not ensure that the solutions produced are physically realizable.

Academic studies concerning the BAP very rarely consider strict time windows for serving vessels. An extensive review of existing work has revealed Hendriks et al. (2013) to be the only academics to consider fixed handling times for the BAP with yard planning objectives. However, with vessel schedule reliability becoming more and more important for the container shipping industry (Sea-Intelligence Maritime Analysis, 2018), the integration of crane assignments as a restricted resource as a means of respecting fixed handling times seems a necessary direction for tactical berth allocation planning. Regarding transshipment flows, the quantitative comparison between fixed and flexible settings may also prove useful to terminal managers, providing an indication of the potential profit/loss incurred at an operational level by (dis)allowing this flexibility.

Finally, with respect to specific QC assignments, as well as crane coverage limitations, there has been insufficient research regarding the impact of combining these features with transshipment-related quadratic yard costs. Given their practical relevance for integrated scheduling approaches, addressing specific crane assignments, limited crane ranges and quay discontinuity issues has significant potential for improving the accuracy and the reliability of berth allocation decisions at a tactical level.

3. The TBACASP

This section presents the general setting and assumptions considered for the Tactical Berth Allocation Problem with Specific Quay Crane Assignments. A MIP formulation is introduced for the fixed-departure variant of the problem and a set of valid inequalities are employed to improve linear programming relaxation. Finally, the base problem description is extended to accommodate a flexible-departure policy for the vessels.

3.1. Problem description

The TBACASP considers a port terminal quay along with its container yard, as shown in Figure 1. The quay is possibly separated into distinct parts, in which case it is assumed that each part corresponds to a different continuous berthing zone. The quay illustrated in Figure 1, for example,

consists of two berthing zones of equal length (Zones 1 and 2). An ordered set of bollards B located along the quay define the positions where vessels may moor. Each bollard $k \in B$ defines a distinct berthing position, meaning that the moored vessel's leftmost part will coincide with k . A fixed distance is assumed between two consecutive bollards of the same zone, which also defines the minimum considered length unit u . The driving distance between any two bollards is considered as known and is equal to the physical distance between them, with an additional distance d^p if they belong to different berth zones.

Figure 1: Setting configuration.

Vertical axis = time and horizontal axis = space. The quay layout with two berthing zones (Zones 1 and 2), set of bollards (B) and set of QCs with their base positions and ranges are demonstrated. Vessels (V) are represented by rectangles.

The pair of crane assignments denoted by red lines in Zone 1 is infeasible due to cranes' overtaking.

The distance between BAs assigned to Vessel 2 and Vessel 4 is denoted by the blue lines. The length fraction $fr_{i,k,c}$ of Vessel 1 covered by QC 7 given the vessel's assigned BA is shown in green.

The yellow rectangle in Zone 2 visualises Vessel 9's arrival for the consecutive calendar week, thus indicating how the container flow $n_{1,9}$ between Vessel 1 and Vessel 9 will occur.

A set V of vessels carrying containers are scheduled to arrive at the terminal on a weekly basis. Each vessel $i \in V$ has scheduled arrival and departure times $start_i$ and end_i , respectively, and must be assigned to a berthing position in the quay, where it remains moored for its entire stay. The assignment

of a vessel $i \in V$ to a bollard $k \in B$ is indicated by a **Bollard Assignment** (BA) (i, k) . Each BA is associated with the ordered set of bollards occupied by the corresponding vessel's length. Vessels with overlapping stays must not only moor at separate bollard sets, but must also ensure they respect a minimum safety distance d^s . The distance $D_{i,k,j,\ell}$ between two BAs (i, k) and (j, ℓ) is equal to the driving distance between their center bollards. The center bollard of a BA is the bollard located closest to the center of its range. If two bollards are equidistant from the center, then the bollard to the left is chosen. The blue lines in Zone 1 from Figure 1 denote the distance $D_{2,k,4,\ell}$ between BAs associated with Vessel 2 and Vessel 4.

During a vessel's stay its export containers must be discharged at its mooring position in order to be transhipped to other vessels. Likewise, the vessel's import containers must also be transferred and loaded onto it. QCs load and unload the vessels, while container transshipments are carried out by special vehicles within the yard. The total number of containers which must be processed by QCs for a vessel i is denoted as n_i . The set of ordered vessel pairs $(i, j), i < j$, which are involved in container transshipments is denoted as S and each vessel pair $(i, j) \in S$ is characterized by the number $n_{i,j} = n_{j,i}$ of containers to be transhipped between i and j . For the purposes of this work it is assumed that the cyclic arrival schedule is repeated every week. In this manner, and without compromising on generality, container transshipment flows concern two consecutive weeks for each pair of vessels and are independent of the vessels' arrival sequence. Figure 1 shows an illustrative example of such a case, which indicates the container flow $n_{1,9}$ between Vessel 1 and Vessel 9, moored at Zone 2.

QCs C are located uniformly along the quay. They can operate within a limited range of fixed length q_c , as shown by the horizontal black lines in Figure 1. They are further characterized by their container handling productivity p_c , which indicates the maximum number of containers they can process per hour. Each crane may serve only one vessel at any time and is not allowed to leave and return to the same vessel. Each vessel i is characterized by a maximum number c_i^{max} of cranes which may be assigned to it during its stay, where c_i^{max} depends on vessel i 's length. Finally, cranes are ordered with respect to their base position and they are not allowed to pass over one another. The red lines in Zone 1 from Figure 1 denote one such infeasible pair of crane assignments.

Each BA (i, k) is associated with a set of compatible cranes. The length fraction of (i, k) reachable by a crane $c \in C$ is denoted as $fr_{i,k,c} \in [0, 1]$, as

illustrated for Vessel 1 in Figure 1. The set of compatible cranes of a BA then depends on a parameter $rcc \in (0, 1]$ which defines the minimum required crane coverage a single crane must meet and contains all those cranes which cover a fraction $fr_{i,k,c} \geq rcc$.

A crane assignment $r = (i, k, c, I^-, I^+)$ indicates the operation of a crane $c \in C$ with respect to a vessel $i \in V$ assigned to BA (i, k) and is associated with the operation's start time $I^- = start_{i,k,c}$ and end time $I^+ = end_{i,k,c}$, respectively. For r to be considered as a candidate crane assignment, the respective interval $[I^-, I^+]$ must overlap with a minimum fraction of i 's stay duration which depends on $fr_{i,k,c}$ and an additional parameter $mcaf \in (0, 1]$, defined as $\frac{I^+ - I^-}{end_i - start_i} \geq \frac{mcaf}{fr_{i,k,c}}$. Parameters rcc and $mcaf$ enable to avoid crane assignments which are inefficient due to their restricted container processing capabilities.

All import and export containers are assumed to be evenly distributed across a vessel's length. Therefore, the number of i 's containers processed by c with r is bounded by $p_{i,k,c}^{max} = fr_{i,k,c} \cdot n_i$ and is equal to $z_r = \min(p_{i,k,c}^{max}, (end_{i,k,c} - start_{i,k,c}) \cdot p_c)$. A vessel i 's (un)loading requirements are fully served by its set of associated crane assignments when (i) its total number of containers are processed and (ii) every part of i is reachable by at least one of the cranes assigned to it.

A feasible berthing schedule consists of a valid set A^S of BAs which assign each vessel to a berthing position and allow for the construction of a valid set of crane assignments, such that all vessel container (un)loading requirements are fully served during their stay. The distance between two moored vessels i, j under a valid set A^S of BAs is equal to the distance between their assigned BAs and is denoted as $D_{i,j}^{A^S} = D_{i,k,j,l}$, where $(i, k), (j, l) \in A^S$.

Given a specific quay layout and a vessel arrival schedule, the TBACASP consists of finding a feasible berthing schedule A^S which minimizes the total container transfer distance $TCTD$, calculated using Equation (1) as follows:

$$TCTD = \sum_{(i,j) \in S} (n_{i,j} \cdot D_{i,j}^{A^S}) \quad (1)$$

3.2. Mathematical formulation

Following the aforementioned problem description, this section introduces a MIP formulation for the TBACASP assuming fixed handling times for the vessels:

Input sets

V : set of vessels

B : set of bollards

V_k : set of vessels compatible with bollard $k \in B$, $V_k \subseteq V$

B_i : set of bollards compatible with vessel $i \in V$, $B_i \subset B$

A : set of possible BAs (i, k) , where $i \in V$ and $k \in B_i$

A_i : set of BAs compatible with vessel $i \in V$

$\tilde{A}_{i,k}$: set of prohibited BAs (j, ℓ) if vessel $i \in V$ is assigned to bollard $k \in B_i$, where $j \in V$ and $\ell \in B_j$

C : set of cranes

$C_{i,k}$: set of cranes compatible with vessel $i \in V$ assigned to bollard $k \in B$, $C_{i,k} \subseteq C$

$K_{i,k}^C$: ordered set containing minimal crane sets which cover the total length of vessel $i \in V$ when assigned to bollard $k \in B$: $K_{i,k}^C = \{C_1, \dots, C_N\}$ with each set $C_h \subseteq C_{i,k}$

S : set of vessel pairs (i, j) involved in container transshipments, where $i, j \in V$ and $i < j$

T : set of timeslots, defined by intervals between every pair of consecutive arrival or departure times of vessels in V , sorted in ascending order, as shown in Figure 2.

T_i : set of timeslots during which vessel $i \in V$ is berthed. $T_i = \{a_i, \dots, d_i\}$ where a_i starts at $start_i$ and d_i ends at end_i , respectively. An illustrative example is shown for Vessel 2 in Figure 2, where $a_2 = t_3$, $d_2 = t_4$ and $T_2 = \{t_3, t_4\}$.

Properties of vessels, bollards, cranes and timeslots

$start_i$: arrival time of vessel $i \in V$, in hours

end_i : departure time of vessel $i \in V$, in hours

Figure 2: Timeslots representation.

Considered timeslots are defined by the intervals between the red lines, which correspond to the arrival and departure times of the vessels.

Vessel 2 is berthed during timeslots t_3 and t_4 and thus $T_2 = \{t_3, t_4\}$.

Vessel 4 arrives on Sunday and departs on Monday. The yellow, transparent rectangle visualises its stay for the previous calendar week. Therefore, its departure time defines the first timeslot t_1 . Timeslots t_1 and t_9 correspond to the same period of the weekly schedule and thus $t_9 = t_1^{cyclic}$.

