Service fairness and value of customer information for the stochastic container relocation problem under flexible service policy

Yuanjun Feng[a], *Dong-Ping Song*[a*], *Dong Li*[a], *Ying Xie*[b]
[a]*School of Management, University of Liverpool, UK*
[b]*Cranfield School of Management, Cranfield University, UK*

(Citation: Feng, Y., Song, D.P., Li, D. and Xie, Y. (2022). Service fairness and value of customer information for the stochastic container relocation problem under flexible service policy, Transportation Research Part E. 167, 102921 https://doi.org/10.1016/j.tre.2022.102921.)

Abstract: This paper considers the optimization of service efficiency and service fairness in a Stochastic Container Relocation Problem (SCRP) under flexible service policies at a container terminal. Under the flexible service policy, the external trucks arriving within the same time window can be served out of sequence, which may raise a concern of service fairness. We incorporate the concept of service fairness into the SCRP in two phases. In phase 1, we propose a multiple sub-time windows-based flexible service policy, under which each time window will be divided into multiple sub-time windows and the flexible service policy is only applied to each individual sub-time window. In phase 2, the SCRP is formulated as a dynamic programming model with two lexicographically ordered objectives representing both relocation efficiency and service fairness, which is solved via a hierarchical iterative approach. In addition, we investigate whether the information of trucks' arrival probability over the time window (which represents the customer preference information) would add value to the terminal operators. Extensive computational experiments are conducted to evaluate the impacts of the number of sub-time windows and to examine the impacts and value of customer preference information in various scenarios. The results show interesting trade-offs between efficiency and fairness. As the number of sub-time windows increases, the service fairness is generally improving (but not guaranteed) while the expected number of relocations is increasing. It is found that the customer preference information can be valuable in some circumstances, especially when each truck indicates a certain arrival sub-time window.

Keywords: OR in port operations, stochastic container relocation, truck service policy, service fairness, value of information

1 Introduction

Container relocation (also known as reshuffling or rehandling) is a major source of inefficiency in the import container retrieval operations at most container terminals. Container relocation occurs when the target container is buried underneath other containers and these blocking containers have to be first moved away so that the target container can be retrieved. The container relocation operations are costly to terminal operators and cause delays to external trucks when retrieving containers. To reduce relocations, the Container Relocation Problem (CRP) has been extensively studied, which focuses on finding a sequence of moves with a minimum number of relocations to retrieve all containers from a yard bay, in response to certain container departure priorities. In reality, the departure priorities of import containers are uncertain due to the unpredictability of external trucks' arrival times. Ports would know the arrival precedence of trucks if the trucks book specific time windows in the truck appointment system or the vehicle booking system, but the exact arrival sequence of the trucks that have booked the same specific time window is still uncertain. The CRP that considers random truck arrivals in the same individual time window is called the CRP with Time Windows (CRPTW) or the Stochastic Container Relocation Problem (SCRP) (Galle et al., 2018). In the majority of the existing studies on the (S)CRP and practice, import containers are retrieved on the basis of first come first served (FCFS), that is, the container retrieval sequence aligns with the arrival order of their designated trucks. Under the flexible service policies where the external trucks are allowed to be served out-of-order, the retrieval sequence of containers can be determined by terminal operators to some extent and thus fewer relocations are needed compared to the FCFS policy (Borjian et al., 2015; Zeng et al. 2019; Feng et al., 2020).

However, the flexible policy might be perceived as unfair to some of the trucks involved, because the out-of-order service may cause some early arriving trucks to be served later than some later arriving trucks. In addition, it may also lead to a larger variation of the truck turnaround times, which further differentiates the customers' experience at container terminals. As a result, the issue of service fairness to customers (i.e., trucks) may become a barrier to implementing the flexible service policy. Considering fairness issues in the container retrieval service can bring tangible benefits to port operators. On one hand, consistently meeting truck turnaround time would enhance the credibility of the appointment system, as well as improve service reliability and customer satisfaction. On the other hand, introducing fairness in the container retrieval service prevents from causing an excessively long turnaround time for any truck, and thus may enhance a port's performance. The National Retail Federation made a recommendation to the U.S. Secretary of Transportation that the range for truck turnaround time should be listed as one of the port performance metrics (Knatz, 2017).

This paper studies the SCRP with flexible service policies by considering the service fairness issue, so as to strike a balance between two performances: the expected number of relocations and the concern of service fairness. In this paper, service fairness refers to avoiding excessively long turnaround time for any truck, in other words, keeping the expected maximum truck turnaround time as small as possible. There are two concepts regarding time slots in the SCRP with flexible service policies. One concept is time windows. A time window refers to an appointment time window in the vehicle booking system (or truck appointment system). The length of the appointment time window is pre-specified and determined at the strategic planning level. The trucks need to book time windows from the truck appointment system and it assumes that their arrival times at the container terminal are within their booked time windows. The other concept is sub-time windows. A terminal operator may divide an appointment time window into multiple time periods at the operational planning level. Each of such time periods can be defined as a sub-time window. The introduction of sub-time windows can facilitate the terminal operator to apply the flexible service policy. For example, only the trucks that arrive within the same specific sub-time window can be served out of sequence. The benefit of narrowing the scope of out-of-order services is to mitigate the service unfairness perceived by truck companies. The terminal operator needs to make operational decisions on the number of sub-time windows in order to achieve the desired trade-off between the relocation efficiency and the service fairness. Another measure to mitigate the service unfairness is embedding the service fairness objective into the relocation minimization procedure so that not only the number of relocations will be minimized but also the service fairness could be improved. This paper extends the model of the SCRP with the Flexible Service policies (SCRP-FS) in Feng et al. (2020) to the case with multiple sub-time windows, and our problem is termed the *SCRP with Multiple sub-time windows-based Flexible Service policy* or *SCRP-MFS*.

Another concern in the SCRP regards the randomness of truck arrivals. Knowing and utilizing the information of customer preference on different sub-time windows may help to reduce the number of relocations, and different scenarios of customer preference information may generate different results. However, in reality, the information about customer preference can be uncertain or may not be available to terminal operators. Therefore, the question remains as to whether it is worthwhile to commit efforts to gather and utilize the customer preference information. In this paper, we will consider various scenarios and evaluate the impacts and values of the customer preference information on the number of relocations.

In summary, this paper aims to achieve the following objectives: (i) to generalize the SCRP-FS into the case with multiple sub-time windows and incorporate the measure for service fairness (i.e., the maximum turnaround time among trucks); (ii) to analyse the impacts of the number of sub-time windows on the system overall efficiency (represented by the expected total number of relocations and the average truck turnaround time) and the service quality to individual trucks (represented by the maximum value and the coefficient of variation of truck turnaround times); (iii) to investigate how customer preference on different sub-time windows impacts the number of relocations (iv) and to evaluate the value of the information of customer preference in reducing the number of relocations in the SCRP-MFS.

This paper makes incremental contributions to the existing literature by incorporating fairness metrics into the optimization of the flexible service policy in the SCRP. It sheds new light on some important issues in the SCRP that applies flexible service policies. This paper can better inform container retrieval policies and improve container retrieval performance from these aspects: (i) demonstrating how the overall efficiency of container retrieval and the quality of service to individual trucks are traded off under different levels of service flexibilities; providing practical insights on how the policies vary with service fairness considerations; all of these can help the terminal operators to determine the appropriate number of sub-time windows in the flexible service policy; (ii) having a more comprehensive understanding of whether the problem is sensitive to customer preference, which can help improve container retrieval efficiency by managing the truck arrival patterns; (iii) and gaining an insight into the value of customer preference information, which can inform decision makers on whether to commit resources to gather such information in order to improve the container retrieval performance.

The rest of the paper is structured as follows. We review the literature regarding the (S)CRP and discuss the service fairness and value of information in Section 2. Section 3 formulates the SCRP-MFS by stochastic dynamic programming. The problem is solved by a heuristic algorithm in Section 4. The computational experiments are reported in Section 5. Finally, we highlight the main findings, generate managerial insights and envisage future research directions.

2 Literature review

We organize the literature into three parts. Section 2.1 reviews the studies on Container Relocation Problem (CRP) and stochastic CRP. Section 2.2 focuses on the service policies and service fairness associated with CRP and SCRP. Section 2.3 discusses the value of information to container handling operations in the yard.

2.1 Container relocation problems

The CRP may be formally defined below: suppose a set of containers are stored in a yard bay and the retrieval priorities among the containers are given, the objective is to empty the bay by minimizing the number of relocations (Ku and Arthanari, 2016b). The CRP needs to determine where to stack the relocated containers. In the standard CRP, the problem setting is static and restricted in the sense that it assumes that i) no containers are coming to the storage stacks in the container retrieval process and ii) relocation is allowed only when the container to be relocated blocks a target container. The static and restricted CRP has been the focus of attention of many researchers. In this paper, we also confine the problem under consideration to this version. With the relaxations of each of the above assumptions, there are two variants of the CRP, i.e., the dynamic CRP and the unrestricted CRP. The interested readers are referred to the works by Wan et al. (2009) and Hakan Akyüz and Lee (2014) for the dynamic CRP and the works by Jin et al. (2015) and Tanaka and Mizuno (2018) for the unrestricted CRP to get in-depth knowledge.

More complicated variants of the CRP come in the form of different settings of the container retrieval priorities. Regarding the retrieval priority, two types of assumptions have been traditionally made (Jang et al., 2013): 1) unique or group retrieval priorities; and 2) deterministic or uncertain retrieval priorities. Table 1 summarizes the literature on the CRP that is most relevant to this paper by these two types of assumptions.

Under the first type of assumptions, it is often assumed that each container has a unique retrieval priority and is regarded as a single group by itself. Otherwise, it may be assumed that grouped containers have the same priorities. In the latter case, the retrieval sequence for the containers with the same retrieval priority is to be determined. Practical examples and academic research of the latter case are as follows. For export containers, their loading precedence is specified by the precedence relationship among clusters of empty slots in a vessel. A cluster of slots is characterised by the attributes of containers, for example, container weight class, destination port and type. When a container with specified attributes is requested for loading into a slot, any container with the same set of attributes can be loaded into that slot (Kang et al., 2006). In Kim and Hong (2006), the retrieval precedence among groups of containers is given in the CRP, and the total number of relocations is minimized by optimizing both the container retrieval sequence and the positions of the relocated containers. As another example, for import containers, when multiple containers are destined for the same consignee or are to be transported by the trucks from the same shipping company, their pickup sequence is not important and thus can be in any order (Jang et al., 2013). In de Melo da Silva et al. (2018), it assumes that the containers in the same group are retrieved by the same customer. The container retrieval sequence and relocation positions are optimized simultaneously to minimize the number of relocations in the initial target group in the Block Retrieval Problem (BRP), and a secondary objective is included to minimize the expected number of relocations for the next group in the Bi-objective Block Retrieval Problem (2BRTP).

The second type of assumptions regards whether the retrieval priorities of different groups of containers are deterministic or uncertain. In reality, the retrieval order of import containers usually depends on the truck arrival order, hence, it is highly uncertain because of the unpredictability of external truck arrivals. While most of the studies on the CRP assume containers are sequenced with a certain prescribed order, several researchers have addressed the problem by considering uncertain retrieval priorities. The uncertain CRPs can be further categorized into two sub-categories (Feng et al., 2020): the online setting and the probabilistic setting. In the online setting, the container retrieval order is revealed gradually over time, and only the future retrievals within a limited look-ahead horizon are known. The goal is to design efficient online heuristics determining the relocation positions using information updated in real-time (e.g., Zehendner et al., 2017; Zhao and Goodchild, 2010). In the probabilistic setting, probability distributions are used to model the containers' retrieval priorities and the model objective is to minimize relevant expected performance measures, such as the expected total number of relocations (Ku and Arthanari, 2016a; Galle et al., 2018; Feng et al., 2020; Tong et al., 2015) and the weighted sum of the expected number of relocations and total retrieval delays (Borjian et al., 2013). For a more detailed review of the uncertain CRPs, we refer the readers to Feng et al. (2020). Here, we would like to note that the uncertain CRP in which groups of containers are ordered by the booked time windows and the trucks arrive at the booked time windows randomly is referred to as the Stochastic Container Relocation Problem (SCRP) in the literature (Galle et al., 2018; Feng et al., 2020). The problem considered in this paper belongs to the SCRP.

Methodologically, the CRPs have been usually modelled using (mixed) integer programming models (e.g., Wan et al., 2009; Tang et al., 2015; Zehendner et al., 2015) or (stochastic) dynamic programming models (e.g., Ku and Arthanari, 2016a; Kim and Hong, 2006; Galle et al., 2018; Feng et al., 2020). To seek optimal solutions, search-based algorithms are mainly used: branch and bound (Expósito-Izquierdo et al., 2015; Kim and Hong, 2006; Tanaka and Takii, 2015), (iterative deepening) A* algorithms (Quispe et al., 2018; Zhu et al., 2012), branch and price (Zehendner and Feillet, 2014) and branch and cut (Bacci et al., 2020). As the CRP has proven to be NP-hard (Caserta et al., 2012), only small-scale instances are able to be solved exactly within reasonable times. As a result, researchers often turn to heuristic approaches to overcome the computational complexities, such as index-based heuristics (Caserta et al., 2012; Hakan Akyüz and Lee, 2014), beam search (Bacci et al., 2019; Ting and Wu, 2017), metaheuristics (Maglić et al., 2020), and other greedy heuristics (Jovanovic and Voß, 2014; Jin et al., 2015) (c.f. Caserta et al. (2020) and the references therein). Recently, a new trend of the solution approach has been developed by using machine learning techniques (Zhang et al., 2020) and reinforcement learning (Jiang et al., 2021).