- a_i : arrival timeslot of vessel $i \in V$, where $a_i \in T$
- d_i : departure timeslot of vessel $i \in V$, where $d_i \in T$
- c_i^{max} : maximum number of cranes that can be assigned to vessel $i \in V$
- $n_{i,j}$: total number of containers that must be transported between vessels i and j , where $(i,j) \in S$ or $(j,i) \in S$
- n_i : total number of containers which must be processed for vessel $i \in V$, such that $\sum_{j \in V} n_{i,j} = n_i$
- $\rho_{i,k,j,\ell}$: total penalty for assigning vessels $i \in V$ and $j \in V$ to bollards $k \in B_i$ and $\ell \in B_j$, equal to their potential contribution to TCTD:
 $\rho_{i,k,j,\ell} = D_{i,k,j,\ell} \cdot n_{i,j}$
- p_c : container handling productivity of crane $c \in C$, meaning the number of containers it can process per hour
- $p_{i,k,c}^{max}$: maximum number of containers which can be transported by crane $c \in C_{i,k}$ if it is assigned to vessel $i \in V$ at bollard $k \in B_i$, such that
 $p_{i,k,c}^{max} = fr_{i,k,c} \cdot n_i$
- e_t : duration of timeslot $t \in T$ in hours
- $|t|$: order of timeslot t , where $|t| = 0$ corresponds to the earliest timeslot.
- t_v^{cyclic} : timeslot occurring after the end of the calendar week and corresponds to the same period as timeslot t' , which occurs at the beginning of the week. It is utilized to accurately model crane-related time overlaps at the marginal cases of the cyclic schedule, for those vessels which arrive before and depart after Sunday 23.59. An illustrative example is shown in Figure 2, where Vessel 4 arrives on Sunday and departs on Monday, thus $t_9 = t_{t_1}^{cyclic}$.
- $mcaf_{i,k,c}$: minimum duration required for operation of crane c to BA (i,k) , defined as $mcaf_{i,k,c} = \frac{mcaf}{fr_{i,k,c}} \cdot (end_i - start_i)$.

Variables:

$x_{i,k}$: binary variable that assumes a value of 1 if vessel $i \in V$ is assigned to BA $(i, k) \in A$ and 0 otherwise

$\gamma_{i,j}$: penalty incurred by vessel pair $(i, j) \in S$, equal to its contribution to TCTD

$y_{i,k,c,t}$: binary variable that assumes a value of 1 if vessel $i \in V$ is assigned to BA $(i, k) \in A$ serviced by crane $c \in C_{i,k}$ at timeslot $t \in T_i$ and 0 otherwise

$z_{i,c}$: number of containers (un)loaded to/from vessel $i \in V$ by crane $c \in C$

$w_{i,c}$: auxiliary binary variable that assumes a value of 1 if crane $c \in C$ serves vessel $i \in V$ at any time and 0 otherwise

$u_{i,k,h}$: auxiliary binary variable that assumes a value of 1 if every crane $c \in C_h$ serves vessel $i \in V$ which is assigned to BA $(i, k) \in A$, with minimal set $C_h \in K_{i,k}^C$, and 0 otherwise

The formulation for the TBACASP is then as follows:

$$\min \sum_{(i,j) \in S} \gamma_{i,j} \quad (2)$$

s.t.

$$\sum_{k \in B_i} x_{i,k} = 1 \quad \forall i \in V \quad (3)$$

$$x_{i,k} + x_{j,\ell} \leq 1 \quad \forall (i, k) \in A, (j, \ell) \in \tilde{A}_{i,k} \quad (4)$$

$$\sum_{\substack{(i,k) \in A: \\ c \in C_{i,k}}} (y_{i,k,c,t} + y_{i,k,c,t}^{cyclic}) \leq 1 \quad \forall c \in C, t \in T \quad (5)$$

$$y_{i,k,c,t} \leq x_{i,k} \quad \forall (i, k) \in A, c \in C_{i,k}, t \in T_i \quad (6)$$

$$z_{i,c} \leq \sum_{k \in B_i} \sum_{t \in T_i} p_{c,t} y_{i,k,c,t} \quad \forall i \in V, c \in C \quad (7)$$

$$z_{i,c} \leq \sum_{k \in B_i} p_{i,k,c}^{max} x_{i,k} \quad \forall i \in V, c \in C \quad (8)$$

$$\sum_{c \in C_i} z_{i,c} = n_i \quad \forall i \in V, c \in C \quad (9)$$

$$\sum_{i \in V} \sum_{\substack{k' \in B_i: c' \in C_{i,k'}: \\ k' > k \quad c' < c}} y_{i,k',c',t} + M_1 \sum_{i \in V} (y_{i,k,c,t} + y_{i,k,c,t_t^{cyclic}}) \leq M_1 \quad \forall c \in C, \forall k \in B, \forall t \in T \quad (10)$$

$$y_{i,k,c,t} \leq y_{i,k,c,t-1} + 1 - \frac{1}{|t|-1} \sum_{t' < t-1} y_{i,k,c,t'} \quad \forall i \in V, c \in C, k \in B_i, t \in T_i / \{0, 1\} \quad (11)$$

$$w_{i,c} \geq \sum_{k \in B_i} y_{i,k,c,t} \quad \forall i \in V, c \in C_{i,k}, t \in T_i \quad (12)$$

$$w_{i,c} \leq \sum_{k \in B_i} \sum_{t \in T_i} y_{i,k,c,t} \quad \forall i \in V, c \in C \quad (13)$$

$$\sum_{c \in C_i} w_{i,c} \leq c_i^{max} \quad \forall (i, k) \in A, t \in T_i \quad (14)$$

$$u_{i,k,h} \leq \frac{1}{|C_h|} \cdot \sum_{c \in C_h} w_{i,c} \quad \forall (i, k) \in A, C_h \in K_{i,k}^C \quad (15)$$

$$x_{i,k} \leq \sum_{C_h \in K_{i,k}^C} u_{i,k,h} \quad \forall (i, k) \in A, C_h \in K_{i,k}^C \quad (16)$$

$$\sum_{t \in T_i} e_t y_{i,k,c,t} + M_2(2 - w_{i,c} - x_{i,k}) \geq mcaf_{i,k,c} \quad \forall (i, k) \in A, c \in C_{i,k} \quad (17)$$

$$\gamma_{i,j} \geq \sum_{\ell \in B_j} \rho_{i,k,j,\ell} \cdot (x_{i,k} + x_{j,\ell} - 1) \quad \forall (i, j) \in S, k \in B_i \quad (18)$$

$$x_{i,k} \in \{0, 1\} \quad \forall (i, k) \in A \quad (19)$$

$$y_{i,k,c,t} \in \{0, 1\} \quad \forall (i, k) \in A, c \in C_{i,k}, t \in T_i \quad (20)$$

$$w_{i,c} \in \{0, 1\} \quad \forall i \in V, c \in C \quad (21)$$

$$u_{i,k,h} \in \{0, 1\} \quad \forall (i, k) \in A, C_h \in K_{i,k}^C \quad (22)$$

Objective Function (2) minimizes the sum of transfer distances for the container flows associated with each pair of vessels, which is equal to the total container transfer distance TCTD, given by Equation (1). Constraints (3) ensure each vessel is assigned to exactly one bollard, while Constraints (4) prohibit overlapping bollard assignments. Constraints (5) ensure that each crane is assigned to at most one BA per timeslot. Constraints (6) link

decision variables x and y , thereby ensuring that cranes are assigned to existing BAs. Constraints (7) and (8) define the upper bound concerning the number of containers in a vessel which are served by each of its assigned cranes, while Constraints (9) ensure they sum to n_i . Constraints (10) prohibit crane overtaking, where M_1 is a sufficiently large number. Constraints (11) ensure crane assignment continuity, preventing cranes from leaving and subsequently returning to a vessel. Constraints (12) and (13) link w auxiliary variables with y variables. Constraints (14) ensure that at most c_i^{max} cranes are assigned to a vessel during its stay. Constraints (15) and (16) define u auxiliary variables, ensuring that the total length of a moored vessel is covered by its assigned cranes. Constraints (17) are the minimum crane assignment duration constraints, where M_2 is a sufficiently large number. Constraints (18) define γ auxiliary variables which enable to determine the TCTD. Finally, Constraints (19)-(22) ensure the integrality of the solutions.

Sufficiently large values for M_1 and M_2 in Constraints (10) and (17) are respectively:

$$M_1 = \sum_{i \in V} \sum_{\substack{k' \in B_i: \\ k' > k}} |c' \in C_{i,k'} : c' < c| \quad (23)$$

$$M_2 = (end_i - start_i) \cdot mcaf/rcc \quad (24)$$

3.3. Valid inequalities

This section introduces a set of valid inequalities for the presented mathematical formulation.

For each timeslot t , an auxiliary vessel i_t^{aux} is utilized to generate groups of incompatible BAs. The stay of this vessel coincides with the corresponding timeslot and its length is equal to safety distance d^s . For each bollard ℓ , the set $\tilde{A}_{t,\ell}$ of all BAs which overlap with the auxiliary BA (i_t^{aux}, ℓ) without considering an additional safety distance is then generated. Each of these sets contains BAs which are pairwise incompatible. Constraints (4) may now be formulated as

$$\sum_{(i,k) \in \tilde{A}_{t,k}} x_{i,k} \leq 1 \quad \forall \ell \in B, t \in T \quad (25)$$

In addition, non-overlapping Constraints (4) imply that vessels with overlapping stays will have a minimum distance from each other. Inequalities related to separation distances have proven effective for other BACAP variants

(Correcher et al., 2019). Given the direct connection of the objective function with separation distances, such inequalities show significant potential for enhancing MIP models for the tactical variants. The following inequalities are therefore also added in order to strengthen the proposed formulation.

For each pair i, j of vessels with overlapping stays, the corresponding contribution to the TCTD is lower bounded by the factor $n_{i,j} \cdot D_{i,j}^{min}$, where $D_{i,j}^{min} = L_i/2 + L_j/2 + d^s$ is the minimum distance between any compatible pair of their BAs. The following constraints may thus be added:

$$\gamma_{i,j} \geq n_{i,j} \cdot D_{i,j}^{min} \quad \forall i, j \in S : (start_i < end_j, start_j < end_i) \quad (26)$$

Finally, the minimum distance between vessels with overlapping stays also lower bounds the contribution to the TCTD of their pairs with other, non-overlapping vessels. In the example shown in Figure 1, Vessel 3's combined TCTD contribution ($\gamma_{3,8} + \gamma_{3,10}$) with respect to Vessel 8 and Vessel 10 is always greater than $D_{8,10}^{min} \cdot \min(n_{3,8}, n_{3,10})$.

This may be generalized for groups of vessels with overlapping stays. Let V^t be the set of all vessels j with stays which include timeslot t . Then let $L_{|j|}$ be the vector containing their lengths, sorted in descending order, and $n_{i,|j|}$ be the vector containing the transshipment values between a vessel $i \in V \setminus V^t$ and all $j \in V^t$, sorted in ascending order. The following constraints may then be added:

$$\sum_{j \in V^t} \gamma_{i,j} \geq \sum_{\substack{j \in V^t: \\ |j| > 1}} (n_{i,|j|} \cdot L_{j,V^t}^{rem}) \quad \forall t \in T, i \in V \setminus V^t \quad (27)$$

where $L_{j,V^t}^{rem} = \sum_{\substack{j' \in V^t: \\ |j'| \geq |j|}} L_{|j'|} + (|V^t| - |j|) \cdot d^s - \frac{L_{|j|} + L_{|j+1|}}{2}$ and indicates

the remaining vessel length sum, obtained by removing the length of the currently longest vessel, after multiplying with the currently smallest $n_{i,j}$, each time from the total vessel length sum. Please note that for settings with more than one berthing zone, the additional distance d^p must also be considered. Therefore, when L_{j,V^t}^{rem} exceeds the length of a single zone, it is replaced by the term $(L_{j,V^t}^{rem} + d^p)$ in Constraints (27).

3.4. Flexible departure setting

Due to the many arrangements which are related to the vessels' strict shipping schedules, terminals are typically obliged to host arriving vessels

for the entirety of their scheduled stay. However, an earlier departure can often be desired provided it raises no conflicts with other terminal obligations. The terminal may allocate the vacated berthing space to other vessels while the departing vessel has more freedom to efficiently plan the voyage to its next destination.

Given these potential benefits, we assume a problem variant where vessels depart as soon as their containers are processed by their assigned cranes. This adaptation significantly alters the problem setting, rendering it impossible to define incompatible BAs in advance and thus prevents us from using the proposed MIP formulation.

It is now possible for two vessels $i, j \in V$ with $start_i < start_j < end_i$, whose scheduled stays initially overlap in time, to be assigned to overlapping positions. This is only permitted if $end_{r_i^{LATEST}} < start_j$, where r_i^{LATEST} and $end_{r_i^{LATEST}}$ denote the last crane assignment serving i and the corresponding operation's end time, respectively. Every other constraint remains unaltered, including the one related to the minimum crane assignment duration.

4. Heuristic

Thanos et al. (2019) introduced a heuristic to provide real-time decision support for the TBACASP. They focused on tuning the algorithm and the effect of incorporating specific crane assignment decisions with respect to its performance. This paper builds upon Thanos et al. (2019)'s heuristic which is generalized to consider a flexible departure time setting. For reasons of convenience, the heuristic components used in that work are also presented following the formulation of Section 3.

Given a quay layout and a set of arrivals, the proposed approach first applies a preprocessing step which involves generating all valid BAs A_i for every vessel $i \in V$. For each BA (i, k) , the set $C_{i,k}$ of compatible cranes is determined. The search space of feasible berthing schedules is subsequently explored and evaluated within a Simulated Annealing (SA) metaheuristic framework (Kirkpatrick et al., 1983) which works on a direct solution representation.

4.1. Solution representation

Assuming all vessels in V are sorted by ascending order of arrival time, a solution $\mathcal{S}$ is defined by:

- a vector A^S of length $|V|$, which at each position A_i^S contains the BA $(i, k) \in A_i$ assigned to i . When i is unassigned, A_i^S is empty.
- a vector $R^{S,i}$ for each $i \in V$, which contains the crane assignments (i, k, c, I^-, I^+) associated with i . When i is unassigned, $R^{S,i}$ is empty.
- a vector V_U^S which contains any unassigned vessels.

4.2. Remove & Reinsert strategy

At each step of the search the algorithm applies a Remove & Reinsert (R&R) strategy to modify the current solution.

Remove step. A set A_{rem} of BAs are first removed from current A^S by means of one of the two following neighbourhood operators:

- RANDOM-REM(n): remove n random BAs from the current solution. Integer n is randomly selected in $[1, maxNumRem]$, where $maxNumRem$ is a parameter defining the maximum number of removed arrivals.
- SPATIO-TEMPORAL-REM(n): randomly select two vessels i, j which are present in the current solution and which overlap in time, such that $\ell_i \geq \ell_j$. Find all BAs $A_{overlap}$ of the current solution which would overlap with vessel i if it were assigned to j 's starting position, as shown in Figure 3 and including t^{cyclic} timeslots. Then remove BAs $A_{rem} = (A_i^S, A_j^S) \cup A_{overlap}$.

Figure 3: Spatio-temporal neighbourhood operator

Operators RANDOM-REM and SPATIO-TEMPORAL-REM are randomly selected with $(1 - stp)$ and stp probabilities, respectively, where $stp \in (0, 1)$ is a parameter of the algorithm. Removed vessels are added to V_U^S

and their associated crane assignment vectors $R^{S,i}$ are emptied. A best-fit constructive method is afterwards applied in an attempt to reinsert the unassigned vessels into the schedule.

Reinsert step. For each unassigned vessel $i \in V_U^S$, the set A_i is iterated over. For all BAs $(i, k) \in A_i$ which incur no conflicts with the current solution's already-assigned BAs, the TCTD contribution of i 's export container flows under (i, k) is computed. The BA with the lowest TCTD contribution is then selected and the algorithm attempts to assign cranes to it.

The crane assignment is achieved by way of $AssignCranes(i, k)$. This fast constructive procedure iterates over all $c \in C_{i,k}$ in decreasing order of $fr_{i,k,c}$, searching each time for the maximum vacant time interval I of c which is capable of generating a valid crane assignment $r = (i, k, c, I^-, I^+)$ which is compatible with all existing crane assignments. In order to help the algorithm avoid inefficient assignment decisions and randomize the procedure in restricted search spaces, valid assignments are skipped with 1% probability. When a set of crane assignments covers (i, k) or the maximum number of cranes is assigned to i , the procedure stops. If a covering set of crane assignments is found, the corresponding BA (i, k) is assigned to i .

Algorithms 1 and 2 in Appendix A provide the pseudocode to complement the textual descriptions. The pseudocode of SA is also provided in Algorithm 3 for completeness.

For initial solution S_0 a first-fit constructive method is applied to insert the entire set of vessels V . In order to assist towards finding a valid berthing schedule during the first steps of the procedure, a cost penalty is added for each unassigned vessel i included in a solution S . This penalty is proportional to i 's length and duration and is equal to $P^U = \ell_i \cdot (end_i - start_i) \cdot p_U$. The value of multiplier p_U is case-dependent and relies on $|V|$ and the total quay length L .

At every step of the search procedure a neighbour is generated and evaluated in order to determine whether or not it should be accepted, depending on its TCTD and SA's current temperature. Various data-structures are utilized to handle the vectors associated with the solution. Geometry and time intervals are also addressed by special data-structures based on interval trees, which enable a significant speed-up with respect to spatial and temporal overlap checks.

4.3. Flexible departure setting

To incorporate flexible departures into the constructive method, an additional step is performed before rejecting a candidate BA (j, ℓ) which is overlapping with a BA (i, k) of the current solution and for which it further holds that $start_i < start_j < end_i$. The algorithm attempts to revise the departure time end_i by applying the following procedure: end_i is first reduced to $end_{r_i}^{LATEST}$. The container surplus of i is subsequently calculated, which is equal to $C_i^{sur} = \sum_{r \in R^{S,i}} z_r - n_i$. If $C_i^{sur} > 0$ then end_i is (further) reduced by a duration $C_i^{sur} / (p_c \cdot |R^{S,i}|)$, which is the minimum duration required to handle C_i^{sur} containers with $|R^{S,i}|$ cranes.

When end_i is revised all associated crane assignments are updated accordingly and their end times are reduced to end_i , or the earliest time after end_i for which $mcaf$ is respected. They then remain unchanged until (i, k) is possibly removed from the solution. BA (j, ℓ) and all subsequent candidate BAs are checked against the revised end time of i , thereby reducing the possibility of overlaps.

5. Experimental evaluation

This section presents an experimental evaluation of the developed algorithm and the quality of the proposed MIP formulation, along with the proposed inequalities. It also analyses the effect of considering flexible departure times to the TCTD with respect to different instance features.

5.1. Benchmark instances and experimental setup

The set of benchmark instances generated by Thanos et al. (2019) is employed for evaluation. This contains 144 instances which cover a wide range of diverse settings and considers historical data provided by a terminal operator, in addition to real-world characteristics described in other datasets (Park and Kim, 2005; Meisel and Bierwirth, 2009).

The features of all employed instances along with their associated values are presented in Table 2. For each group of small, medium and large-sized instances, partitioned with respect to their quay length, all possible combinations of the corresponding feature values are present in these instances. 2-zone settings contain two berthing zones of length $L/2$ each. For each combination of quay length L and number of vessels $|V|$, the vessel specifications are identical across all sixteen instances. The respective container

transshipments depend on the provisional container load N^P , which indicates the average container load of each vessel. Please note that medium and large quay lengths constrain the instances with the largest vessel sets $|V|$ very tightly for the BAP. Thanos et al. (2019) fully detail the instance-generation procedure.¹

Feature	Small	Medium	Large
Quay length L (m)	1000	2000	4000
Number of berthing zones	1,2	1,2	1,2
Crane coverage range q_c (m)	200,400	600,800	600,800
Number of cranes $ C $	10,20	20,30	40,60
Number of vessels $ V $	10,20	20,25,30	40,50,60
Provisional Container load N^P	800,1000	800,1000	800,1000

Table 2: Instance templates. For each quay length, all possible combinations of the corresponding column are applied, thus resulting in 32 instances for small-sized, 48 instances for medium-sized and 48 instances for large-sized group.

All experiments are conducted on a Dell Precision T3620, Intel Core i7-6700 with 32GB RAM with the algorithm written in Java. The MIP is solved using CPLEX version 12.8.0, configured to use eight threads with a time limit of 5 hours for each instance.

The additional distance d^p for pairs of bollards belonging to different berthing zones in 2-zone settings and the safety distance d^s required between two BAs were set to 500m and 30m, respectively. The container handling productivity p_c was set to 30 containers per hour for all cranes.

Algorithm parameters *maxNumRem* and *stp* were set to 15 and 0.4, respectively, for all instances, based on the tuning suggested by Thanos et al. (2019).

For experiments with small-sized instances, where the MIP formulation is also solved, the maximum value $u = 10m$ was employed for the minimum considered length unit. Setting-related parameters *rcc* and *mcaf* were both set to 0.8, except for the instances noted with an asterisk (*) in the corresponding tables, for which the values were reduced as specified to ensure

¹All instances are available upon request and will be publicly available upon acceptance

feasibility. SA's parameters were tuned to $T_0 = 10^5$ and $cs = 2 \cdot 10^{-5}$, based on the tuning suggested by Thanos et al. (2019).