Finally, it should be noted that there are several related problems regarding container relocation, such as storage space allocation (e.g., Zhou et al., 2020) and storage location assignment (e.g., Feng et al., 2022) that focus on the initial stacking of containers into the yard, and container marshalling operation that focuses on re-arranging the positions of the stacked containers before they leave the yard (e.g., Parreño-Torres et al., 2020). In addition, in seaport rail terminals, prestaging operation is performed to pre-move the containers from the storage yard to local storage areas near the train before the train arrives in order to meet the departure deadline of trains (Xie and Song, 2018).

2.2 Service policies and service fairness

A service policy for CRP is to determine how the arriving trucks are served at the container yard. The service policy plays an important role in improving the efficiency of the import container retrieval process. The most commonly used service policy is the first-come-first-serve (FCFS) rule for trucks arriving at the yard. However, the FCFS policy does not lead to the most efficient container retrievals. A few studies have proposed flexible service policies for CRP, which have been proved to be more effective than the FCFS policy. The term "flexible service policy" refers to a service policy that allows the arriving trucks to be served out-of-order in contrast to the FCFS policy. It was coined by Borjian et al. (2015) and then followed by Feng et al. (2020). When a group of containers have the same retrieval priority, i.e. they are exchangeable in terms of retrieval sequence during the retrieval process (Kim and Hong 2006; de Melo da Silva et al. 2018), the out-of-order service policy is implicitly accepted by customers, and no concern of service fairness is raised. On the other hand, when containers are not exchangeable, a flexible service policy can cause extra waiting times for some customers and may raise the issue of service unfairness. The service unfairness herein refers to the difference in the waiting times among trucks or customers when out-of-order service is implemented. For example, under the truck appointment system, a truck books a time window and the containers to be retrieved, therefore, each container is bound with a specific truck. Namely, even though multiple containers have been booked in the same specific time window, these containers are not exchangeable as they are requested by different external trucks. In this case, retrieving the containers in an order different from the truck arrival order will influence the waiting time of individual customers (i.e., trucks). It may be perceived to be unfair if a truck experiences out-of-order retrievals and gets served later than expected. The problem under consideration in this paper falls into this category. In a couple of papers (Zeng et al., 2019; Zhao and Goodchild, 2010), the range of out-of-order retrievals is defined to be a group of containers that have been booked in the same time window, but the optimization objectives mainly focus on the number of relocations. In Borjian et al. (2013), a maximum service delay is set for each container, and the objective function is to minimize the weighted sum of the expected number of relocations and total delays in retrievals. Recently, Azab and Morita (2022a) introduce a new problem — the Block Relocation Problem with Appointment Scheduling (BRPAS) — aiming at improving import container relocation operations by coordinating with appointment scheduling. It assumes that the terminal operator can reschedule the container pickup times requested in the appointment system and determine the final appointments and the pickup order for each container. An appointment shift allowance is introduced to control the gap between the requested pickup times and the allocated appointment times. The FCFS policy is applied to the containers in the same subgroup to reduce truck waiting times.

As far as we know, no studies have explicitly treated service fairness as part of objective functions to be optimized in the CRP and SCRP contexts. Only two papers (Borjian et al., 2015; Feng et al., 2020) have mentioned service equity (fairness) and evaluated the impact of the relocation-minimization-oriented solutions on the service equity in the CRP and SCRP. In Borjian et al. (2015), the level of service flexibility is controlled by specifying a maximum number of containers that are allowed to be retrieved out-of-order before each truck. The research concludes that the flexible retrieval planning can decrease the total number of relocations and trucks average waiting times while maintaining service equity for each truck in the long term. However, they study the CRP in a deterministic setting and assume that the exact arrival time of the external trucks is known. Feng et al. (2020) investigate the SCRP that applies the two sub-time windows-based flexible service policy (termed the SCRP-FS). The trucks with the same booked time window are referred to as a batch. A batch of trucks arrive at its booked time window randomly and their exact arrival order is revealed batch by batch. They propose two objectives, one is minimizing the expected total number of relocations (primary objective) and the other one is minimizing the total truck waiting times in each retrieval batch (secondary objective). However, the service fairness is neither optimized nor balanced against the number of relocations.

The fairness concerns in a general sense have been emphasized in the operations research and management science literature and across multiple industrial applications, e.g. urban transportation systems (Li et al., 2019; Wu et al., 2021), vehicle routing problems (Matl et al., 2018), healthcare appointment systems (Qi, 2017), resource allocation problems (Bertsimas et al., 2011, 2013), and vessel scheduling at ports (Zhang et al., 2017; Wang et al., 2017; Jia et al., 2022). There are two types of performance indicators that are commonly used to measure service fairness in the OR literature (Jia et al., 2022). One is the min-max metric that minimizes the maximum disutility of the players in the system; the other is the min-difference metric that minimizes the difference between the disutilities of any two players. The characteristics of our problem are that the arriving trucks are booked on specific time windows and the service unfairness arises from the trucks that are booked on the same time window. The service fairness is optimized by using the min-max metric that minimizes the maximum truck turnaround time. The min-max metric has been used in some application-specific OR literature (e.g., Li et al., 2019; Wu et al., 2021; Teye and Bell, 2016). In addition, we also evaluate the service fairness by another metric, the coefficient of variation of the truck turnaround time, to show the extent of variability of the turnaround time in relation to its mean.

2.3 Value of information

Relevant information could reduce the degree of uncertainty and improve yard operation performance (Zuidwijk and Veenstra, 2015). Utilizing information on truck arrival times gathered through the truck appointment system, a few studies have examined its impact on the number of relocations required in the container retrieval process. Zhao and Goodchild (2010) develop two heuristics to address the CRP with time windows where containers are grouped and ordered by the trucks' arrival time windows and new information on the exact truck arrival order is updated one truck at a time; this model is defined as the online model by Galle et al. (2018). Their findings show that significant reductions in relocations can be achieved by just knowing in which groups a truck will arrive. Ku and Arthanari (2016a) also address the CRP in the context of the online model, but they formulate the problem into a stochastic dynamic programming model and solve it optimally using a decision tree scheme. Different from using the online model, Galle et al. (2018) consider the batch model in which the exact truck arrival order is updated a batch (group) at a time and formulate it as a multi-stage optimization problem. They prove the value of taking into account the "within batch" information (arrival sequence of the external trucks in a batch) in reducing the expected number of relocations both theoretically and numerically.

Some studies utilize the arrival information of trucks or vessels to determine the stacking positions of containers in a storage yard in order to improve the operational efficiency during the container retrieval process. van Asperen et al. (2013) investigate how the information of truck announcement time impacts the performance of online container stacking operations; their findings show that an average announcement time of 0.5-24 hours can significantly improve stacking efficiency. Gharehgozli and Zaerpour (2018) propose a shared stacking policy that utilizes the information on the arrival time windows of barges to determine the stacking locations of outbound containers; compared to the practical staking policy, this policy proves to reduce up to 30% of the total retrieval time. In addition, the information related to truck arrivals has also been utilized in container (p)re-marshalling operations to rearrange the container stacking positions in order to improve future container retrieval efficiency. Covic (2017) introduce an online rule-based solution method for container re-marshalling by taking the truck arrival information into consideration. The results show that imprecise arrival information of trucks, not deviating above a certain threshold, can significantly reduce truck waiting time. Kim and Yi (2021) develop heuristic algorithms to locate and pre-marshal import containers by utilizing various pieces of information associated with truck arrivals (e.g. truck appointment, truck dispatching notice, container dwell time distribution, and external trucks' real-time positions). The results show that when all these sources of information are utilized, the truck system time and the number of relocations during a container retrieval can be reduced by 47% and 98%, respectively. Recently, Feng et al. (2022) proposes a smart stacking strategy for import containers by making use of containers' customer identity information, where the containers that are destined for the same customers are stacked in the same piles to reduce the need of relocations when they are collected in the future. It is shown that the smart stacking strategy can improve the operational efficiency significantly compared to random stacking.

Different from the aforementioned studies, this paper is to evaluate the value of customer preference information in terms of the reduction of the expected number of relocations in the SCRP that applies flexible service policies. The customer preference information is measured as the probabilities of an external truck arriving in each sub-time window within its booked time window. The motivation is based on the fact that the specific probability distribution of the arrival sequence of the external trucks in a booked time window is hard to predict. The existing studies on the SCRP (Galle et al., 2018; Ku and Arthanari, 2016a) simply assume that the probabilities of any possible arrival orders of the trucks booked in an appointment time are the same. Such an assumption may not capture the customer (i.e., truck) behaviours adequately as the customers may have their preferred arrival segments within the booked time window.

There have been studies about shifting trucks' arrival times from their preferred arrival times to reduce truck congestion in port areas (e.g., Chen et al., 2013a, b; Phan and Kim, 2015) and to reduce relocations (Azab and Morita, 2022a,b). In a recent study on the SCRP-FS, Feng et al. (2020) consider unequal probabilities of the truck arrival order that vary with the customer's preference for each sub-time window of the booked time window. It shows that the container retrieval operational efficiencies under different customer preference scenarios vary due to different needs of relocations, however, whether utilizing the customer preference information can improve the operational efficiency and the extent of the improvement has not been investigated.

2.4 Research gap

Table 1 summarizes the comparisons of this paper with the most relevant literature from four aspects. Only two papers discuss the issue of service fairness (equity) among trucks in the CRP. The first paper, Borjian et al. (2015), considers the service fairness in a deterministic context and focuses on evaluating the number of trucks that are served out of sequence before each external truck. The second paper, Feng et al. (2020), studies the CRP in a probabilistic setting and evaluates the impacts of out-of-order retrievals on individual trucks in the SCRP-FS by the maximum truck turnaround time, but their model is limited to a special case in which each appointment time window consists of two sub-time windows. More importantly, neither of the studies optimize the service fairness, nor have they explicitly addressed the trade-off decision between the expected total number of relocations and the service fairness of all trucks. This paper attempts to fill these gaps. Specifically, we generalize the SCRP-FS to the case with multiple sub-time windows, termed SCRP-MFS, and incorporate the service fairness into objective functions. Through such generalization, we can control the level of service flexibility more accurately, making the mathematical model more relevant to reality. Applying multiple sub-time windows is beneficial to reduce the maximum truck turnaround time and the coefficient of variation of the turnaround times among different trucks (see our experiments' result in Section 5.1), which can both reflect the service fairness. It is worth noticing that shortening the appointment time window length is not equivalent to dividing a time window into multiple sub-windows, from the aspects of both contractual relationships and operational performances. First, a shorter appointment time window represents a much stricter contractual relationship between the terminal operator and the truck company than a longer time window because truck companies are more likely to be penalized for late-show. Second, we have conducted some preliminary experiments, which show that the computational results between scenario "A" where a 30 minutes time window is divided into 2 sub-time windows and scenario "B" where a 60 minutes time window is divided into 4 sub-time windows could be very different. Therefore, considering multiple sub-time windows is a new problem and deserves in-depth research.

In addition, regarding the probabilistic model of truck arrivals, only Feng et al. (2020) has modelled the truck arrival probabilities by customer preference based on two sub-time windows. But trucks arriving at multiple sub-time windows remains an unresearched question in the CRP discipline. Besides, the value added by the customer information in reducing the number of relocations has not been assessed, and it remains unclear as to whether it is worthwhile to commit resources to gather and utilize such information. This research also aims to fill these two gaps, by proposing a general probabilistic model of truck arrivals at multiple sub-time windows and evaluating the impacts and the value of customer preference information on the number of relocations.

Table 1 Classification of the most relevant literature on the CRP.

Literature	Unique or group priority	Certainty of priorities	Service policy	Performance metrics
Borjian et al. (2013)	Group	Deterministic; Uncertain	Flexible	WS of ENR and RD
Borjian et al. (2015)	Unique	Deterministic	Flexible	WS of NR and RD; ORT*
de Melo da Silva et al. (2018)	Group	Uncertain	Flexible	NR; ENR
Galle et al. (2018)	Unique	Uncertain	FCFS	ENR
Kim and Hong (2006)	Unique; Group	Deterministic	FCFS; Flexible	NR
Ku and Arthanari (2016a)	Unique	Uncertain	FCFS	ENR
Zeng et al. (2019)	Unique	Deterministic	Flexible	NR
Zhao and Goodchild (2010)	Unique	Deterministic; Uncertain	FCFS; Flexible	NR; CHTD*

Feng et al. (2020)	Unique	Uncertain	Flexible with 2-sub-windows	ENR; total TWT; maximum TTT*
This paper	Unique	Uncertain	Flexible with multi-sub-windows	ENR; maximum TTT; average and CoV of TTT*

NR: number of relocations; ENR: expected number of relocations; TWT: truck waiting times; TTT: truck turnaround times: RD: retrieval delays; CHTD: crane horizontal travel distance; WS: weighted sum; CoV: coefficient of variation; ORT: out-of-order retrievals performed before a truck is served.