For experiments with medium and large-sized instances, the minimum considered length unit was set to $u = 2m$. SA's parameters were tuned to $T_0 = 2 \cdot 10^5$ and $cs = 7 \cdot 10^{-5}$, while parameters rcc and $mcaf$ were set to 0.7 and 0.01, respectively, based on the tuning suggested by Thanos et al. (2019).

5.2. Computational results

The first set of experiments compares the performance of CPLEX solving the MIP formulation proposed in Section 3.2 including the valid inequalities of Section 3.3 with a single run of the heuristic (SA) for small-sized instances. The runtime of SA was restricted to 15 minutes. The results for 1-zone and 2-zone settings are presented in Tables 3 and 4, respectively. The tables provide the MIP solver's processing time, lower bound (LB), the cost of the best solution found (Best Int) and respective gap. For the heuristic algorithm, the cost of the best solution found (TCTD), the time it took to find this solution and the respective gap from the LB are shown. In all cases the value of Gap equals to $\frac{Cost-LB}{Cost}$, where $Cost$ is the corresponding solution's cost.

Five 1-zone and five 2-zone instances with $|V| = 10$ were solved to optimality by the MIP solver. For the instances with $|V| = 20$ CPLEX returned gaps of up to 79%. Gaps are generally larger for 2-zone settings, where additional distances incurred by quay discontinuity impose further difficulties in obtaining high-quality LBs. The poor performance of MIP approach for the instances with increased crane range ($q_c = 400$) also highlights the impact of this feature with respect to the problem size.

For instances solved to optimality, the heuristic also found optimum solutions within very short computation times. SA always results in better or equal TCTDs compared to those found by the integer-programming approach, requiring far shorter processing times. Optimality gaps remain large, especially for instances with $|V| = 20$.

To enable a fair comparison between the heuristic and CPLEX, experiments with SA were repeated with 5 hours of runtime for all small-sized instances. These results are reported in the same manner in Tables A.13 and A.14 in Appendix A. CPLEX's results are also included for completeness.

Compared to its performance with 15 minutes of runtime, SA improved the solutions and respective gaps for 7 instances. For the other 25 instances,

Instances (1-zone)				MIP				SA			
V	C	q_c	N_v^P	Time(s)	LB	Best Int	Gap	Time(s)	TCTD	Gap	
10	10	200	800*	1,886.69	750,730.00	750,730	0%	3.00	750,730	0%	
			1000*	828.55	920,510.00	920,510	0%	12.62	920,510	0%	
		400	800	7,842.78	750,730.00	750,730	0%	5.84	750,730	0%	
			1000	9,324.48	918,460.00	918,460	0%	4.44	918,460	0%	
		20	200	800	16,133.36	750,730.00	750,730	0%	6.53	750,730	0%
				1000	18,000.14	597,092.66	918,460	35%	4.62	918,460	35%
			400	800	18,000.03	530,790.83	750,730	29%	6.33	750,730	29%
				1000	18,000.36	568,585.97	924,490	38%	5.02	918,460	38%
	20	10	200	18,000.19	1,042,363.32	1,720,970	39%	85.35	1,716,020	39%	
			1000*	18,000.23	1,239,889.00	2,302,680	46%	101.63	2,266,000	45%	
			400	800	18,000.23	932,815.00	1,951,200	52%	71.76	1,670,430	44%
				1000	18,000.30	1,239,889.00	2,721,800	54%	39.08	2,198,280	44%
		20	200	800	18,000.25	932,815.00	1,807,600	48%	804.22	1,658,670	44%
				1000	18,000.61	1,239,889.00	2,220,250	44%	204.03	2,158,840	43%
			400	800	18,001.22	932,815.00	2,129,100	56%	171.80	1,672,940	44%
				1000	18,001.06	1,239,889.00	2,630,070	53%	832.38	2,170,870	43%

Table 3: Results obtained by CPLEX (TO=5h) and SA (TO=15 min) for small-sized, 1-zone instances using $u = 10$. Parameters ($rcc, mcaf$) are set to (0.8, 0.8) except for instances noted with asterisk (*), for which they were reduced to (0.7, 0.7) in order to ensure feasibility. SA's parameters are set to $T_0 = 10^5$ and $cs = 2 \cdot 10^{-5}$

Instances (2-zone)				MIP				SA		
$ V $	$ C $	q_c	N_v^P	Time(s)	LB	Best Int	Gap	Time(s)	TCTD	Gap
10	10	200	800	117.31	2,188,895.00	2,188,895	0%	5.92	2,188,895	0%
			1000	11.36	2,670,665.00	2,670,665	0%	43.91	2,670,665	0%
		400	800	15,578.19	2,138,645.00	2,138,645	0%	2.69	2,138,645	0%
			1000	18,000.13	1,861,640.23	2,589,665	28%	2.22	2,589,665	28%
	20	200	800	5,513.02	2,138,645.00	2,138,645	0%	0.52	2,138,645	0%
			1000	4,745.77	2,593,655.00	2,593,655	0%	10.42	2,593,655	0%
		400	800	18,000.55	1,639,785.70	2,143,005	23%	1.30	2,138,645	23%
			1000	18,000.31	1,659,878.21	2,589,665	36%	7.89	2,589,665	36%
	20	10	200	18,000.25	1,983,289.42	5,529,955	64%	85.25	5,506,365	64%
			1000*	18,000.34	1,631,616.00	7,423,535	78%	344.34	7,173,015	77%
		400	800	18,001.02	1,239,039.00	5,935,580	79%	862.92	5,184,505	76%
			1000	18,000.36	1,631,616.00	7,898,555	79%	29.88	6,840,685	76%
	20	200	800	18,003.31	1,239,039.00	5,260,375	76%	17.40	5,200,175	76%
			1000	18,003.17	1,631,616.00	6,770,955	76%	13.01	6,741,335	76%
		400	800	18,001.17	1,239,039.00	5,602,060	78%	61.39	5,176,145	76%
			1000	18,000.81	1,631,616.00	7,569,570	78%	101.80	6,665,325	76%

Table 4: Results obtained by CPLEX (TO=5h) and SA (TO=15min) for small-sized 2-zone instances with $|V| = 10$ and $|V| = 20$ using $u = 10$. Parameters ($rcc, mcaf$) are set to (0.8, 0.8) except for the instance noted with asterisk (*), for which they were reduced to (0.4, 0.4) in order to ensure feasibility. SA's parameters are set to $T_0 = 10^5$ and $cs = 2 \cdot 10^{-5}$

the same TCTDs were found. To assess the statistical significance of the comparison, the Wilcoxon signed-rank test (Wilcoxon, 1945) was conducted at a 5% significance level for the solutions of CPLEX and SA over all 32 small-sized instances. The obtained p-value was equal to 0.000196, providing strong evidence that the heuristic's superiority is statistically significant.

To analyse the influence of the proposed distance-related inequalities (Section 3.3), the Linear Programming (LP) relaxation of the MIP model was solved with and without Constraints (26) and (27) for the small-sized instances. The strengthened formulation generated improved bounds for all instances, showing that it strictly dominates the original formulation. The average improvement was 165% for 1-zone and 225% for 2-zone instances. Tables A.15 and A.16 in Appendix A report the solutions obtained by CPLEX in detail.

The next set of experiments compares the solutions obtained with fixed and flexible departure settings. Both variants of the heuristic were tested over all medium and large-sized instances given 15 minutes of runtime and each run was repeated 20 times with different random seed values. Tables 5-8 report the average TCTD without considering infeasible solutions and the number of infeasible solutions obtained for each instance across 20 runs. The gap between average TCTDs considering fixed and flexible departure settings is also detailed. For the same set of experiments, Tables A.17 and A.18 in Appendix A report the p-values obtained by Wilcoxon rank-sum test (Wilcoxon, 1945) conducted at a 5% significance level between the sets of solutions found for fixed and flexible departure settings for each instance.

As expected, flexible departure times significantly improve solution quality across all instances in a statistically significant manner. The corresponding gaps consistently increase when it comes to instances with higher numbers of available cranes ($|C|$) and container loads (N_v^P), where the flexible setting enables to reduce TCTD by up to 26% for medium-sized and 20% for large-sized instances. The heuristic's behaviour with respect to these instances also demonstrates how the flexible setting is generally capable of better exploiting available cranes, providing consistently better solutions compared to instances with fewer $|C|$. This does not hold for increased crane range $|q_c|$, which frequently results in increased costs for both variants. Regarding infeasibility rates in congested instances, the flexible setting enables the algorithm to significantly decrease and eliminate infeasible solutions for almost all instances.

The final set of experiments was conducted to analyse the trade-off be-

Instances (1-zone)				Fixed		Flex		Gap
$ V $	$ C $	q_c	N_v^P	TCTD(avg)	# Inf	TCTD(avg)	# Inf	
20	20	600	800	3,050,245.10	0	2,568,946.70	0	16%
			1000	3,916,949.20	0	3,102,530.20	0	21%
		800	800	3,060,654.90	0	2,588,017.00	0	15%
			1000	3,934,854.40	0	3,085,724.00	0	22%
	30	600	800	3,055,667.40	0	2,464,925.50	0	19%
			1000	3,913,611.60	0	3,045,836.10	0	22%
		800	800	3,060,887.70	0	2,532,793.80	0	17%
			1000	3,941,172.70	0	3,087,658.40	0	22%
25	20	600	800	5,210,686.30	0	4,646,956.80	0	11%
			1000	6,650,247.60	0	5,630,927.50	0	15%
		800	800	5,256,707.30	0	4,790,610.60	0	9%
			1000	6,963,043.89	1	5,787,078.50	0	17%
	30	600	800	5,251,932.20	0	4,542,396.10	0	14%
			1000	6,678,973.80	0	5,427,521.80	0	19%
		800	800	5,449,817.56	2	4,692,290.60	0	14%
			1000	6,877,213.18	3	5,541,612.50	0	19%
	30	20	600	6,532,083.26	1	5,216,353.90	0	20%
			1000	8,275,124.00	5	6,504,240.80	0	21%
		800	800	6,765,644.67	5	5,530,273.80	0	18%
			1000	8,157,760.50	12	6,692,248.30	0	18%
30	30	600	800	6,510,246.84	1	4,932,488.80	0	24%
			1000	8,319,170.77	7	6,140,269.50	0	26%
		800	800	6,161,958.62	7	5,179,067.56	2	16%
			1000	7,995,634.18	9	6,396,858.50	0	20%

Table 5: Results generated by SA with TO=15min for medium-sized 1-zone instances with fixed and flexible departure settings using $u = 2m$. Parameters ($rcc, mcaf$) are set to (0.7, 0.01). SA's parameters are set to $T_0 = 2 \cdot 10^5$ and $cs = 7 \cdot 10^{-5}$

Instances (2-zone)				Fixed		Flex		Gap
$ V $	$ C $	q_c	N_v^P	TCTD(avg)	# Inf	TCTD(avg)	# Inf	
20	20	600	800	6,014,792.65	0	4,739,262.85	0	21%
			1000	7,630,909.85	0	5,794,540.65	0	24%
		800	800	6,007,925.75	0	5,073,110.80	0	16%
			1000	7,630,540.45	0	5,829,080.05	0	24%
	30	600	800	6,013,906.55	0	4,869,860.40	0	19%
			1000	7,656,507.15	0	5,697,968.15	0	26%
		800	800	6,021,316.60	0	5,118,432.80	0	15%
			1000	7,644,665.05	0	5,762,033.40	0	25%
	25	20	600	10,047,003.65	0	9,507,589.95	0	5%
			1000	12,534,487.85	0	11,431,006.15	0	9%
		800	800	10,079,017.60	0	9,544,092.20	0	5%
			1000	12,594,277.35	0	11,355,137.05	0	10%
	30	600	800	10,077,318.05	0	9,472,757.20	0	6%
			1000	12,740,469.00	0	11,298,272.80	0	11%
		800	800	10,105,361.25	0	9,555,801.95	0	5%
			1000	12,530,952.10	0	11,428,753.75	0	9%
30	20	600	800	11,710,115.94	2	10,575,776.05	0	10%
			1000	15,112,052.10	0	13,022,303.15	0	14%
		800	800	11,816,674.68	1	10,718,481.35	0	9%
			1000	15,050,812.21	1	13,041,150.70	0	13%
	30	600	800	11,886,743.53	5	10,482,228.15	0	12%
			1000	14,996,523.65	3	12,985,981.00	0	13%
		800	800	11,760,491.89	1	10,861,075.50	0	8%
			1000	15,154,315.18	3	12,913,992.65	0	15%

Table 6: Results generated by SA with TO=15min for medium-sized 2-zone instances with fixed and flexible departure settings using $u = 2m$. Parameters ($rcc, mcaf$) are set to (0.7, 0.01). SA's parameters are set to $T_0 = 2 \cdot 10^5$ and $cs = 7 \cdot 10^{-5}$

Instances (1-zone)				Fixed		Flex		Gap
$ V $	$ C $	q_c	N_v^P	TCTD(avg)	# Inf	TCTD(avg)	# Inf	
40	40	600	800	13,395,025.80	0	11,607,413.70	0	13%
			1000	15,811,718.90	0	13,300,665.70	0	16%
		800	800	13,553,886.20	0	11,581,902.60	0	15%
			1000	15,983,987.70	0	13,405,220.60	0	16%
	60	600	800	13,376,795.60	0	11,361,350.40	0	15%
			1000	15,885,259.60	0	12,998,885.80	0	18%
		800	800	13,605,043.40	0	11,332,825.50	0	17%
			1000	16,000,320.60	0	13,018,814.60	0	19%
	50	40	600	19,022,596.80	0	16,598,703.40	0	13%
			1000	24,759,345.00	0	20,826,607.40	0	16%
		800	800	19,232,388.50	0	16,684,677.10	0	13%
			1000	25,355,314.60	0	20,953,397.60	0	17%
	60	600	800	19,152,835.20	0	16,342,837.00	0	15%
			1000	24,656,933.20	0	20,195,981.60	0	18%
		800	800	19,222,865.50	0	16,377,153.00	0	15%
			1000	25,130,495.20	0	20,134,681.10	0	20%
60	40	600	800	23,907,282.63	1	20,688,270.40	0	13%
			1000	32,560,166.27	5	26,303,775.40	0	19%
		800	800	24,073,310.40	5	21,406,059.20	0	11%
			1000	32,640,791.00	12	27,698,712.20	0	15%
	60	600	800	23,905,160.60	0	20,650,420.60	0	14%
			1000	32,279,207.25	4	25,960,561.30	0	20%
		800	800	23,824,742.13	5	20,599,275.10	0	14%
			1000	32,594,000.46	7	26,194,973.20	0	20%

Table 7: Results generated by SA with TO=15min for large-sized 1-zone instances with fixed and flexible departure settings using $u = 2m$. Parameters $(rcc, mcaf)$ are set to $(0.7, 0.01)$. SA's parameters are set to $T_0 = 2 \cdot 10^5$ and $cs = 7 \cdot 10^{-5}$

Instances (2-zone)				Fixed		Flex		Gap
$ V $	$ C $	q_c	N_v^P	TCTD(avg)	# Inf	TCTD(avg)	# Inf	
40	40	600	800	20,307,333.35	0	17,865,894.30	0	12%
			1000	24,376,544.45	0	20,305,139.75	0	17%
		800	800	20,563,122.65	0	18,767,662.65	0	9%
			1000	24,471,413.70	0	21,330,746.10	0	13%
	60	600	800	20,485,387.65	0	17,128,145.10	0	16%
			1000	24,525,999.35	0	19,625,744.05	0	20%
		800	800	20,840,405.20	0	18,602,284.85	0	11%
			1000	24,568,929.20	0	20,684,531.50	0	16%
	50	40	600	28,145,745.30	0	26,003,492.45	0	8%
			1000	36,796,398.45	0	31,703,401.50	0	14%
		800	800	28,345,373.35	0	26,871,186.95	0	5%
			1000	37,464,243.80	0	33,161,922.95	0	11%
60	60	600	800	27,998,834.70	0	25,887,109.65	0	8%
			1000	36,685,971.70	0	31,234,240.80	0	15%
		800	800	28,481,687.55	0	26,558,187.60	0	7%
			1000	36,901,831.80	0	32,714,407.05	0	11%
	40	600	800	33,623,494.12	3	31,320,296.95	0	7%
			1000	45,827,426.82	3	40,487,075.40	0	12%
		800	800	33,909,027.08	7	31,639,650.60	0	7%
			1000	46,452,052.67	8	42,161,511.95	0	9%
	60	600	800	33,817,669.29	6	30,597,966.70	0	10%
			1000	45,647,517.08	7	39,511,669.15	0	13%
		800	800	33,942,633.64	6	31,666,983.45	0	7%
			1000	46,573,482.90	10	41,259,977.00	0	11%

Table 8: Results generated by SA with TO=15min for large-sized 2-zone instances with fixed and flexible departure settings using $u = 2m$. Parameters $(rcc, mcaf)$ are set to $(0.7, 0.01)$. SA's parameters are set to $T_0 = 2 \cdot 10^5$ and $cs = 7 \cdot 10^{-5}$

tween proposed heuristic's runtime and solution quality. Both variants of the heuristic were now given 30 minutes of runtime to solve all medium- and large-sized instances and each run was again repeated 20 times with different random seed values. Tables 9-12 report the obtained results in the same manner as Tables 5-8. For each run of the same set of experiments, the gap of the best solution obtained after $(\frac{i}{6} * numTotalIterations)$ SA iterations from the first feasible solution found, where *numTotalIterations* is the total number of SA iterations performed during the run, was computed for $i \in [1, 6]$. Tables A.19-A.22 in Appendix A report the average gap for each set of instances and for each of the heuristic's variants for both 1-zone and 2-zone settings. The results are divided into groups with respect to $|V|$ and $|C|$ and infeasible solutions are ignored. For each group the total number of feasible solutions found after the same number of SA iterations across all 160 corresponding runs is also reported.

The flexible setting allows for greater improvements with the extended runtime, as shown by the generally increased gaps in Tables 9-12 compared to Tables 5-8. These improvements are more significant for the medium-sized instances and are also demonstrated by the gaps from the first feasible solutions in Tables A.19-A.22, which are higher for the flexible setting for all groups of instances.

Regarding the general behaviour of the proposed heuristic over time, for instances with small numbers of vessels ($|V| = 20$ for medium- and $|V| = 40$ for large-sized instances) the fixed-setting variant results in no significant improvements after the first few SA iterations. For the same instances, the flexible setting continues to exhibit improvements until two thirds of the SA iterations are completed. As problem size grows, greater differences are observed for both variants during the final stages of the search process. However, for the fixed setting this improvement is primarily incurred by the increased numbers of feasible solutions found, while for the flexible setting the respective gaps are also consistently improved.

6. Conclusions and future directions

Recent developments concerning management techniques for maritime freight transportation along with ever-increasing logistical pressures are leading port terminals to a transition away from rules of thumb to algorithmic approaches. However, when the BAP is addressed at a tactical level, various real-world limitations are often ignored by current academic models.

Instances (1-zone)				Fixed		Flex		Gap
$ V $	$ C $	q_c	N_v^P	TCTD(avg)	# Inf	TCTD(avg)	# Inf	
20	20	600	800	3,047,241.20	0	2,534,322.30	0	17%
			1000	3,905,797.90	0	3,077,190.10	0	21%
		800	800	3,054,758.00	0	2,553,922.10	0	16%
			1000	3,923,289.80	0	3,072,637.40	0	22%
	30	600	800	3,050,170.00	0	2,426,236.60	0	20%
			1000	3,904,200.40	0	2,981,369.60	0	24%
		800	800	3,057,365.90	0	2,497,960.50	0	18%
			1000	3,931,894.80	0	3,017,534.20	0	23%
	25	20	600	5,192,481.20	0	4,601,248.80	0	11%
			1000	6,589,028.20	0	5,572,620.80	0	15%
		800	800	5,233,928.80	0	4,736,763.50	0	9%
			1000	6,917,734.40	0	5,595,521.30	0	19%
30	30	600	800	5,219,472.60	0	4,500,142.00	0	14%
			1000	6,618,608.60	0	5,408,416.20	0	18%
		800	800	5,351,298.50	0	4,595,902.40	0	14%
			1000	6,859,661.89	1	5,437,813.00	0	21%
	20	600	800	6,503,853.37	1	5,116,810.20	0	21%
			1000	8,131,171.50	4	6,238,748.10	0	23%
		800	800	6,704,686.63	4	5,433,050.00	0	19%
			1000	8,183,077.60	10	6,592,751.80	0	19%
	30	600	800	6,478,587.89	1	4,878,788.70	0	25%
			1000	8,273,387.00	6	6,036,001.00	0	27%
		800	800	6,150,943.73	5	5,273,895.40	0	14%
			1000	7,825,844.57	6	6,318,008.60	0	19%

Table 9: Results generated by SA with TO=30min for medium-sized 1-zone instances with fixed and flexible departure settings using $u = 2m$. Parameters ($rcc, mcaf$) are set to (0.7, 0.01). SA's parameters are set to $T_0 = 2 \cdot 10^5$ and $cs = 7 \cdot 10^{-5}$