Note: performance metrics without "*" represent optimized objectives, and those with "*" represent evaluated metrics.

Methodologically, this paper follows the frameworks of Feng et al. (2020), but there are several differences. Firstly, in the multiple sub-window problem (SCRP-MFS), the probabilistic model of truck arrivals is much more complicated, the number of decision variables is increased, and the stochastic dynamic programming formulation becomes more complicated. Therefore, our SCRP-MFS problem is more challenging than the SCRP-FS in Feng et al. (2020). Secondly, our model considers a different secondary objective that minimizes the maximum truck turnaround time. This objective function is commonly used to measure service fairness, which is an important performance indicator that has not been optimized in container relocation literature. Thirdly, the new objective function and the structure of the multiple sub-time windows necessitates several adaptions to the existing SEM heuristic algorithm: (i) a new procedure is embedded into the sequencing rule to optimize the maximum truck turnaround time; (ii) the calculation methods of the two key indexes in the relocation rule are generalized so that the two indexes can be applied to the situation of multiple sub-time windows. Fourthly, this study further explicitly evaluates the value of the customer preference information, which helps to identify which situations are more beneficial to gather and utilize customer preference information. To achieve that, we first make decisions under the assumption of no such information being utilized, and then we apply the decisions to the simulation environment where the truck arrival times are generated according to the customer preference information. Finally, we conduct statistical tests to validate the impact and value of the customer preference information.

3. The SCRP-MFS

This section first describes the SCRP-MFS and then introduces the probabilistic model of truck arrivals. Next, we present the problem formulation. Last, we analyze the model and develop the handling approach.

3.1 Problem description

Section 3.1.1 introduces the definitions and notations used in the SCRP-MFS, and Section 3.1.2 introduces the problem assumptions. A list of the essential modelling notations is provided in Appendix A.1.

3.1.1 Definitions and notations

A bay is composed of S stacks and T tiers and it accommodates $C = (S-1)T+1$ containers. Each container corresponds to a truck. A truck books a time window to retrieve its container and will reach the terminal within its booked time window. Furthermore, the trucks (and the corresponding containers) are grouped into a set of batches and sub-batches:

(1) A ***batch*** of trucks (containers) refers to a set of trucks (containers) that have booked the same time window. B_k and C_k respectively denotes the set of containers and the number of containers in batch k, $k \in \{1,...,K\}$.

(2) We know the arriving sequence of the truck batches, but the exact arriving sequence of the trucks in each individual batch is uncertain, which will be revealed gradually as the container retrieval operations proceed.

(3) Each appointment time window is split into W ($W \in \mathbb{Z}^+$) sub-time windows with identical time lengths. A ***sub-batch*** of trucks (containers) refers to a subset of trucks (containers) that have arrived (whose trucks have arrived) within the same sub-time window $w \in \{1,...,W\}$.

(4) The flexible service policy is applied, that is, the containers/trucks in the same sub-batch can be retrieved/served out of sequence.

Each container has three attributes: priority label, unique ID, and customer preference:

(1) The ***priority label***, denoted by l_i, $i \in \{1,...,C\}$, traces the retrieval priorities among containers. The priority label is updated as the retrieval operation proceeds. In the beginning, containers in batch k are labeled by a *batch priority*, denoted by L_k, which represents the arriving precedence of truck batches (Fig. 1(a)). Define $L_k = 1 + \sum_{j=1}^{k-1} C_j$, which implies that for any given L_k, a unique $k \in \{1,...,K\}$ can be determined. Then, once the arrival order of the trucks in batch k becomes known, the priority labels of the containers in batch k are updated to the *sub-batch priority* that indicates the arrival precedence among sub-batches of trucks (see Fig. 1(c)). Specifically, the priority label of a truck in batch k that is revealed to arrive in sub-time window w is updated to $L_k + \sum_{w'=1}^{w-1} n_{kw'}$, where $n_{kw'}$ represents the number of external trucks in batch k that have arrived within sub-time window w'. Finally, once the retrieval sequence of the containers in batch k is determined (within $[L_k, L_k + C_k - 1]$), their labels are updated to reflect the determined retrieval sequence.

(2) The ***unique ID***, denoted by u_i, $i \in \{1,...,C\}$, differentiates individual containers/trucks (Fig. 1(b)).

(3) The ***customer preference***, denoted by $p_{i,w}^W$, $i \in \{1,...,C\}$, $w \in \{1,...,W\}$, represents the probability of truck u_i arriving within sub-time window w of its appointed time window (Fig. 1(d)). Each appointment time window is divided into W sub-time windows equally. The values of $p_{i,w}^W$ can be derived from the truck appointment system that requires trucks to provide their preferences when booking a window; alternatively, the values of $p_{i,w}^W$ can be estimated from historical data as the proportion of truck u_i arriving at sub-time window w.

3.1.2 Assumptions

In the SCRP-MFS, we have the following assumptions.

A1: The blocking containers are relocated to the stacks only within a single bay.

A2: A container can be relocated only when it is located above the target container.

A3: No containers will be added to the bay during the entire retrieval process.

A4: (information revealing time) The truck arrival sequence is revealed batch by batch. After all containers in a batch have been collected, the arrival sequence of the next batch of trucks becomes known.

A5: (service beginning time) The service beginning time of a batch is at the end of the appointed time window of the batch.

A6: (probabilities of truck arrivals) (1) The probability distribution of a truck's arrival over sub-time windows depends on customer preference, and (2) the trucks' arrival sequence in a sub-batch follows uniform distribution.

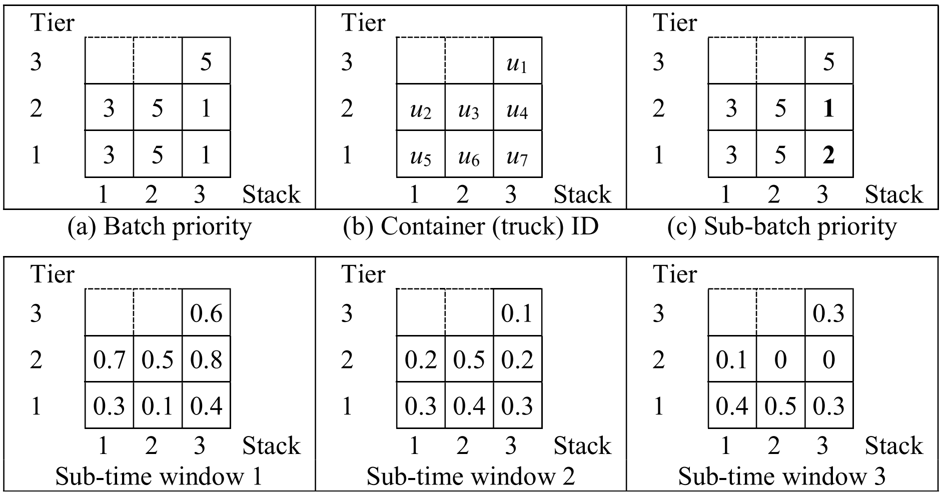
A1 to A3 are generic to the standard CRP. A4 and A5 ensure that the arrival sequence of the trucks in a batch has been revealed before the start of the retrieval service of the batch. A6 is about the probabilistic distribution of external truck arrivals, which will be explained in section 3.2.

3.2 General probabilistic model of truck arrivals

The arrival order of the trucks in a batch is stochastic. We characterise the truck arrival order in the same way as Feng et al. (2020), who models it by customer preference. The customer preference refers to a truck's arrival probabilities for each sub-time window of its booked time window. In this paper, we propose a general probabilistic model representing truck arrivals. The model extends the two sub-time windows model in Feng et al. (2020) to multiple sub-time windows.

Each time window is split into W sub-time windows with the same time lengths. Let ζ_k^W, $k \in \{1,...,K\}$, refer to a random set of scenarios of which trucks in batch k arrive at each of the W sub-time windows of the appointed time window, and let $p(\zeta_k^W)$ refer to its probability. $\zeta_k^W = \left\{\xi_{k,w}^W \mid w \in \{1,...,W\}\right\}$, where $\xi_{k,w}^W$ is a random set of the trucks in batch k that arrive in sub-time window w. $\xi_{k,w}^{W}$, $w \in \{1,...,W\}$, are mutually exclusive and collectively exhaustive, that is, $\bigcup_{w\in\{1,...,W\}} \xi_{k,w}^{W} = B_k$ and $\bigcap_{w\in\{1,...,W\}} \xi_{k,w}^{W} = \varnothing$. The random variables in $\xi_{k,w}^{W}$ take values in B_k, and $\{u_i \in \xi_{k,w}^{W}\}$ represents the event that truck u_i arrives in the sub-time window w. By definition, $p_{i,w}^{W} = P(u_i \in \xi_{k,w}^{W})$, where $u_i \in B_k$ and $w \in \{1,...,W\}$. Let $\zeta^{W} = \bigcup_{k\in\{1,...,K\}} \zeta_k^{W}$ denote the random set of truck arrival scenarios for all the batches. There are W^{C_k} scenarios for ζ_k^{W} and $\prod_{k=1}^{K} W^{C_k}$ scenarios for ζ^{W}.

The probabilities of ζ_k^{W} can be calculated by $p_{i,w}^{W}$. Fig. 1 gives an example to illustrate the attributes of containers and explain the probabilistic model. In this example, $W = 3$. Fig. 1(a), (b) and (d) constitute the initial bay configuration. In Fig. 1(c), the bold numbers represent the sub-batch priority of the first batch that has been revealed. Fig. 1(d) shows the customer preference of each container/truck for each sub-time window, i.e., $p_{i,w}^{W}$. Now we explain the general probabilistic model of truck arrivals by taking the example of ζ_1^3. ζ_1^3 has nine scenarios, which are respectively $\{\xi_{1,1}^3=\{u_4,u_7\},\xi_{1,2}^3=\varnothing,\xi_{1,3}^3=\varnothing\}$, $\{\xi_{1,1}^3=\varnothing,\xi_{1,2}^3=\{u_4,u_7\},\xi_{1,3}^3=\varnothing\}$, $\{\xi_{1,1}^3=\varnothing,\xi_{1,2}^3=\varnothing,\xi_{1,3}^3=\{u_4,u_7\}\}$, $\{\xi_{1,1}^3=\{u_4\},\xi_{1,2}^3=\{u_7\},\xi_{1,3}^3=\varnothing\}$, $\{\xi_{1,1}^3=\{u_4\},\xi_{1,2}^3=\varnothing,\xi_{1,3}^3=\{u_7\}\}$, $\{\xi_{1,1}^3=\varnothing,\xi_{1,2}^3=\{u_4\},\xi_{1,3}^3=\{u_7\}\}$, $\{\xi_{1,1}^3=\{u_7\},\xi_{1,2}^3=\{u_4\},\xi_{1,3}^3=\varnothing\}$, $\{\xi_{1,1}^3=\{u_7\},\xi_{1,2}^3=\varnothing,\xi_{1,3}^3=\{u_4\}\}$, $\{\xi_{1,1}^3=\varnothing,\xi_{1,2}^3=\{u_7\},\xi_{1,3}^3=\{u_4\}\}$. The probabilities of these scenarios are computed as: $p\left(\{\xi_{1,1}^3=\{u_4,u_7\},\xi_{1,2}^3=\varnothing,\xi_{1,3}^3=\varnothing\}\right)=0.8\times0.4=0.32$; $p\left(\{\xi_{1,1}^3=\varnothing,\xi_{1,2}^3=\{u_4,u_7\},\xi_{1,3}^3=\varnothing\}\right)=0.2\times0.3=0.06$; $p\left(\{\xi_{1,1}^3=\varnothing,\xi_{1,2}^3=\varnothing,\xi_{1,3}^3=\{u_4,u_7\}\}\right)=0\times0.3=0$; $p\left(\{\xi_{1,1}^3=\{u_4\},\xi_{1,2}^3=\{u_7\},\xi_{1,3}^3=\varnothing\}\right)=0.8\times0.3=0.24$; $p\left(\{\xi_{1,1}^3=\{u_4\},\xi_{1,2}^3=\varnothing,\xi_{1,3}^3=\{u_7\}\}\right)=0.8\times0.3=0.24$; $p\left(\{\xi_{1,1}^3=\varnothing,\xi_{1,2}^3=\{u_4\},\xi_{1,3}^3=\{u_7\}\}\right)=0.2\times0.3=0.06$; $p\left(\{\xi_{1,1}^3=\{u_7\},\xi_{1,2}^3=\{u_4\},\xi_{1,3}^3=\varnothing\}\right)=0.4\times0.2=0.08$; $p\left(\{\xi_{1,1}^3=\{u_7\},\xi_{1,2}^3=\varnothing,\xi_{1,3}^3=\{u_4\}\}\right)=0.4\times0=0$; $p\left(\{\xi_{1,1}^3=\varnothing,\xi_{1,2}^3=\{u_7\},\xi_{1,3}^3=\{u_4\}\}\right)=0.3\times0=0$. According to the real truck arrival order, ζ_1^3 will be revealed to be one of these nine scenarios. For example, if ζ_1^3 is revealed to be $\{\xi_{1,1}^3=\{u_4\},\xi_{1,2}^3=\{u_7\},\xi_{1,3}^3=\varnothing\}$, $\{\xi_{1,1}^3=\{u_4\},\xi_{1,2}^3=\varnothing,\xi_{1,3}^3=\{u_7\}\}$, or $\{\xi_{1,1}^3=\varnothing,\xi_{1,2}^3=\{u_4\},\xi_{1,3}^3=\{u_7\}\}$, the sub-batch priority for the first batch will be updated to be that shown in Fig. 1(c), which indicates that truck u_4 and u_7 arrive in different sub-time windows and u_4 arrives before u_7.



(d) Customer preference

Fig. 1 An example for illustrating containers' attributes

3.3 Problem formulation

The SCRP-MFS proposed in this research is a multi-stage sequential decision-making problem where information is revealed dynamically. The truck arrival information is revealed batch by batch (i.e., stage). At each stage, decisions are made based on the revealed information. Mathematically, it can be formulated as a stochastic dynamic programming model (SDP) similar to the Sooo model in Feng et al. (2020). However, the two models differ in several ways. While both models consider random truck arrivals in a time window, SCRP-MFS models the arrival in multiple sub-time windows rather than in two sub-time windows as considered in the Sooo model. Correspondingly, SCRP-MFS determines the retrieval sequence of the containers in multiple sub-batches, as opposed to the retrieval sequence in two sub-batches in the Sooo model. Last, the objective functions are different. This study introduces a secondary objective function measuring service fairness, which has been rarely considered in the CRP literature. In the following, we first introduce the objective functions of the SCRP-MFS. Then, we present the formulation regarding each objective.

3.3.1 Objective functions

The main subjects involved in the SCRP-MFS are terminal operators and trucks. We formulate the SCRP-MFS as a multi-objective optimization problem aiming to improve the relocation efficiency from the terminal operators' perspective and meanwhile to mitigate the unfairness among trucks from the perspective of the worst-off trucks. We adopt two lexicographically ordered objectives. The first objective is primary aiming to minimize the expected total number of relocations, which reflects the relocation efficiency. The secondary objective aims to minimize the maximum truck turnaround time, which reflects service fairness and is called min-max fairness in this paper. The min-max index is a well-studied objective function used to measure fairness in transportation systems (e.g., Li et al., 2019; Wu et al., 2021), the healthcare industry (e.g., Qi 2017) and other domains of operational research (see the references in Yang et al., 2013). It has also been used in berth allocation problems in container terminals to minimize the maximum lateness of any vessel where each vessel has a due departure time (Teye and Bell, 2016). The adoption of lexicographically ordered objectives indicates that we still take the relocation efficiency as our principal aim and the fairness concern is considered under the promise that the number of relocations is minimized. This is because the number of relocations has a direct effect on the truck turnaround time. The minimization of the number of relocations is beneficial for reducing the truck turnaround time, but multiple solutions that have the same minimal number of relocations can result in different maximum truck turnaround times. Next, we introduce the formulations regarding the primary objective and secondary objective respectively.

3.3.2 Primary objective for relocation efficiency

The three key elements of the SDP model, that is, stage, state, and action, are defined as follows.

Stage: the sequence number of the batch to be retrieved. A stage refers to a batch.

State: the state of a stage is the state of the bay of the stage, including the stacking positions of the remaining containers and the attributes of these containers.

Action: a feasible action is defined as a sequence of moves to retrieve a batch of containers, which is made up of two types of actions: (i) *sequencing* - the retrieval/service sequences of the containers/trucks in each sub-batch, and (ii) *relocating* - the stacking positions of the relocated containers.

The following notations are defined and used in the SDP model. These notations are also included in Appendix A.1.

W: the number of sub-time windows in an appointment time window (a decision variable).

K: the total number of batches/stages.

k: the stage number, $k \in \{1,...,K\}$, and stage k refers to the kth batch to be retrieved.

ζ_k^W : the set of scenarios of the sub-batches of stage k, $k \in \{1,...,K\}$ (a random variable).

S_k : the input state of stage k, $k \in \{1,...,K\}$.

$p(\zeta_k^W)$: The probability of ζ_k^W. $p(\zeta_k^W)$ is calculated by the general probabilistic model of truck arrivals presented in section 3.2.

$\mathbf{a}_k(S_k,\zeta_k^W)$: The actions (a decision variable) taken for retrieving the kth batches of containers given S_k and ζ_k^W. $\mathbf{a}_k(S_k,\zeta_k^W)=\{\mathbf{a}_k^S(S_k,\zeta_k^W),\mathbf{a}_k^R(S_k,\zeta_k^W)\}$, wherein $\mathbf{a}_k^S(S_k,\zeta_k^W)$ represents the retrieval sequence for the containers in each sub-batch at stage k given S_k and ζ_k^W, and $\mathbf{a}_k^R(S_k,\zeta_k^W)$ represents the relocation positions that respect $\mathbf{a}_k^S(S_k,\zeta_k^W)$. For notational convenience, we omit the dependence on (S_k,ζ_k^W) from $\mathbf{a}_k(S_k,\zeta_k^W)$ and use $\mathbf{a}_k$ instead.

$r_k(\mathbf{a}_k \mid S_k,\zeta_k^W)$: The number of relocations during action $\mathbf{a}_k$ on the bay of state S_k given ζ_k^W.

$t_k(S_k,\zeta_k^W,\mathbf{a}_k)$: The state transition function that maps S_k, ζ_k^W, and $\mathbf{a}_k$ into the next state S_{k+1}. By $t_k(S_k,\zeta_k^W,\mathbf{a}_k)$, the kth batch of containers revealed by ζ_k^W are retrieved from state S_k according to action $\mathbf{a}_k$, after which S_{k+1} is obtained.

$f_k^W(S_k)$: The minimum expected total number of relocations to retrieve the remaining K-k+1 batches of containers from state S_k when each appointment time window is divided into W sub-time windows.

The primary objective is formulated as follows, which is a recursive function representing the minimum expected total number of relocations:

$$f_1^W(S_1)=E\left[\min_{\mathbf{a}_1}\left\{r_1(\mathbf{a}_1 \mid S_1,\zeta_1^W)+f_2^W(S_2)\right\}\right]=\sum_{\zeta_1^W} p(\zeta_1^W)\min_{\mathbf{a}_1}\left[r_1(\mathbf{a}_1 \mid S_1,\zeta_1^W)+f_2^W(S_2)\right],$$

where $$S_2=t_1(S_1,\zeta_1^W,\mathbf{a}_1) \quad (1)$$

Generally, the problem for any stage k can be formulated as follows:

$$f_k^W(S_k)=\sum_{\zeta_k^W} p(\zeta_k^W)\min_{\mathbf{a}_k}\left[r_k(\mathbf{a}_k \mid S_k,\zeta_k^W)+f_{k+1}^W(S_{k+1})\right],\ k\in\{1,\dots,K\},$$

where $$S_{k+1}=t_k(S_k,\zeta_k^W,\mathbf{a}_k),\ \text{for } k\in\{1,...,K\},\ \text{and } f_{K+1}^W(S_{K+1})=0 \quad (2)$$

3.3.3 Secondary objective for service fairness

The secondary objective is to minimize the maximum turnaround time among all of the trucks. The *turnaround time* of truck i is defined as the difference between its arrival time $\tilde{a}_i$ and its departure time $\tilde{d}_i$, that is, the retrieval service completion time. Let $g_{k,i}^{\zeta_k^W}=(\tilde{d}_i-\tilde{a}_i)$ denote the turnaround time of truck i in batch k ($i\in B_k$) under ζ_k^W ($\zeta_k^W \subset \zeta^W$). Then, the secondary objective function can be expressed as

$$\text{Min} \sum_{\zeta^W} p(\zeta^W) \max_{k\in\{1,\dots,K\}, i\in B_k} g_{k,i}^{\zeta_k^W} \tag{3}$$

Note that the secondary objective in (3) is optimized under the promise that the primary objective in (1) has achieved optimality, which means that the two objectives are optimized sequentially. Given the decisions $\mathbf{a}_k(S_k, \zeta_k^W)$ that are optimal regarding the primary objective, we now derive the explicit expression of $g_{k,i}^{\zeta_k^W}$ by using the following notations.

$o_i^{\zeta_k^W}$: the service order of truck *i*, $i \in B_k$, under ζ_k^W. $o_i^{\zeta_k^W}$ is implied in the retrieval sequence decision $\mathbf{a}_k^S(S_k, \zeta_k^W)$.

$r_i^{\zeta_k^W}$: the number of relocations performed for serving truck *i*, $i \in B_k$, under ζ_k^W. $r_i^{\zeta_k^W}$ is implied in the relocation decision $\mathbf{a}_k^R(S_k, \zeta_k^W)$.

t^{ret} : the time of per retrieval move.

t^{rel} : the time of per relocation move.

e_k: the end of the appointed time window of batch *k*.

s_k: the service starting time of batch *k*.

c_k: the time of completing the retrieval of the last container in batch *k*.

Given the decisions $\mathbf{a}_k(S_k, \zeta_k^W)$ of batch *k*, c_k and s_{k+1} can be obtained. According to A5, when beginning serving a batch, all the trucks in the batch have arrived at the yard stack, which means no idle time exists between serving any two trucks in the batch. Therefore, c_k is calculated by

$$c_k = s_k + \sum_{i\in B_k} (t^{rel} \cdot r_i^{\zeta_k^W} + t^{ret}) \tag{4}$$

Given e_k and c_{k-1}, according to A5, s_k is calculated by

$$s_k = \max\{e_k, c_{k-1}\},\ k \in \{2,\dots,K\};\ s_1 = e_1. \tag{5}$$

Given the above expressions, $g_{k,i}^{\zeta_k^W}$ is calculated by

$$g_{k,i}^{\zeta_k^W} = \tilde{d}_i - \tilde{a}_i = \left(s_k + \sum_{j\in B_k, o_j^{\zeta_k^W} < o_i^{\zeta_k^W}} (t^{rel} \cdot r_j^{\zeta_k^W} + t^{ret}) + t^{rel} \cdot r_i^{\zeta_k^W} + t^{ret} \right) - \tilde{a}_i, \tag{6}$$

3.4 Model analysis and handling approach

Let $\gamma^{(1)}$ and $\gamma^{(2)}$ denote the primary objective and the secondary objective, respectively. Then, the SCRP-MFS can be formulated as follows:

$$\min \gamma^{(1)} = f_1^W(S_1),\ f_1^W(S_1) \text{ is defined in Eq. (1)}$$

$$\min \gamma^{(2)} = \sum_{\zeta^W} p(\zeta^W) \max_{k\in\{1,\dots,K\}, i\in B_k} g_{k,i}^{\zeta_k^W},\quad g_{k,i}^{\zeta_k^W} \text{ is defined in Eqs. (4)-(6)} \tag{7}$$

There are two levels of decisions in the model. At the higher level, the number of sub-time windows *W* directly impacts the level of flexibility in optimizing the container retrieval sequence and thus the objective values. At the lower level, the container retrieval sequence $\mathbf{a}_k^S(S_k, \zeta_k^W)$ and the container relocation positions $\mathbf{a}_k^R(S_k, \zeta_k^W)$ are two-fold decision variables. We handle the model by a hierarchical iterative approach, under the framework presented in Fig. 2. At the outer hierarchy, we determine the number of sub-time windows, while at the inner hierarchy, we determine the container retrieval sequence and relocation positions. The two hierarchies are incorporated in an iterative process. At each iteration, we update the value of W, that is, the number of sub-time windows. Accordingly, we solve the model where W is treated as a parameter. To solve this model, the focus is to solve the SDP with the primary objective. Then, among the multiple solutions that optimize the primary objective, the one with the minimal secondary objective value can be selected as the optimal solution for the SCRP-MFS with a specific W. The optimal solutions of the SDP model can be obtained by optimizing the recursive equation (2) backwards from stage K to stage 1. However, solving the SDP model is very time-consuming for practical scale problems due to the curse of dimensionality (Feng et al., 2020). The case with multiple sub-time windows can be more time-consuming, as it creates an increased number of possible sub-batches and corresponding scenarios. To accomplish extensive experiments in a reasonable time, the model is solved by a heuristic algorithm. When the model is solved, the container retrieval sequence and the relocation positions are updated and the relevant performance measures are evaluated by simulation.

The idea behind the scheme of updating W is to increase the value of W if the maximum truck turnaround time is smaller than that in the previous iteration and to terminate the iteration process either if the maximum truck turnaround time does not improve for Q_1 consecutive iterations or if it reaches the iteration limit Q_2. The solution procedure is motivated by the following two observations.

Observation 1. The primary objective function (i.e. the expected total number of relocations) is increasing as the number of sub-windows increases.

Observation 2. The secondary objective function (i.e. the maximum truck turnaround time) is generally decreasing but not always decreasing as the number of sub-windows increases.