Instances (2-zone)				Fixed		Flex		Gap
$ V $	$ C $	q_c	N_v^P	TCTD(avg)	# Inf	TCTD(avg)	# Inf	
20	20	600	800	5,985,690.05	0	4,616,731.80	0	23%
			1000	7,598,366.20	0	5,486,128.95	0	28%
		800	800	5,990,385.50	0	4,991,661.10	0	17%
			1000	7,598,039.85	0	5,559,157.80	0	27%
	30	600	800	6,002,274.65	0	4,764,518.50	0	21%
			1000	7,607,127.00	0	5,382,093.35	0	29%
		800	800	6,001,099.35	0	4,993,246.80	0	17%
			1000	7,600,660.10	0	5,634,499.55	0	26%
	25	20	600	10,024,855.45	0	9,411,862.65	0	6%
			1000	12,454,337.30	0	11,324,839.65	0	9%
		800	800	10,060,595.15	0	9,468,533.25	0	6%
			1000	12,534,290.55	0	11,273,199.55	0	10%
	30	600	800	10,062,872.50	0	9,417,496.75	0	6%
			1000	12,599,724.25	0	11,223,991.95	0	11%
		800	800	10,053,913.00	0	9,503,955.50	0	5%
			1000	12,517,380.50	0	11,221,978.15	0	10%
30	20	600	800	11,561,753.05	0	10,295,632.00	0	11%
			1000	14,934,674.35	0	12,914,543.05	0	14%
		800	800	11,735,427.11	1	10,438,649.35	0	11%
			1000	15,015,756.47	1	12,676,899.85	0	16%
	30	600	800	11,683,463.28	2	10,422,060.35	0	11%
			1000	14,907,739.30	0	12,494,762.85	0	16%
		800	800	11,639,413.37	1	10,694,051.15	0	8%
			1000	15,148,529.83	2	12,692,881.45	0	16%

Table 10: Results generated by SA with TO=30min for medium-sized 2-zone instances with fixed and flexible departure settings using $u = 2m$. Parameters ($rcc, mcaf$) are set to (0.7, 0.01). SA's parameters are set to $T_0 = 2 \cdot 10^5$ and $cs = 7 \cdot 10^{-5}$

Instances (1-zone)				Fixed		Flex		Gap
$ V $	$ C $	q_c	N_v^P	TCTD(avg)	# Inf	TCTD(avg)	# Inf	
40	40	600	800	13,314,114.90	0	11,456,576.20	0	14%
			1000	15,617,315.70	0	13,107,192.10	0	16%
		800	800	13,415,156.80	0	11,466,160.50	0	15%
			1000	15,773,856.20	0	13,176,758.50	0	16%
	60	600	800	13,291,444.10	0	11,161,244.70	0	16%
			1000	15,749,476.50	0	12,766,017.90	0	19%
		800	800	13,434,331.10	0	11,214,007.90	0	17%
			1000	15,810,249.50	0	12,852,998.30	0	19%
	50	40	600	18,889,397.00	0	16,461,152.20	0	13%
			1000	24,536,669.60	0	20,625,244.90	0	16%
		800	800	19,096,466.90	0	16,742,992.90	0	12%
			1000	24,997,169.10	0	20,689,293.00	0	17%
60	60	600	800	18,928,078.50	0	16,256,488.90	0	14%
			1000	24,329,372.40	0	19,960,790.80	0	18%
		800	800	19,019,539.20	0	16,172,133.10	0	15%
			1000	24,831,433.30	0	19,959,840.50	0	20%
	40	600	800	23,733,286.53	1	20,438,044.50	0	14%
			1000	32,239,802.25	4	25,782,324.50	0	20%
		800	800	23,950,767.20	5	21,184,455.00	0	12%
			1000	32,606,455.00	10	26,877,800.10	0	18%
	60	600	800	23,659,473.80	0	20,047,201.80	0	15%
			1000	32,047,691.18	3	25,393,739.20	0	21%
		800	800	23,737,394.24	3	20,319,558.80	0	14%
			1000	32,270,535.38	7	25,865,053.50	0	20%

Table 11: Results generated by SA with TO=30min for large-sized 1-zone instances with fixed and flexible departure settings using $u = 2m$. Parameters $(rcc, mcaf)$ are set to $(0.7, 0.01)$. SA's parameters are set to $T_0 = 2 \cdot 10^5$ and $cs = 7 \cdot 10^{-5}$

Instances (2-zone)				Fixed		Flex		Gap
$ V $	$ C $	q_c	N_v^P	TCTD(avg)	# Inf	TCTD(avg)	# Inf	
40	40	600	800	20,130,042.45	0	17,545,321.10	0	13%
			1000	23,991,707.10	0	19,491,000.50	0	19%
		800	800	20,383,341.10	0	18,180,489.35	0	11%
			1000	24,150,683.70	0	20,854,552.40	0	14%
	60	600	800	20,240,866.45	0	17,015,344.05	0	16%
			1000	24,120,563.10	0	19,223,158.45	0	20%
		800	800	20,553,120.40	0	18,356,082.90	0	11%
			1000	24,127,852.65	0	20,349,493.55	0	16%
	50	40	600	27,963,182.35	0	25,716,764.55	0	8%
			1000	36,469,247.05	0	31,697,071.75	0	13%
		800	800	28,163,032.30	0	26,660,490.75	0	5%
			1000	37,122,037.70	0	32,487,747.80	0	12%
	60	600	800	27,807,301.80	0	25,402,102.50	0	9%
			1000	36,339,735.15	0	30,822,047.30	0	15%
		800	800	28,275,992.55	0	26,157,511.70	0	7%
			1000	36,451,162.25	0	32,208,727.45	0	12%
60	40	600	800	33,432,238.72	2	31,265,515.15	0	6%
			1000	45,507,108.58	1	40,003,036.25	0	12%
		800	800	33,733,099.64	6	31,588,225.05	1	6%
			1000	46,189,855.62	7	41,037,954.60	0	11%
	60	600	800	33,552,300.00	4	30,520,308.95	0	9%
			1000	45,410,524.13	5	38,913,445.20	0	14%
		800	800	33,640,153.87	5	31,334,619.75	0	7%
			1000	46,154,032.17	8	40,324,064.65	0	13%

Table 12: Results generated by SA with TO=30min for large-sized 2-zone instances with fixed and flexible departure settings using $u = 2m$. Parameters $(rcc, mcaf)$ are set to $(0.7, 0.01)$. SA's parameters are set to $T_0 = 2 \cdot 10^5$ and $cs = 7 \cdot 10^{-5}$

To this end, the present paper introduced the TBACASP, which integrates Specific Quay Crane Assignment decisions into the Tactical Berth Allocation Problem with the aim of reducing the transshipment-related yard costs. The presented MIP formulation brings together numerous real-world attributes, all of which contribute toward the highly realistic representation of an actual terminal. The model was solved to optimality for instances with 10 vessels and 100 berth slots. However, due to unacceptably long computation times, a heuristic intended for real-time decision support based on the developed formulation was also proposed. Extensive computational experiments confirm the strong performance of the heuristic over a set of diverse instances.

The heuristic can be employed under both fixed and flexible departure times settings for the arriving vessels. It thus enables the quantification of hidden gains which can be achieved by allowing the vessels to depart as soon as their container requirements are served. Results demonstrate how permitting flexible departure times can significantly improve berthing schedules and reduce container transshipment distances by more than 25% with minimal computational effort.

The operational effort required by terminals to enable this flexibility is not considered in this paper. Future work which sets out to quantify the correlated costs would therefore enable an accurate assessment of its impact.

The present paper lays the foundation for bringing the computational model of tactical berth allocation closer to reality. The incorporation of additional objectives such as yard occupancy, container relocations and balancing quay crane usage to the model would thus be of great significance, compounding further improvements upon the derived schedules. Finally, although the heuristic was implemented with the intent of producing high quality solutions in short runtimes, future research geared towards effectively addressing the dynamic variant of the TBACASP would also represent a relevant contribution to the domain.

References

- C. Bierwirth, F. Meisel, A follow-up survey of berth allocation and quay crane scheduling problems in container terminals, *European Journal of Operational Research* 244 (2015) 675–689.
- C. Bierwirth, F. Meisel, A survey of berth allocation and quay crane scheduling problems in container terminals, *European Journal of Operational Research* 202 (2010) 615–627.

- F. Li, J.-B. Sheu, Z.-Y. Gao, Solving the continuous berth allocation and specific quay crane assignment problems with quay crane coverage range, *Transportation Science* 49 (2015) 968–989.
- Y. M. Park, K. H. Kim, A scheduling method for Berth and Quay cranes, *Container Terminals and Automated Transport Systems: Logistics Control Issues and Quantitative Decision Support* (2005) 159–181.
- A. Lim, The berth planning problem, *Operations research letters* 22 (1998) 105–110.
- P. Hansen, C. Oguz, A note on formulations of the static and dynamic berth allocation problems, *Les Cahiers du GERAD* 30 (2003) 1–17.
- A. Imai, H. C. Chen, E. Nishimura, S. Papadimitriou, The simultaneous berth and quay crane allocation problem, *Transportation Research Part E: Logistics and Transportation Review* 44 (2008) 900–920.
- F. Meisel, C. Bierwirth, Heuristics for the integration of crane productivity in the berth allocation problem, *Transportation Research Part E: Logistics and Transportation Review* 45 (2009) 196–209.
- C. Zhang, L. Zheng, Z. Zhang, L. Shi, A. J. Armstrong, The allocation of berths and quay cranes by using a sub-gradient optimization technique, *Computers and Industrial Engineering* 58 (2010) 40–50.
- G. Giallombardo, L. Moccia, M. Salani, I. Vacca, Modeling and solving the Tactical Berth Allocation Problem, *Transportation Research Part B: Methodological* 44 (2010) 232–245.
- B. Raa, W. Dullaert, R. Van Schaeren, An enriched model for the integrated berth allocation and quay crane assignment problem, *Expert Systems with Applications* 38 (2011) 14136–14147.
- C. Yang, X. Wang, Z. Li, An optimization approach for coupling problem of berth allocation and quay crane assignment in container terminal, *Computers and Industrial Engineering* (2012).
- F. Meisel, C. Bierwirth, A Framework for Integrated Berth Allocation and Crane Operations Planning in Seaport Container Terminals, *Transportation Science* 47 (2013) 131–147.

- I. Vacca, M. Salani, M. Bierlaire, An Exact Algorithm for the Integrated Planning of Berth Allocation and Quay Crane Assignment, *Transportation Science* 47 (2013) 148–161.
- E. Lalla-Ruiz, J. L. González-Velarde, B. Melián-Batista, J. M. Moreno-Vega, Biased random key genetic algorithm for the Tactical Berth Allocation Problem, *Applied Soft Computing Journal* 22 (2014) 60–76.
- M. Rodriguez-Molins, M. A. Salido, F. Barber, A GRASP-based metaheuristic for the Berth Allocation Problem and the Quay Crane Assignment Problem by managing vessel cargo holds, *Applied Intelligence* 40 (2014) 273–290.
- Y. B. Türkoğullari, Z. C. Taşkin, N. Aras, I. K. Altinel, Optimal berth allocation and time-invariant quay crane assignment in container terminals, *European Journal of Operational Research* 235 (2014) 88–101.
- H. Xiaolong, X. Gong, J. Jo, A new continuous berth allocation and quay crane assignment model in container terminal, *Computers & Industrial Engineering* 89 (2015) 15–22.
- Z. H. Hu, Heuristics for solving continuous berth allocation problem considering periodic balancing utilization of cranes, *Computers and Industrial Engineering* (2015).
- A. Karam, A. B. Eltawil, Functional integration approach for the berth allocation, quay crane assignment and specific quay crane assignment problems, *Computers and Industrial Engineering* 102 (2016) 458–466.
- M. Liu, C. Y. Lee, Z. Zhang, C. Chu, Bi-objective optimization for the container terminal integrated planning, *Transportation Research Part B: Methodological* 93 (2016) 720–749.
- Y. B. Türkoğullari, Z. C. Taşkin, N. Aras, I. K. Altinel, Optimal berth allocation, time-variant quay crane assignment and scheduling with crane setups in container terminals, *European Journal of Operational Research* 254 (2016) 985–1001.
- J. F. Correcher, R. Alvarez-Valdes, A biased random-key genetic algorithm for the time-invariant berth allocation and quay crane assignment problem, *Expert Systems with Applications* 89 (2017) 112–128.

- X. Xiang, C. Liu, L. Miao, Reactive strategy for discrete berth allocation and quay crane assignment problems under uncertainty, *Computers and Industrial Engineering* 126 (2018) 196–216.
- A. Agra, M. Oliveira, MIP approaches for the integrated berth allocation and quay crane assignment and scheduling problem, *European Journal of Operational Research* (2018).
- J. F. Correcher, R. Alvarez-Valdes, J. M. Tamarit, New exact methods for the time-invariant berth allocation and quay crane assignment problem, *European Journal of Operational Research* 275 (2019) 80–92.
- H. Ma, S. Chung, H. Chan, L. Cui, An integrated model for berth and yard planning in container terminals with multi-continuous berth layout, *Annals of Operations Research* 273 (2019) 409–431.
- M. P. M. Hendriks, E. Lefeber, J. T. Udding, Simultaneous berth allocation and yard planning at tactical level, *OR Spectrum* 35 (2013) 441–456.
- Sea-Intelligence Maritime Analysis, Global Liner Performance Report – August 2018, https://www.sea-intelligence.com/images/products/glp_84.pdf, 2018.
- E. Thanos, T. Van den Bosche, H. Gambini Santos, W. Vancroonenburg, G. Vanden Berghe, Feature-based tuning of a Tactical Berth Allocation heuristic, in: R. Bai, Z.-L. Chen, L. Jiawei, G. Kendall, B. McCollum (Eds.), *Proceedings of the 9th MISTA conference*, 12 - 15 Dec 2019, Ningbo, China, pp. 410–428.
- S. Kirkpatrick, C. D. Gelatt, M. P. Vecchi, Optimization by simulated annealing, *Science* 220 (1983) 671–680.
- F. Wilcoxon, Individual comparisons by ranking methods, *Biometrics bulletin* 1 (1945) 80–83.

Appendix A.

Algorithm 1 Best-fit constructive method

Input: A current solution $\mathcal{S}$

```

1: for all  $i \in V_U^{\mathcal{S}}$  do
2:    $Best, BestR \leftarrow \emptyset$ 
3:    $BestCost \leftarrow \infty$ 
4:   for all  $(i, k) \in A_i$  do
5:     if  $noconflicts((i, k), A^{\mathcal{S}})$  then
6:       if  $Cost((i, k), A^{\mathcal{S}}) < BestCost$  then
7:          $AssignCranes((i, k))$ 
8:         if  $Covered((i, k), R^{\mathcal{S}, i})$  then
9:            $Best \leftarrow (i, k)$ 
10:           $BestCost \leftarrow Cost((i, k), A^{\mathcal{S}})$ 
11:           $BestR \leftarrow R^{\mathcal{S}, i}$ 
12:        end if
13:      end if
14:    end if
15:  end for
16:  if  $Best \neq \emptyset$  then
17:     $A_i^{\mathcal{S}} \leftarrow Best$ 
18:     $R^{\mathcal{S}, i} \leftarrow BestR$ 
19:    Remove  $i$  from  $V_U^{\mathcal{S}}$ 
20:  end if
21: end for

```

Algorithm 2 *AssignCranes*(i, k)

Input: A current solution $\mathcal{S}$ and a BA $(i, k) \in A^{\mathcal{S}}$

```

1:  $R_{OLD}^{\mathcal{S}} \leftarrow R^{\mathcal{S},i}$ 
2:  $R^{\mathcal{S},i} \leftarrow \emptyset$ 
3: Sort  $C_{i,k}$  in descending order of  $fr_{i,k,c}$ 
4: for all  $c \in C_{i,k}$  do
5:    $maxInterval \leftarrow [start_i, end_i]$ 
6:   for all  $r \in (R^{\mathcal{S},1}, \cup \dots R^{\mathcal{S},|V|})$  do
7:     if Incompatible( $r, (i, k, c, maxInterval^-, maxInterval^+)$ ) then
8:        $maxInterval \leftarrow maxInterval \setminus maxInterval \cap [start_r, end_r]$ 
9:       if getLongestContinuousInterval( $maxInterval$ )  $< mcaf \cdot (end_i - start_i)$  then
10:        break
11:      end if
12:    end if
13:  end for
14:  if getLongestContinuousInterval( $maxInterval$ )  $\geq mcaf \cdot (end_i - start_i)$  then
15:    if getRandom(0, 1)  $< 0.01$  then
16:      continue
17:    end if
18:     $maxInterval \leftarrow \text{getLongestContinuousInterval}(maxInterval)$ 
19:     $R^{\mathcal{S},i} \leftarrow R^{\mathcal{S},i} \cup (i, k, c, maxInterval^-, maxInterval^+)$ 
20:    if Covered(( $i, k$ ),  $R^{\mathcal{S},i}$ )  $\vee (|R^{\mathcal{S},i}| = maxNumCranes_i)$  then
21:      break
22:    end if
23:  end if
24: end for

```

Algorithm 3 *Simulated annealing*

Input: Initial solution S_0 , initial temperature T_0 , cooling rate cs

```

1:  $S \leftarrow S_0$ 
2:  $S^* \leftarrow S_0$ 
3:  $T \leftarrow T_0$ 
4:  $dT \leftarrow cs * T_0$ 
5: while time limit not exceeded do
6:   Get random neighbour  $S'$  of  $S$ 
7:    $\Delta \leftarrow TCTD(S') - TCTD(S)$ 
8:   if  $\Delta < 0$  then
9:      $S \leftarrow S'$ 
10:    if  $TCTD(S') < TCTD(S^*)$  then
11:       $S^* \leftarrow S$ 
12:    end if
13:  else
14:    Get random  $p \in [0, 1]$ 
15:    if  $p < e^{-\Delta/T}$  then
16:       $S \leftarrow S'$ 
17:    end if
18:  end if
19:   $T \leftarrow T - dT$ 
20:  if  $T < 10^{-3}$  then
21:     $T \leftarrow T_0$ 
22:  end if
23: end while
24: return  $S^*$ 

```

Instances (1-zone)				MIP				SA		
$ V $	$ C $	q_c	N_v^P	Time(s)	LB	Best Int	Gap	Time(s)	TCTD	Gap
10	10	200	800*	1,886.69	750,730.00	750,730	0%	3.27	750,730	0%
			1000*	828.55	920,510.00	920,510	0%	12.11	920,510	0%
		400	800	7,842.78	750,730.00	750,730	0%	5.10	750,730	0%
			1000	9,324.48	918,460.00	918,460	0%	4.96	918,460	0%
	20	200	800	16,133.36	750,730.00	750,730	0%	8.83	750,730	0%
			1000	18,000.14	597,092.66	918,460	35%	4.11	918,460	35%
		400	800	18,000.03	530,790.83	750,730	29%	8.20	750,730	29%
			1000	18,000.36	568,585.97	924,490	38%	5.30	918,460	38%
	20	10	200	18,000.19	1,042,363.32	1,720,970	39%	90.12	1,716,020	39%
			1000*	18,000.23	1,239,889.00	2,302,680	46%	106.43	2,266,000	45%
		400	800	18,000.23	932,815.00	1,951,200	52%	7,653.12	1,658,140	44%
			1000	18,000.30	1,239,889.00	2,721,800	54%	16,095.53	2,191,820	43%
		20	200	18,000.25	932,815.00	1,807,600	48%	1,034.71	1,658,140	44%
			1000	18,000.61	1,239,889.00	2,220,250	44%	221.72	2,158,840	43%
		400	800	18,001.22	932,815.00	2,129,100	56%	1,446.46	1,658,670	44%
			1000	18,001.06	1,239,889.00	2,630,070	53%	7,021.54	2,158,840	43%

Table A.13: Results obtained by CPLEX and SA (TO=5h) for small-sized 1-zone instances with $|V| = 10$ and $|V| = 20$ using the same configuration with Table 3

Instances (2-zone)				MIP				SA		
$ V $	$ C $	q_c	N_v^P	Time(s)	LB	Best Int	Gap	Time(s)	TCTD	Gap
10	10	200	800	117.31	2,188,895.00	2,188,895	0%	5.59	2,188,895	0%
			1000	11.36	2,670,665.00	2,670,665	0%	40.27	2,670,665	0%
		400	800	15,578.19	2,138,645.00	2,138,645	0%	2.79	2,138,645	0%
			1000	18,000.13	1,861,640.23	2,589,665	28%	2.33	2,589,665	28%
	20	200	800	5,513.02	2,138,645.00	2,138,645	0%	0.62	2,138,645	0%
			1000	4,745.77	2,593,655.00	2,593,655	0%	9.14	2,593,655	0%
		400	800	18,000.55	1,639,785.70	2,143,005	23%	8.09	2,138,645	23%
			1000	18,000.31	1,659,878.21	2,589,665	36%	4.59	2,589,665	36%
	20	10	200	18,000.25	1,983,289.42	5,529,955	64%	103.67	5,506,365	64%
			1000*	18,000.34	1,631,616.00	7,423,535	78%	1850.79	7,148,495	77%
		400	800	18,001.02	1,239,039.00	5,935,580	79%	977.85	5,183,625	76%
			1000	18,000.36	1,631,616.00	7,898,555	79%	1041.82	6,806,185	76%
		20	200	18,003.31	1,239,039.00	5,260,375	76%	21.68	5,200,175	76%
			1000	18,003.17	1,631,616.00	6,770,955	76%	15.17	6,741,335	76%
		400	800	18,001.17	1,239,039.00	5,602,060	78%	18.16	5,176,145	76%
			1000	18,000.81	1,631,616.00	7,569,570	78%	20.51	6,665,325	76%