The rationality of the two observations is explained here. There is a trade-off between relocation efficiency and service fairness when optimizing the SCRP-MFS. There is no ultimate solution with the lowest number of relocations and the smallest maximum truck turnaround time. If W is set to be a small value, the solution will turn out to have a smaller number of relocations but a longer maximum truck turnaround time. On the contrary, if W is set to be a higher value, the solution will show the opposite feature. However, a larger value of W does not necessarily guarantee a shorter maximum truck turnaround time. This is because the turnaround time of a truck is also influenced by the number of relocations that are needed to retrieve the target container. Under a larger W, since the flexibility in optimizing the container retrieval sequence is very small, a greater number of relocations may be needed to retrieve a container, and thus it may cause longer turnaround times for some trucks. When such a point appears, there is not much need to continue increasing the value of W since the benefit of reducing the maximum truck turnaround time may be tiny. Initially, W is set to be a small value W_0 (e.g., 2) to allow a great extent of flexibility for optimising the container retrieval sequence. Then, by scanning W from small to large values, we are able to find a certain point that best balances the two objectives. However, it is the decision maker's choice to finally determine the value of W, which depends on whether the relocation efficiency or the service fairness is emphasized more by the terminal operators.

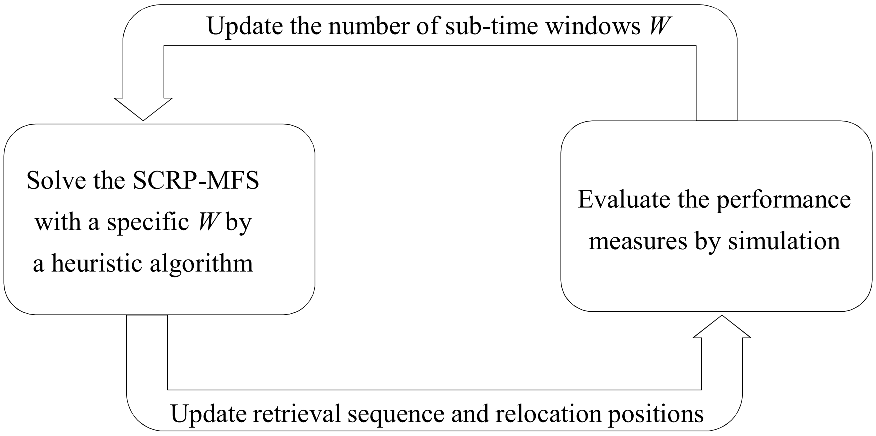


Fig. 2 Framework of the model handling approach

4. Heuristic algorithm

It has been computationally proved that the SEM (Sequencing based Expected Minmax) heuristic is able to obtain near-optimal solutions of the SCRP-FS (Feng et al., 2020). In this section, we extend the SEM heuristic so that it can be applied to solve the SCRP-MFS. One of the main extensions we made to the SEM heuristic is on generalizing the BIS index and DIS index, which are two key criteria used for selecting the relocation positions. The details of the two indexes are referred to in section 4.2.2. In the SEM heuristic, the calculation methods of these two indexes are only applicable to the situation of two sub-batches. When considering multiple sub-batches, a general calculation method is needed. In addition, we embed a procedure to reduce the maximum truck turnaround time.

Section 4.1 introduces the outline of the extended SEM heuristic algorithm. Section 4.2 introduces the heuristic rules. Section 4.3 describes the general computing methods of the BIS index and DIS index.

4.1 Algorithm outline

The extended SEM shares the same outline as the SEM. There are two decisions to be made: sequencing and relocating, which are made by two heuristic rules. First, by using the sequencing rule, we determine the retrieval sequence for one container at a time. Then, according to the relocating rule, the relocation positions are determined.

The notations used in the heuristic are defined below, which are used throughout this section. These notations are also included in Appendix A.2.

t_i: the ith target container to be retrieved, $i \in \{1,...,C\}$.

$n(u)$: the number of relocations for retrieving container u.

X : the state of the bay (bay configuration). Let X_0 represent the initial bay configuration.

lmin: the smallest priority label of the containers in X.

Θ : the set of containers labeled *lmin* in X.

Φ_{kw}: the trucks in batch k that have arrived at sub-time window w.

The following presents the outline of the extended SEM heuristic:

Step 0. *Initialization*. Let $\mathrm{X} = \mathrm{X}_0$, k = 1 and i = 1.

Step 1. *Reveal the truck arrival information*. If $k > K$, stop; otherwise, reveal the trucks arrival information for batch k, that is, Φ_{kw}, $w \in \{1,...,W\}$, and go to Step 2.

Step 2. *Update the bay configuration*. Update X according to the revealed truck arrival information: for w from 2 to W, add $\sum_{w'=1}^{w-1} |\Phi_{kw'}|$ to the priority labels of each container in sub-batch w of batch k.

Step 3. *Determine the target container*. Identify *lmin* and construct Θ . If $|\Theta| = 1$, let the single container in Θ

be t_i; otherwise, determine t_i using the ***Sequencing Rule*** and update X accordingly.

Step 4. *Relocate the blocking containers*. Calculate $n(t_i)$. If $n(t_i) = 0$, go to step 5; otherwise, determine the relocation positions for each of the $n(t_i)$ blocking containers from the top down using the ***Relocating Rule*** and relocate these blocking containers accordingly, and as a result, X is updated.

Step 5. *Retrieve the target container*. Retrieve t_i from X. If $i = \sum_{j=1}^{k} C_j$, $k = k + 1$ and go to step 1; otherwise, $i = i + 1$ and go to step 3.

In the above steps, the main extension we made is the way of updating the bay configuration in Step 2. When considering multiple sub-batches, the priority labels of the containers in each sub-batch $w > 1$ need to be updated according to the number of trucks in all of its former sub-batches.

4.2 Heuristic rules

In the next two sub-sections, we introduce the sequencing rule and the relocation rule used in the extended SEM algorithm.

4.2.1 Sequencing rule

The main idea of the sequencing rule is to minimize the number of relocations needed in the next retrieval, and thus the container that has the smallest number of blocking containers is selected from the set of the candidate containers as the target container. If multiple containers have the same least number of blocking containers, we select the one with the earliest truck arrival time as the target container to reduce the maximum truck turnaround time. The sequencing rule is presented below.

Step 1. *Identify the number of blocking containers of each candidate target container*. Given X, Θ and *lmin*, calculate the $n(u)$ of each container $u \in \Theta$.

Step 2. *Determine the target container*. Sort $\{n(u) : u \in \Theta\}$ in non-decreasing order of $n(u)$. Select the one with the smallest $n(u)$ in Θ as the target container t_i. If multiple ones are having the smallest $n(u)$, we select the one with the earliest truck arrival time as the target container.

Step 3. *Update the bay configuration*. Update X by adding one to the priority labels of the containers in $\Theta \backslash t_i$.

The example in Fig. 3 is used for illustration. In the example, $T = 4$, $S = 5$, $W = 3$, the length of a time window equals 30 minutes, and the system starting time equals zero. The priority matrix shows how the containers' priority labels update as the retrieval proceeds. The preference matrix shows each container's customer preference for each sub-time window. The container ID corresponds to the container in each slot. The truck arrival time matrix reveals the truck arrival time (in minutes) for each container. "×" represents that the truck arrival time for the corresponding container has not been revealed at the current step. The containers in bold represent that the truck arrival information of these containers has just been revealed at the current step. The containers in shaded slots represent the target container. The containers in striped slots represent the blocking containers. In Fig. 3, step 0 reveals that in the first batch, trucks u_6 and u_{10} have arrived at the same sub-time window that is earlier than that of truck u_{13}. The set of candidate target containers is $\Theta = \{u_6, u_{10}\}$ since both u_6 and u_{10} have the smallest priority label (i.e., 1). As u_6 and u_{10} have an equal number of blocking containers (i.e., one blocking container), the target container should be the one with the earlier truck arrival time. In our example, the truck arrival time of u_6 which equals 7 is earlier than that of u_{10} which equals 9. Therefore, we select $u_6$as the target container, and accordingly, the priority label of u_{10} is increased by one, leading to an updated priority label (i.e., 2) at step 1.

4.2.2 Relocating rule

We extend the computing methods of the two important indexes in the SEM heuristic (i.e., BIS and DIS) to the situations of multiple sub-batches. *BIS*(s), which represents the blocking index with sequencing (BIS) of a stack s, is defined as the probability of a container being blocking if relocated to s. $DIS(s)$, which represents the delay index with sequencing (DIS) of a stack s, is defined as the probability of the containers with the highest priority in stack s being the first one to be retrieved within its batch. These two indexes are used for selecting the best relocation stack when ties on the candidate relocation stacks occur. They are particularly important for the problem with larger batches as the ties will occur frequently and need to be broken by some criteria. Before introducing the computing methods of the two indexes (see section 4.3), we first present the framework of the relocating rule in the following.

In the following, we define the notations for describing the relocating rule and these notations are used throughout this section. These notations are also included in Appendix A.2.

c: the blocking container to be relocated;

$\hat{s}$: the stack where the blocking container c is located before relocating;

$m(s)$: the smallest priority label of the containers in stack $s \in \{1,\ldots,S\}$. We let $m(s)$ for an empty stack s equal $C+1$;

$h(s)$: the number of containers in stack s, $s \in \{1,\ldots,S\}$;

S_C: the set of candidate stacks;

s^* : the selected relocation stack.

The following heuristic rule determines the relocation position for the blocking container c from stack $\hat{s}$.

[Condition 1] There is a non-full stack $s \neq \hat{s}$ that satisfies $m(s) > l_c$.

Let $M = \min_{s \in \{1,\ldots,S\} \setminus \hat{s}} \{m(s) : h(s) < T, m(s) > l_c\}$. The stack satisfying $m(s) = M$ is selected. Ties are broken by choosing the leftmost one from the highest ones.

[Condition 2] For all non-full stacks $s \neq \hat{s}$, $m(s) \leq l_c$.

Let $M = \max_{s \in \{1,\ldots,S\} \setminus \hat{s}} \{m(s) : h(s) < T\}$. The set of candidate stacks is represented by $S_C = \{s \,|\, s \in \{1,\ldots,S\} \setminus \hat{s}, h(s) < T, m(s) = M\}$. If $|S_C| = 1$, the only stack satisfying $m(s) = M$ is selected. Otherwise, ties are broken in the following way. If $M = l_c$, the stack in S_C with the minimum BIS is selected, that is, $s^* = \arg\min \{BIS(s) \,|\, s \in S_C\}$; If $M < l_c$, the stack in S_C with the minimum DIS is selected, that is, $s^* = \arg\min \{DIS(s) \,|\, s \in S_C\}$. Further ties are broken by choosing the leftmost one from the highest ones.

The basic idea of the above relocating rule is to avoid or delay the blocking container being relocated again in the future.

4.3 Two key indexes

In the following subsections 4.3.1 and 4.3.2, the methods of computing the two key indexes (BIS and DIS) used in the relocation rule are introduced.

4.3.1 Method of computing BIS

The BIS index is needed to break the first kind of tie in [Condition 2] where multiple candidate stacks satisfying $m(s) = M$ and $M = l_c$. Let M_s be the set of containers in candidate stack s and labeled M. Container c will block if relocated to s only in the situation where at least one container in M_s belongs to the sub-batches prior to the sub-batch of c. $BIS(s)$ is calculated by considering all the scenarios of the sub-batches of c except for the scenario of the first sub-batch. The reason why there is no need to consider the scenario of c in the first sub-batch is that if c is relocated to s in this scenario, according to the sequencing rule, c will be the first to be retrieved in stack s and thus will not block. The pseudocode of calculating $BIS(s)$ is given in Algorithm 1. In Algorithm 1, $p(w,u_i)$ denotes the probability that container u_i is in the sub-batches not before sub-batch w; $p(w)$ denotes the probability of all containers in M_s belonging to the sub-batches not before sub-batch w, and thus $(1- p(w))$ is the probability of at least one container in M_s belonging to the sub-batches before sub-batch w.

Algorithm 1: BIS

```
Input: c ← the blocking container to be relocated
       M ← the maximum of the smallest priority label in each candidate stack
       s ← the stack whose BIS is to be calculated
       M_s ← the set of containers in stack s and with label M
       W ← the number of sub-batches
BIS(s) = 0
for w = 2 to W do
    p(w) = 1
    for each container u_i ∈ M_s do
        p(w,u_i) = 0
        for k = w to W do
            p(w,u_i) = p(w,u_i) + p_ik
        end
        p(w) = p(w) · p(w,u_i)
    end
    BIS(s) = BIS(s) + p_cw·(1- p(w))
end
Return BIS(s)
```

Taking step 1 in Fig. 3 for example. At step 1, u_6 is the target container and u_7 is the blocking container. We need to determine the relocating stack for u_7. According to the state of step 1, $M = 4$ and the set of candidate stacks $S_C = \{1, 2\}$. By Algorithm 1, $BIS(1) = 0.2\times(1\text{-}0.2) + 0.5\times(1\text{-}0.1) = 0.61$; $BIS(2) = 0.2\times(1\text{-}0.9\times0.8) + 0.5\times(1\text{-}0.7\times0.6) = 0.346$. As $BIS(2) < BIS(1)$, we select stack 2 as the relocation stack for u_7.

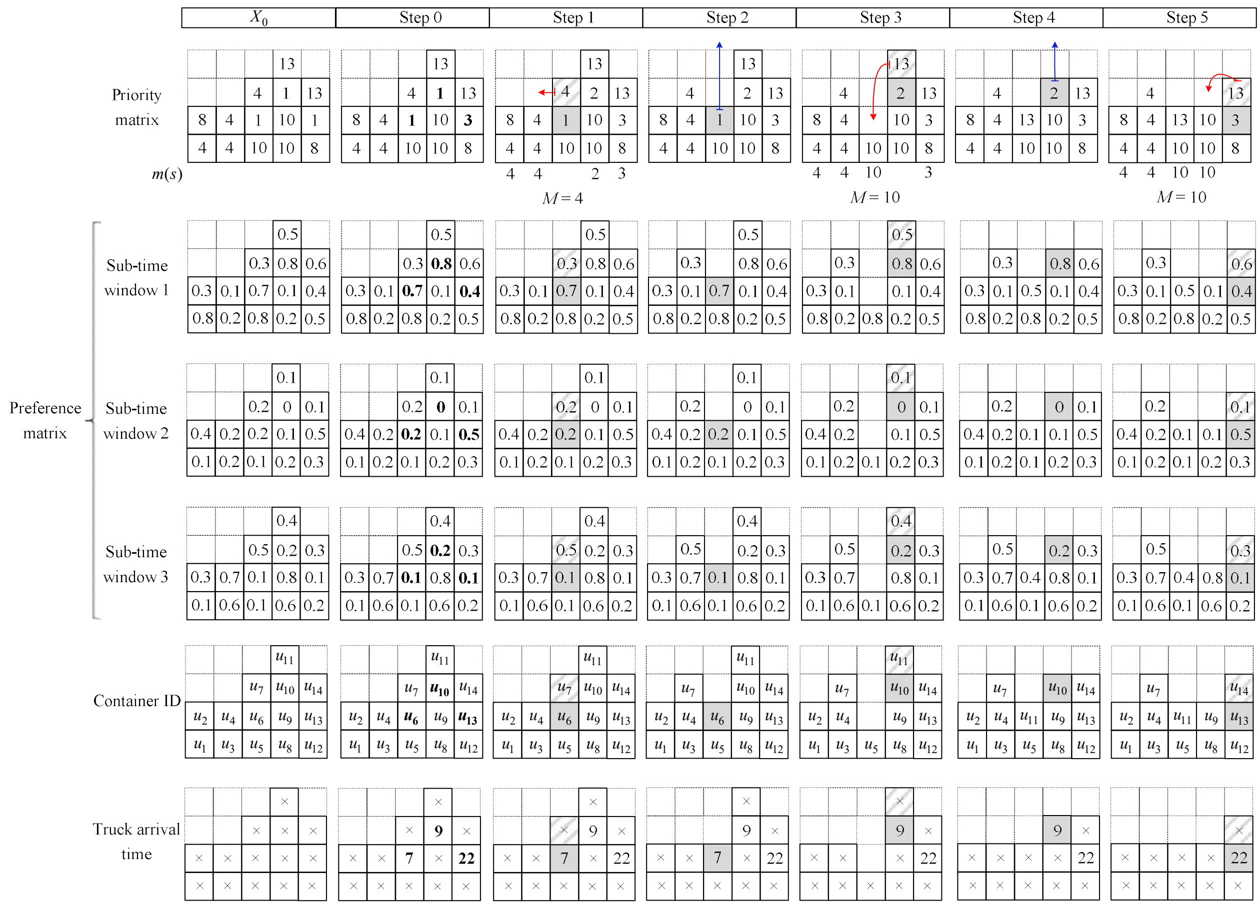


Fig. 3 An example to illustrate the decisions made by the extended SEM heuristic

4.3.2 Method of computing DIS

The DIS index is needed when a tie occurs in [Condition 2] where more than one candidate stacks s satisfying $m(s) = M$ and $M < l_c$. Since there may be more than one container with the same highest priority in a stack, $DIS(s)$ is calculated by $DIS(s) = \sum_{u_i \in M_s} DIS(s, u_i)$, where M_s denotes the set of containers in stack s and with the highest priority, that is, the smallest label M. $DIS(s, u_i)$ denotes the probability of container u_i that has the smallest label in stack s being the first one to be retrieved within its batch. $DIS(s, u_i)$ is calculated by taking all the scenarios of the sub-batch of container u_i into account. In the scenario where u_i belongs to sub-batch w, u_i will surely be the first one to be retrieved within its batch when the following three conditions are satisfied simultaneously:

(i) the containers above u_i and label M are in the sub-batches behind w;

(ii) the containers below u_i and label M are in the sub-batches not before w;

(iii) for each of the other candidate stacks $s' \in S_C \setminus s$, for each container u_j labeled M in stack s': if $n(u_j) \le n(u_i)$ (recall that $n(u_i)$ denotes the number of blocking containers above u_i), u_j are in the sub-batches behind w; otherwise, u_j are in the sub-batches not before w.

The pseudocode of calculating $DIS(s)$ is shown in Algorithm 2. In Algorithm 2, lines 11-19, lines 20-28, and lines 29-47 are to calculate the probability of the above condition (i), (ii) and (iii), respectively. In line 48, $p_{iw} \cdot p(w, u_i)$ represents the probability that container u_i is in sub-batch w and is the first container to be retrieved within its batch. $DIS(s, u_i)$ is obtained by taking all the scenarios of the sub-batch of u_i (line 48) into account. $DIS(s)$ is obtained by considering all the containers with label M in stack s (line 50).

Algorithm 2: DIS

1 **Input:** $B \leftarrow$ the state of the current bay

```
        M ← the maximum of the smallest priority label in each candidate stack
        W ← the number of sub-batches
        s ← the stack whose DIS is to be calculated; S_C ← the set of candidate stacks
        M_s ← the set of containers in stack s and with label M
DIS(s)= 0
for each container u_i ∈ M_s do
    DIS(s, u_i) = 0
    for w = 1 to W do
        p(w,u_i) = 1
        for each container u_j above u_i do
            if the label of u_j equals M then
                sum = 0
                for k = w + 1 to W do
                    sum = sum + p_jk
                end
                p(w,u_i) = p(w,u_i) · sum
            end
        end
        for each container u_j below u_i do
            if the label of u_j equals M then
                sum = 0
                for k = w to W do
                    sum = sum + p_jk
                end
                p(w,u_i) = p(w,u_i) · sum
            end
        end
        n(u_i) ← the number of blocking containers above u_i
        for each stack s′ ∈ S_C / s do
            for each container u_j in stack s′ do
                if the label of u_j equals M then
                    n(u_i) ← the number of blocking containers above u_i
                    sum = 0
                    if n(u_j) <= n(u_i) then
                        for k = w + 1 to W do
                            sum = sum + p_jk
                        end
                    else
                        for k = w to W do
                            sum = sum + p_jk
                        end
                    end
                    p(w,u_i) = p(w,u_i) · sum
                end
            end
        end
        DIS(s, u_i) = DIS(s, u_i) + p_iw · p(w,u_i)
    end
    DIS(s) = DIS(s) + DIS(s, u_i)
end
Return DIS(s)
```

Taking step 5 in Fig. 3 for example. At step 5, the blocking container to be relocated is u_{14}. M = 10 and the set of candidate stacks S_C = {3, 4}. By Algorithm 2, $DIS(3, u_5)$ = 0.8×0.9×0.8+0.1×0.8×0.6+0.1×0×0 = 0.624; $DIS(4, u_9)$ = 0.1×1×1+0.1×0.8×0.2+0.8×0.6×0.1 = 0.164; $DIS(4, u_8)$ = 0.2×0.9×0.2+0.2×0.8×0.1+0.6×0×0 = 0.052. Therefore, $DIS(3)$

= $DIS(3, u_5)$ = 0.624, and $DIS(4) = DIS(4, u_9) + DIS(4, u_8) = 0.216$. As $DIS(4) < DIS(3)$, stack 4 is selected as the relocation stack for u_{14}.

5. Computational experiments

Three sets of experiments are performed. Firstly, the impact of the number of sub-time windows is evaluated. Secondly, we investigate the impact of customer preference. Thirdly, we evaluate the value of customer preference information. The algorithm is coded in MATLAB 2018a and the experiments are performed on a desktop computer with Intel® Core ™ i5-6500 processor, 3.20 GHz CPU and 8 GB of RAM. The computational times of all the experiment instances are within several milliseconds.

Table 2 lists the parameters used in the experiments. The number of tiers (T) in a bay varies from 3 to 6 and the number of stacks (S) in a bay varies from 5 to 10, which covers the dimension of the bay in most container terminals. In total, we have 24 problem classes that are characterised by T and S. The utilisation rate (u) of the bay is set to be 67%. Given T, S and u, the number of containers (C) is calculated by $C = [T*S*u]$, where $[x]$ means the integer closest to x. Two batch sizes {6, 12} are considered, that is, there are on average 6 containers or 12 containers per batch. For each problem class, we have 30 instances that vary in the containers' stacking positions and the number of containers in each batch. Besides, the appointment time window length and the time of per relocation move (termed unit relocation time) and the time of per retrieval move (termed unit retrieval time) also follow the settings in Feng et al. (2020).

Table 2 Parameters setting for the experiments.

Parameter	Range of scenarios	Fixed parameters
Dimension of the bay (T×S)	[3, 6] × [5, 10]	
Utilization rate (u)		67%
Customer preference scenario (CPS)	{homogeneous, heterogeneous, exact}	
Average batch size	{6, 12}	
Length of an appointment time window		30 minutes
Unit relocation time		2 minutes
Unit retrieval time		4 minutes

With regards to the customer preference, three Customer Preference Scenarios (**CPS**s) are considered to characterise whether the customer preference information is available and whether the arrival sub-time windows are certain, which is detailed in Table 3. In Scenario 1 of homogeneous CPS, the preference for each sub-time window w, $w \in \{1,...,W\}$, is evenly distributed, that is, each truck has an equal probability (i.e., $1/W$) to arrive at each sub-time window. This scenario is equivalent to no customer preference information and is used as the baseline. In Scenario 2 of heterogeneous CPS, the probabilities of a truck arriving at different sub-time windows follow the distribution of p_w. We assume that p_w is generated from the uniform distribution $U(0,1)$. This scenario represents the situation where each truck has different preferences for different sub-time windows. In the above two scenarios, each truck will arrive at a sub-time window randomly. In Scenario 3 of exact CPS, each truck is uniformly assigned to one sub-time window which it will arrive at with 100% probability. In this scenario, we know which sub-time window each truck will arrive at exactly.

Table 3 Customer Preference Scenarios (CPSs).

Scenarios \ Sub-time windows	Probabilities: 1	…	w	…	W	Preference information availability	Arrival sub-time windows
1. Homogeneous	$1/W$	$1/W$	$1/W$	$1/W$	$1/W$	Unavailable	Uncertain
2. Heterogeneous	p_1	…	p_w	…	p_W	Available	Uncertain
3. Exact	0	0	1	0	0	Available	Certain

In order to estimate the objective values, we need to sample the customer preference and truck arrival times. The samples are generated through a Monte Carlo simulation. The number of samples required to obtain a relative error γ is calculated by

$$n(\gamma) = \delta^2 \left((1+\gamma) Z_{1-\alpha/2} / \gamma\mu\right)^2 \tag{8}$$

where δ^2 and μ respectively represent the variance and mean of the objective values and $Z_{1-\alpha/2}$ is the 1 - $\alpha/2$ percentile of the normal distribution (Law and Kelton, 2000). In our experiments, we want to estimate the total number of relocations, the average truck turnaround time, and the maximum truck turnaround time, with a relative error of 5% (γ = 5%) and a confidence level of 90% (α = 10%) respectively for each indicator. We conduct two-stage preliminary experiments to calculate, first, the number of the samples of the truck arrival times required, denoted by $n_1(\gamma)$, and then, that of the customer preference, denoted by $n_2(\gamma)$, for each problem class under the three CPSs, respectively. Note that for Scenario 1 of homogeneous CPS, we do not need to calculate $n_2(\gamma)$ as there is only one possibility of the customer preference (that is, the probability for each sub-time window is equal) and thus $n_2(\gamma)$ = 1. At the first stage, given a random instance and a random sample of customer preference, we calculate δ^2 and μ for each performance indicator based on ten random samples of truck arrival times. Then, using Eq. (8), $n_1(\gamma)$ is obtained by taking the greatest value among the number of samples required for each performance indicator. At the second stage, given an instance, we calculate δ^2 and μ based on ten random samples of customer preference (the result of each sample of customer preference is the average over $n_1(\gamma)$ random samples of truck arrival times). Then, using Eq. (8), we obtain $n_2(\gamma)$ by taking the greatest value among the number of samples required for each performance indicator. Depending on parameters T, S, W and the CPS, the values of $n(\gamma)$ vary significantly between zero and several hundred. For example, for the problem class with T =3, S = 5 and W =2 under Scenario 2 of heterogeneous CPS, $n_1(\gamma)$ = 555 and $n_2(\gamma)$ = 30; while for the same problem class under the Scenario 3 of exact CPS, $n_1(\gamma)$ = 12 and $n_2(\gamma)$ = 537. If the resulted $n(\gamma)$ is less than five, we force $n(\gamma)$ to be five, which means that we make at least five repetitions respectively on the customer preference and truck arrival times for each instance.