Table A.14: Results obtained by CPLEX and SA (TO=5h) for small-sized 2-zone instances with $|V| = 10$ and $|V| = 20$ using the same configuration with Table 4

Instances (1-zone)				MIP	MIP + Constraints (26), (27)	Improvement
$ V $	$ C $	q_c	N_v^P			
10	10	200	800	194,950.00	477,675.00	145%
			1000	222,010.00	567,180.00	155%
		400	800	194,950.00	477,675.00	145%
			1000	222,010.00	567,180.00	155%
	20	200	800	194,950.00	477,675.00	145%
			1000	222,010.00	567,180.00	155%
		400	800	194,950.00	477,675.00	145%
			1000	222,010.00	567,180.00	155%
20	10	200	800	333,300.00	932,815.00	180%
			1000	440,840.00	1,239,889.00	181%
		400	800	333,300.00	932,815.00	180%
			1000	440,840.00	1,239,889.00	181%
	20	200	800	333,300.00	932,815.00	180%
			1000	440,840.00	1,239,889.00	181%
		400	800	333,300.00	932,815.00	180%
			1000	440,840.00	1,239,889.00	181%
Average improvement						165%

Table A.15: Solutions of the LP relaxation obtained by CPLEX for small-sized, 1-zone instances using the parameter settings of Table 3. **Improvement** equals to $\frac{LP_{with} - LP}{LP}$, where LP and LP_{with} are the solutions of the original formulation and the strengthened formulation, respectively.

Instances (2-zone)				MIP	MIP + Constraints (26), (27)	Improvement
$ V $	$ C $	q_c	N_v^P			
10	10	200	800	423,822.31	651,997.79	54%
			1000	659,282.86	896,760.18	36%
		400	800	194,950.00	616,675.00	216%
			1000	222,010.00	736,180.00	232%
	20	200	800	194,950.00	616,675.00	216%
			1000	222,010.00	736,180.00	232%
		400	800	194,950.00	616,675.00	216%
			1000	222,010.00	736,180.00	232%
20	10	200	800	333,300.00	1,239,039.00	272%
			1000	440,840.00	1,631,616.00	270%
		400	800	333,300.00	1,239,039.00	272%
			1000	440,840.00	1,631,616.00	270%
	20	200	800	333,300.00	1,239,039.00	272%
			1000	440,840.00	1,631,616.00	270%
		400	800	333,300.00	1,239,039.00	272%
			1000	440,840.00	1,631,616.00	270%
Average improvement				225%		

Table A.16: Solutions of the LP relaxation obtained by CPLEX for small-sized, 2-zone instances using the parameter settings of Table 4. **Improvement** equals to $\frac{LP_{with} - LP}{LP}$, where LP and LP_{with} are the solutions of the original formulation and the strengthened formulation, respectively.

Instances				p-values for Fixed-Flex comparisons	
$ V $	$ C $	q_c	N_v^P	1-zone settings	2-zone settings
20	20	600	800	5.61E-07	5.61E-07
			1000	5.61E-07	5.61E-07
		800	800	5.61E-07	5.61E-07
			1000	5.61E-07	5.61E-07
	30	600	800	5.61E-07	5.61E-07
			1000	5.61E-07	5.61E-07
		800	800	5.61E-07	5.61E-07
			1000	5.61E-07	5.61E-07
25	20	600	800	5.61E-07	5.61E-07
			1000	5.61E-07	5.61E-07
		800	800	5.61E-07	5.61E-07
			1000	5.61E-07	5.61E-07
	30	600	800	5.61E-07	5.61E-07
			1000	5.61E-07	5.61E-07
		800	800	5.61E-07	5.61E-07
			1000	5.61E-07	5.61E-07
	30	600	800	5.61E-07	7.36E-07
			1000	5.61E-07	5.61E-07
		800	800	5.61E-07	1.44E-06
			1000	5.61E-07	5.61E-07
30	20	600	800	5.61E-07	5.61E-07
			1000	5.61E-07	5.61E-07
		800	800	5.61E-07	5.61E-07
			1000	5.61E-07	5.61E-07
	30	600	800	5.61E-07	5.61E-07
			1000	5.61E-07	5.61E-07
		800	800	3.92E-05	3.13E-06
			1000	5.61E-07	5.61E-07

Table A.17: p-values obtained by Wilcoxon rank-sum test for comparisons between fixed and flexible departure settings in Tables 5 and 6

$ V $	Instances			p-values for Fixed-Flex comparisons	
	$ C $	q_c	N_v^P	1-zone settings	2-zone settings
40	40	600	800	5.61E-07	5.61E-07
			1000	5.61E-07	5.61E-07
		800	800	5.61E-07	5.61E-07
			1000	5.61E-07	5.61E-07
	60	600	800	5.61E-07	5.61E-07
			1000	5.61E-07	5.61E-07
		800	800	5.61E-07	5.61E-07
			1000	5.61E-07	5.61E-07
	50	40	600	800	5.61E-07
			1000	5.61E-07	5.61E-07
		800	800	5.61E-07	5.61E-07
			1000	5.61E-07	5.61E-07
	60	600	800	5.61E-07	5.61E-07
			1000	5.61E-07	5.61E-07
		800	800	5.61E-07	5.61E-07
			1000	5.61E-07	5.61E-07
60	40	600	800	5.61E-07	5.61E-07
			1000	5.61E-07	5.61E-07
		800	800	5.61E-07	8.43E-07
			1000	5.61E-07	5.61E-07
	60	600	800	5.61E-07	5.61E-07
			1000	5.61E-07	5.61E-07
		800	800	5.61E-07	6.43E-07
			1000	5.61E-07	5.61E-07

Table A.18: p-values obtained by Wilcoxon rank-sum test for comparisons between fixed and flexible departure settings in Tables 7 and 8

Instances		i=1		i=2		i=3		i=4		i=5		i=6	
V	C	Gap	#F	Gap	#F	Gap	#F	Gap	#F	Gap	#F	Gap	#F
20	20	26.92%	160	27.31%	160	27.44%	160	27.53%	160	27.58%	160	27.63%	160
	30	24.27%	160	24.71%	160	24.89%	160	24.98%	160	25.04%	160	25.08%	160
25	20	10.04%	158	10.62%	159	10.96%	159	11.08%	159	11.26%	159	11.33%	160
	30	10.17%	139	10.86%	144	11.17%	146	11.11%	154	11.45%	154	11.41%	159
30	20	5.77%	100	6.40%	116	6.77%	122	7.10%	128	7.20%	137	7.44%	139
	30	6.06%	90	6.54%	110	6.97%	113	7.01%	123	7.45%	125	7.44%	137

Table A.19: Average gaps from first feasible solution (Gap) and numbers of feasible solutions (#F) found after $(i/6 * numTotalIterations)$ SA iterations for **medium-sized** instances with **fixed** departure setting

Instances		i=1		i=2		i=3		i=4		i=5		i=6	
V	C	Gap	#F	Gap	#F	Gap	#F	Gap	#F	Gap	#F	Gap	#F
20	20	29.70%	160	30.85%	160	31.76%	160	32.40%	160	32.93%	160	33.16%	160
	30	30.07%	160	31.52%	160	32.42%	160	32.98%	160	33.40%	160	33.54%	160
25	20	17.41%	160	18.20%	160	18.72%	160	19.07%	160	19.35%	160	19.53%	160
	30	18.48%	160	19.25%	160	19.73%	160	20.20%	160	20.46%	160	20.69%	160
30	20	15.66%	157	17.38%	159	18.43%	159	19.06%	160	19.56%	160	19.81%	160
	30	16.67%	150	18.24%	154	19.08%	155	19.69%	157	20.13%	160	20.44%	160

Table A.20: Average gaps from first feasible solution (Gap) and numbers of feasible solutions (#F) found after $(i/6 * numTotalIterations)$ SA iterations for **medium-sized** instances with **flexible** departure setting

Instances		i=1		i=2		i=3		i=4		i=5		i=6	
V	C	Gap	#F	Gap	#F	Gap	#F	Gap	#F	Gap	#F	Gap	#F
40	40	18.49%	160	19.68%	160	20.23%	160	20.42%	160	20.74%	160	20.77%	160
	60	18.38%	160	19.51%	160	19.99%	160	20.27%	160	20.62%	160	20.68%	160
50	40	12.35%	160	13.16%	160	13.72%	160	14.00%	160	14.21%	160	14.33%	160
	60	12.75%	160	13.78%	160	14.20%	160	14.54%	160	14.89%	160	14.97%	160
60	40	4.84%	67	5.40%	86	5.70%	101	5.87%	110	6.03%	120	6.07%	124
	60	5.29%	66	5.56%	87	5.72%	103	5.92%	110	6.12%	121	6.21%	125

Table A.21: Average gaps from first feasible solution (Gap) and numbers of feasible solutions (#F) found after $(i/6 * numTotalIterations)$ SA iterations for **large-sized** instances with **fixed** departure setting

Instances		i=1		i=2		i=3		i=4		i=5		i=6	
$ V $	$ C $	Gap	#F	Gap	#F	Gap	#F	Gap	#F	Gap	#F	Gap	#F
40	40	19.99%	160	20.93%	160	21.62%	160	22.00%	160	22.41%	160	22.51%	160
	60	18.54%	160	19.35%	160	20.02%	160	20.44%	160	20.67%	160	20.70%	160
50	40	15.84%	160	16.89%	160	17.55%	160	17.96%	160	18.30%	160	18.33%	160
	60	17.10%	160	18.03%	160	18.69%	160	19.04%	160	19.35%	160	19.47%	160
60	40	10.55%	149	11.97%	153	12.53%	156	13.13%	156	13.38%	158	13.53%	159
	60	10.79%	154	12.19%	157	13.07%	159	13.61%	159	13.96%	160	14.11%	160

Table A.22: Average gaps from first feasible solution (Gap) and numbers of feasible solutions (#F) found after $(i/6 * numTotalIterations)$ SA iterations for **large-sized** instances with **flexible** departure setting

HIGHLIGHTS

- The continuous berth allocation problem with time-variant, specific quay crane assignments is studied at a tactical level.
- A MIP model is introduced and instances with 10 vessels, 20 cranes and 100 berth positions are solved to optimality.
- A Remove & Reinsert heuristic is developed which can accommodate both fixed and flexible departure time settings.
- Allowing flexible departure times enables to reduce container transshipment distances by up to 26%.