The parameters of the developed hierarchical iterative approach for handling the model are selected to be: Q_1 = 2, Q_2 = 5, and W_0 = 2, by trials based on the experiment parameters.

5.1 Impact of the number of sub-time windows

In this section, we evaluate the impact of the number of sub-time windows on system overall efficiency and service fairness. We first discuss the results for small batch size (sub-section 5.1.1) and then for large batch size (sub-section 5.1.2). We also compare our results with the literature (sub-section 5.1.3). At the end of this section, we provide managerial insights (sub-section 5.1.4). Apart from the maximum truck turnaround time and the total number of relocations that are optimized, two more evaluation indicators are proposed to evaluate the performance and are listed as follows:

Average truck turnaround time (AveT): AveT is the mean of the total turnaround times of all the trucks, which indicates the turnaround time for each truck on average. AveT has a positive correlation with the total number of relocations and can also represent system overall efficiency.

Coefficient of variation of the truck turnaround time (CVT): Coefficient of variation (CV), also known as relative standard deviation, is defined as the ratio of the standard deviation to the mean, which is a standardized measure of the dispersion of a probability distribution or frequency distribution. As the means of the turnaround times under different numbers of sub-time windows are different, we use the CV to show the extent of variability of the turnaround time in relation to its mean. CVT can also represent service fairness.

The experiments in this section are based on Scenario 1 of homogeneous CPS. Multiple problem classes are constructed to execute the experiments; these classes are characterised by different combinations of the number of tiers (T), the number of stacks (S), and the number of containers (C) (see Table 4 and Table 5). The SCRP-MFS where each appointment time window is divided into two sub-time windows (i.e., W = 2) is regarded as the benchmark. For each problem class, the results obtained under different numbers of sub-time windows (in the range of [3, 6]) are compared with the benchmark. In Table 4, columns "Rel" and "AveT" respectively give the total number of relocations and the average truck turnaround time for a problem class in the benchmark, which are obtained by taking the average of 30 instances, each one containing $n_1(\gamma)$ samples. Under the scenarios of "W = 3", "W = 4", "W = 5" and "W = 6", column "Rel%" reports the relative difference between the total number of relocations of the considered scenario and the benchmark, indicating the percentage increase of the total number of relocations when compared with the benchmark; column "AveT%" reports the relative difference between the average truck turnaround time of the considered scenario and the benchmark, revealing the percentage increase in the average turnaround time in comparison with the benchmark.

Table 4 Comparisons of the total number of relocations and average truck turnaround times under different numbers of sub-time windows with the benchmark (average batch size =6).

Problem class			$W = 2$		$W = 3$		$W = 4$		$W = 5$		$W = 6$	
T	S	C	Rel	AveT	Rel%	AveT%	Rel%	AveT%	Rel%	AveT%	Rel%	AveT%
3	5	10	2.27	29.16	14.0%	1.5%	21.1%	2.4%	25.5%	2.8%	27.6%	3.1%
	6	12	3.09	30.62	9.6%	1.5%	14.3%	2.1%	18.9%	3.0%	20.2%	3.0%
	7	14	3.35	31.15	9.0%	1.5%	13.7%	2.2%	16.2%	2.5%	18.1%	2.9%
	8	16	4.04	30.64	7.5%	0.9%	10.2%	1.8%	12.6%	2.2%	14.1%	2.4%
	9	18	4.41	31.00	6.6%	1.7%	11.0%	2.6%	11.2%	**2.3%**	12.2%	2.7%
	10	20	4.74	29.48	5.6%	1.1%	8.3%	1.6%	9.4%	1.9%	10.1%	2.0%
4	5	13	4.33	32.65	9.7%	2.0%	17.7%	3.3%	18.9%	4.0%	20.6%	4.1%
	6	16	6.28	32.78	6.8%	2.0%	10.5%	3.0%	12.1%	3.6%	14.4%	4.1%
	7	19	6.59	30.98	6.3%	1.7%	10.8%	2.5%	12.1%	3.0%	13.8%	3.4%
	8	21	7.02	32.66	5.7%	1.7%	9.6%	2.7%	**9.5%**	3.0%	11.7%	3.2%
	9	24	8.80	32.72	5.3%	1.9%	**5.0%**	**1.7%**	7.6%	2.8%	7.9%	**2.7%**
	10	27	9.30	31.47	4.4%	1.1%	6.6%	2.1%	7.4%	2.5%	7.9%	2.7%
5	5	17	8.57	34.92	7.4%	2.4%	10.4%	4.0%	12.5%	4.5%	14.7%	5.3%
	6	20	9.02	34.61	6.4%	2.5%	9.9%	3.5%	12.2%	4.3%	12.2%	4.5%
	7	23	10.89	34.80	4.0%	1.7%	7.5%	3.3%	7.9%	3.4%	10.4%	4.9%
	8	27	12.61	34.00	4.8%	2.1%	6.3%	2.7%	7.7%	3.7%	8.8%	3.9%
	9	30	14.59	36.19	3.5%	1.6%	6.2%	3.3%	6.2%	3.3%	**6.0%**	3.7%
	10	34	15.32	34.15	3.8%	1.9%	4.3%	2.3%	5.8%	3.0%	6.5%	3.3%
6	5	20	12.61	35.79	6.9%	3.9%	10.6%	5.5%	11.7%	6.3%	12.0%	**6.2%**
	6	24	13.77	37.01	5.4%	2.5%	8.1%	4.6%	9.3%	5.3%	11.3%	6.1%
	7	28	16.69	40.19	3.2%	2.2%	5.2%	3.4%	7.0%	4.5%	7.0%	4.5%
	8	32	18.01	39.68	3.9%	2.3%	5.6%	3.7%	8.2%	5.1%	8.7%	5.3%
	9	36	19.18	37.41	3.7%	2.6%	5.4%	3.7%	6.2%	3.9%	7.2%	4.5%
	10	40	22.33	37.55	0.7%	1.0%	3.4%	2.9%	4.6%	3.7%	4.6%	**3.5%**

Note: (1) The CPS is "homogeneous"; (2) "W = 2" is the benchmark.

In Table 5, columns "MaxT" and "CVT" respectively report the maximum truck turnaround time and the CV of the truck turnaround time for a problem class in the benchmark, which is obtained by taking the average of 30 instances each of which contains $n_1(\gamma)$ samples. Columns "MaxT%" and "CVT%" respectively represent the percentage reduction in the maximum turnaround time and the CV of the turnaround time when compared with the benchmark.

Table 5 Comparisons of the maximum truck turnaround times and the coefficients of variation under different numbers of sub-time windows with the benchmark (average batch size =6).

Problem class			W = 2		W = 3		W = 4		W = 5		W = 6	
T	*S*	*C*	MaxT	CVT	MaxT%	CVT%	MaxT%	CVT%	MaxT%	CVT%	MaxT%	CVT%
3	5	10	39.57	0.25	-2.7%	-7.3%	-3.8%	-10.1%	-4.1%	-10.8%	-4.5%	-11.1%
	6	12	41.55	0.24	-2.5%	-6.1%	-3.8%	-9.4%	-4.1%	-10.7%	-4.5%	-10.5%
	7	14	42.92	0.25	-2.7%	-4.1%	-3.7%	-6.0%	-4.2%	-6.6%	-4.5%	-7.8%
	8	16	42.83	0.24	-2.8%	-4.7%	-3.8%	-7.2%	-4.5%	-8.6%	-5.0%	**-8.1%**
	9	18	44.03	0.25	-3.1%	-4.3%	-3.8%	-7.1%	-4.7%	-8.0%	-5.1%	**-7.6%**
	10	20	42.27	0.24	-3.1%	-4.9%	-4.2%	-7.6%	-5.2%	-9.1%	-5.5%	**-9.0%**
4	5	13	44.73	0.25	-3.0%	-5.7%	-3.5%	-8.4%	-4.2%	-9.8%	-4.8%	-9.8%
	6	16	46.74	0.26	-3.0%	-8.4%	-3.9%	-10.8%	-4.4%	-11.8%	-4.6%	**-11.3%**
	7	19	44.59	0.26	-2.4%	-3.2%	-3.8%	-7.3%	-4.6%	-8.2%	-5.0%	-8.7%
	8	21	47.24	0.25	-3.3%	-8.0%	-4.3%	-9.8%	-4.4%	**-7.3%**	-5.1%	-10.2%
	9	24	48.25	0.26	-2.9%	-5.4%	-4.9%	-9.4%	**-4.8%**	-9.5%	-5.8%	-10.0%
	10	27	46.13	0.25	-2.7%	-5.3%	-3.3%	-6.9%	-4.2%	-7.9%	-4.9%	-8.8%
5	5	17	49.78	0.27	-2.7%	-7.5%	-3.3%	-8.0%	-3.8%	-10.0%	-4.0%	-10.6%
	6	20	49.80	0.27	-2.2%	-5.1%	-3.2%	-7.6%	-3.5%	-8.0%	-3.5%	**-7.5%**
	7	23	50.43	0.27	-2.7%	-6.4%	-2.9%	-7.5%	-4.3%	-9.2%	**-3.7%**	**-8.7%**
	8	27	51.01	0.27	-2.6%	-3.6%	-4.5%	-9.4%	**-4.3%**	**-8.0%**	-5.2%	-10.3%
	9	30	53.98	0.29	-3.2%	-5.3%	-3.7%	-5.4%	-4.6%	-6.6%	**-4.5%**	**-5.1%**
	10	34	51.91	0.30	-2.3%	-2.0%	-4.3%	-6.1%	-4.7%	**-6.0%**	**-4.4%**	**-5.9%**
6	5	20	51.26	0.26	-1.3%	-5.1%	-1.7%	-9.4%	-2.2%	-10.9%	-2.8%	**-10.3%**
	6	24	54.21	0.27	-2.0%	-5.9%	-2.2%	-8.4%	-3.2%	-10.1%	**-2.7%**	-10.4%
	7	28	59.28	0.31	-2.5%	-5.2%	-3.5%	-7.4%	**-3.1%**	-8.9%	-3.6%	-9.0%
	8	32	59.67	0.31	-3.1%	-3.9%	-3.4%	-6.6%	-3.5%	**-6.0%**	-3.9%	-7.3%
	9	36	58.27	0.30	-3.2%	-5.6%	-4.1%	-7.3%	-4.7%	-9.9%	-4.7%	**-8.9%**
	10	40	58.15	0.33	-2.8%	-2.9%	-3.6%	**-2.8%**	**-3.4%**	-3.5%	**-3.5%**	-4.8%

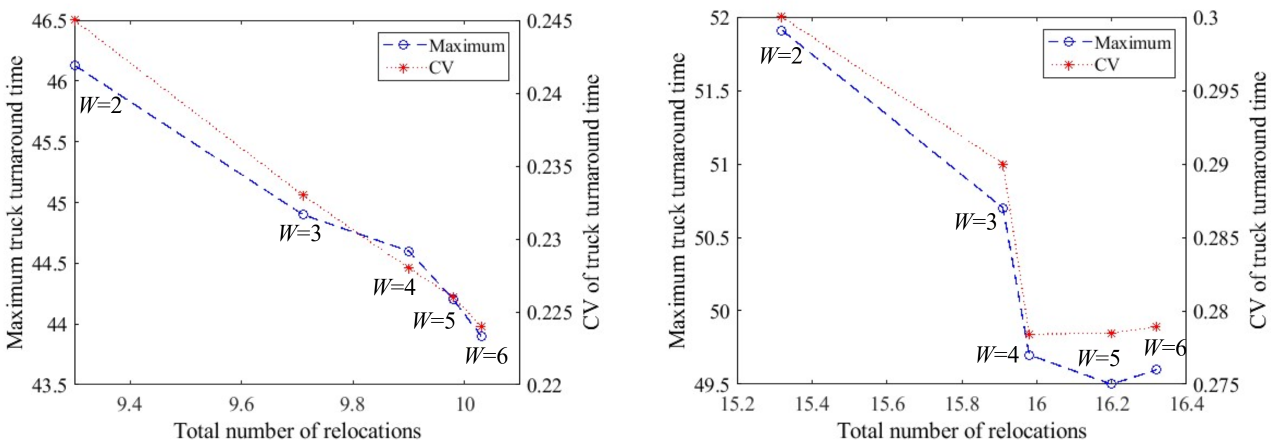
Note: (1) The CPS is "homogeneous"; (2) "*W* = 2" is the benchmark.

5.1.1 Small batch size

The results for the average batch size being 6 are shown in Table 4 and Table 5. From Table 4 and Table 5, we can see that the number of sub-time windows has a significant impact on the results of the SCRP-MFS. Firstly, it can be seen in Table 4 that as the number of sub-time windows increases, both the total number of relocations and the average truck turnaround times are generally increasing. This trend is in agreement with intuition. The reason is that when each appointment time window is split into more sub-time windows, the flexibility to optimize the truck service sequence is reduced. When the out-of-order retrieval policy is implemented within a smaller sub-time window, there are fewer trucks in each sub-time window, and thus the opportunity for optimising the retrieval sequence decreases. As a result, the values of the two performance measures are increasing as *W* increases. It is noticed that there are a few cases (highlighted in bold in Table 4) slightly deviating from this trend. This is probably caused by the sample size. We have conducted additional experiments using a fixed larger sample size (=1000) for each problem class in Table 4. The results show that the two performance measures are indeed increasing with *W* for all the cases. A more revealing observation from Table 4 is that the increase of the two performance measures is disproportional to the increase of the sub-time windows. Specifically, the rates of increase of both Rel% and AveT% tend to decrease as *W* increases.

Secondly, by increasing the number of sub-time windows, generally, both the maximum truck turnaround time and the CV of the turnaround time decrease, but for some problem classes, they show an increasing trend after a certain number of sub-time windows. The fundamental reason for the decreasing trend is the same as in Table 4. Because the scope of out-of-order services is narrowed, the differences between the arrival times of trucks in a sub-batch get smaller, and thus the extra waiting times incurred to the early arriving trucks when later arriving trucks are served before them are shortened. On the other hand, the increasing trend is not surprising, which is attributed to the increasing number of relocations when the flexibility of the out-of-order retrieval decreases. In our experiment, when the number of sub-time windows is six, the flexible service policy is almost reduced to the FCFS policy as on average each sub-batch has only one truck. In this case, since there is little scope for out-of-order service, a truck may experience many relocations and thus have a longer turnaround time; and as some trucks may experience excessively long turnaround times, the variance of turnaround time may also increase.

To have a visual understanding of the impacts of the number of sub-time windows, we take two representative problem classes and display the trade-off between the total number of relocations, the maximum truck turnaround time, and the CV of truck turnaround time in Fig. 4. Each point in Fig. 4 corresponds to the results under a specific number of sub-time windows (from two to six). As shown in Fig. 4(a), both the maximum value and the CV of the truck turnaround time exhibit a decreasing trend as the total number of relocations increases (with the number of sub-time windows increasing from two to six). While in Fig. 4(b), as the number of sub-time windows increases, these two performance measures first decrease when the number of sub-time windows is within four and then tend to increase when the number of sub-time windows is over four.



(a) $T = 4$, $S = 10$ (b) $T = 5$, $S = 10$

Fig. 4 Trade-off between total number of relocations, maximum truck turnaround time, and CV of truck turnaround time (average batch size =6)

Moreover, it is observed that when the number of sub-time windows is over four, increasing the number of sub-time windows would not influence the results of the SCRP-MFS greatly. The reason is that the average numbers of trucks per sub-batch are very small, and as a result, their flexibilities of out-of-order retrievals are similar. When there are six trucks per batch on average, for the scenarios of $W = 5$ and 6, there is on average 1.2 and 1 truck per sub-batch, respectively. As shown in Fig. 4, when W is within four, the relevant measures exhibit a significant changing trend with the increase of W. As W continues to increase, the changing trend tends to be gentle.

Furthermore, we can observe the changing trend of the values of "Rel%" and "CVT%" is that, generally, the smaller the S and T, the greater (absolute) values of "Rel%" and "CVT%". Regarding "AveT%" and "MaxT%", as T increases, the absolute values of "MaxT%" show a slightly decreasing trend while the values of "AveT%" show an opposite trend. The values of "MaxT%" are not sensitive to S while the values of "AveT%" decrease as S increases. These observations indicate that, generally, the number of sub-time windows has a more significant influence on the bay with smaller dimensions. This is because smaller-scale bays are more sensitive to the flexibility of the out-of-order retrieval. To be specific, in a narrower bay (i.e., smaller S), there is less opportunity to find a good stack for the relocated container that will not cause future relocations; and in a lower bay (i.e., smaller T), as the containers are stacked in lower stacks, a larger proportion of relocations can be avoided by out-of-order retrievals. In both cases, it relies more on the optimization of the retrieval sequence to reduce the number of relocations.

5.1.2 Large batch size

The results for the average batch size being 12 are shown in Tables B.1 and B.2 in Appendix B.1. Generally, the observations are the same as the experiments where the average batch size is 6 (Tables 4 and 5). It is worth noticing that, in the problem classes with higher tiers (T = 5, 6), as the number of sub-time windows increases, the maximum truck turnaround times do not show a decreasing trend at the beginning, which is different from the case where the average batch size is 6. This is due to the increasing number of relocations when the service flexibility decreases. However, the CV of the truck turnaround time has been decreasing as the number of sub-time windows increases. This indicates that although the maximum truck turnaround time increases, the difference in the turnaround times between trucks reduces. These observations demonstrate that the best number of sub-time windows also depends on the batch size.

5.1.3 Comparison with existing literature

We have compared our results under SCRP-MFS with the results of Feng et al. (2020) for $W = 2$ obtained by their SEM heuristic, as shown in Tables B.3 and B.4 in Appendix B.2. When $W = 2$, our results of the average truck turnaround times and the total number of relocations are very similar to the results in Feng et al. (2020) (see Table B.3), but the maximum truck turnaround time and the CV of the turnaround time are better than those results in Feng et al. (2020) (see Table B.4). This is because the maximum truck turnaround time is optimized under SCRP-MFS. Accordingly, when W = 3-6, the relative differences between the results under SCRP-MFS and Feng et al. (2020) are calculated for the four performance metrics, i.e., (1) the total number of relocations, (2) the average truck turnaround times, (3) the maximum truck turnaround times, and (4) the CV of truck turnaround times. The results of metrics (1) and (2) are similar to the differences presented in Table 4 where $W = 2$ under SCRP-MFS is used as the benchmark; while the results of metrics (3) and (4) are more significant than those presented in Table 5 where $W = 2$ under SCRP-MFS is used as the benchmark. This is because the maximum truck turnaround times in Feng et al. (2020) are longer.

5.1.4 Management insights

The above results indicate that applying a larger number of sub-time windows can be beneficial to improve service fairness but at the expense of a higher number of relocations and average truck turnaround time. In addition, it should be noticed that service fairness is not necessarily always increasing as the number of sub-time windows increases. Meanwhile, the choice of the number of sub-time windows also depends on the batch size. These findings provide practical insights on the application of flexible service policy and the determination of the number of sub-windows. When the batch size is small, for terminals that are more customer-centric or use a large bay layout, a relatively large number of sub-time windows can have a good balance between the relocation efficiency and the service fairness; on the other hand, if the number of relocations is the primary performance metric of the terminals or a small bay layout is used, applying a relatively small number of sub-time windows appears to be a good choice. When the batch size is large, applying a small number of sub-time windows can tradeoff between the relocation efficiency and the service fairness.

Moreover, the choice of the length/number of the sub-time windows should consider other parameters such as the time of per retrieval move and the time of per relocation move (termed unit retrieval or relocation time). If the unit retrieval time or the unit relocation time is relatively large, each truck would have a larger turnaround time. In such cases, reducing the number of relocations is likely to have a more significant impact on the maximum truck turnaround time than the increase of the number of sub-time windows. As a result, increasing W (i.e. decreasing the length of sub-time windows) is not necessarily reducing the maximum truck turnaround time (i.e. improving service fairness performance). Therefore, when the yard operations are less efficient (requiring a larger unit retrieval time or unit relocation time), it is probably preferable to choose a larger length of sub-time windows to focus more on the relocation performance. Nevertheless, further research is required to investigate how the yard handling efficiency influences the design of sub-time windows in relation to the relocation performance and the service fairness performance.

5.2 Impact of the customer preference scenario

This section investigates whether the SCRP-MFS is sensitive to the CPSs by comparing the total number of relocations under different CPSs. We first discuss the results for small batch size (sub-section 5.2.1) and then for large batch size (sub-section 5.2.2). At the end of this section, we provide managerial insights (sub-section 5.2.3).

5.2.1 Small batch size

Table 6 compares the total number of relocations of the SCRP-MFS with W = 2 under different CPSs when the average batch size equals 6. The results for W = 3 - 6 are given in Tables B.5 – B.8 in Appendix B.3. The total number of relocations under the heterogeneous CPS and the exact CPS for each problem class are obtained by taking the average of 30 instances, each instance containing $n_1(\gamma)$ samples of customer preference, each customer preference sample containing $n_2(\gamma)$ samples of truck arrival times. The middle three columns report the relative difference of the total number of relocations between each pair of CPSs, which is calculated based on the results under the former CPS. For example, for the column "Ho vs. He", Gap = (heterogeneous [Rel] – homogeneous [Rel]) / homogeneous [Rel]$\times$ 100%. The last three columns report the p-value of the paired t-test for each pair of CPSs. At a 5% significance level, the results that differ significantly between the two CPSs are highlighted in Table 6.

Table 6 Comparisons of the total number of relocations between different customer preference scenarios (W = 2, average batch size = 6).

Problem class			Gap			p-Value		
T	*S*	*C*	Ho vs. He	Ho vs. Ex	He vs. Ex	Ho vs. He	Ho vs. Ex	He vs. Ex
3	5	10	0.06%	-1.31%	-1.36%	0.92	**0.03***	0.06
	6	12	-0.48%	-1.62%	-1.15%	0.47	**0.03***	0.20
	7	14	-0.83%	0.57%	1.42%	**0.05***	0.45	0.11
	8	16	0.88%	-0.18%	-1.05%	0.25	0.89	0.46
	9	18	-0.46%	-0.19%	0.27%	0.53	0.77	0.73
	10	20	-1.40%	0.58%	2.00%	**0.02**	0.21	**0.01***
4	5	13	0.32%	-2.28%	-2.59%	0.75	**0.00***	**0.01***
	6	16	-1.30%	-2.09%	-0.81%	**0.04***	**0.00***	0.27
	7	19	-0.20%	-0.23%	-0.04%	0.77	0.81	0.97
	8	21	-0.52%	-0.07%	0.46%	0.22	0.89	0.41
	9	24	-0.88%	-0.79%	0.10%	0.09	0.21	0.89
	10	27	-0.37%	-0.37%	0.00%	0.39	0.65	1.00
5	5	17	-1.73%	-3.40%	-1.71%	**0.01***	**0.00***	0.08
	6	20	0.68%	-0.64%	-1.30%	0.28	0.25	**0.04***
	7	23	0.10%	-1.07%	-1.17%	0.88	0.08	0.16
	8	27	-0.65%	-0.12%	0.53%	**0.05***	0.81	0.29
	9	30	-0.34%	-0.62%	-0.28%	0.37	0.37	0.66
	10	34	0.31%	0.03%	-0.27%	0.33	0.93	0.46
6	5	20	-0.16%	-2.92%	-2.76%	0.81	**0.00***	**0.00***
	6	24	-0.96%	-2.31%	-1.36%	0.16	**0.05***	0.15
	7	28	-0.59%	-0.63%	-0.04%	0.20	0.37	0.95
	8	32	-0.51%	-0.62%	-0.11%	0.31	0.23	0.82
	9	36	-0.93%	0.54%	1.48%	**0.02***	0.53	0.06
	10	40	-0.16%	0.60%	0.76%	0.61	0.30	0.23

Notes: "Ho vs. He" represents homogeneous vs. heterogeneous; "Ho vs. Ex" represents homogeneous vs. exact; "He vs. Ex" represents heterogeneous vs. exact. * Significance level =5%.

From Table 6 and Tables B.5 – B.8 in Appendix B.3, some interesting observations can be identified. Firstly, the gap values in columns "Ho vs. Ex" and "He vs. Ex" show that, generally, the total number of relocations under the exact CPS is lower than that under the homogeneous and heterogeneous CPSs. This is because the influence of the truck arrival uncertainties on the service sequence can be avoided under the exact CPS. Under this CPS, the trucks in a batch are served in a deterministic order since each truck will arrive at a certain sub-time window. Secondly, particularly, the results of the homogeneous CPS (Ho) and the exact CPS (Ex) significantly differ at a 5% level for the problem classes with a narrower bay (S = 5, 6). This significance can be explained by the frequent occurrence of [Condition 2] in the heuristic (section 4.2.2) for a bay that has fewer stacks. For a bay with a smaller S, there is less opportunity to find a good stack for the relocated container that will not cause future relocations, and thus [Condition 2] will occur frequently. Therefore, the BIS and DIS indexes need to be used frequently to determine the relocation stacks. The calculation of these two indexes relies on customer preference information. The exact CPS can lead to a more accurate value of the two indexes and thus a better decision on the relocation stacks.

5.2.2 Large batch size

The results for the average batch size being 12 are shown in Tables B.9 to B.13 in Appendix B.3. Generally, the results of the homogeneous CPS (Ho) and the exact CPS (Ex) significantly differ at a 5% level for the problem classes with a higher bay (T = 5, 6); the results of the heterogeneous CPSs (He) and the exact CPS (Ex) also significantly differ at a 5% level for the problem classes with a higher bay (T = 5, 6). These observations are different from the experiments when the average batch size is 6 where the results under different customer preference scenarios do not significantly differ in most cases (Table 6 and Tables B.5 - B.8). This is because, in the case of a larger batch size, there are more containers with the same batch priority, hence the ties on the candidate relocation stacks occur more frequently, which need to be broken by the BIS and DIS indexes. The exact CPS can lead to a more accurate value of the two indexes and thus a better decision on the relocation stacks.

5.2.3 Management insights

The results in the above tables indicate that the impact of the customer preference scenario depends on the batch size. When the batch size is 12, the number of relocations in the exact CPS is significantly fewer for a high bay than that in the homogeneous CPS and the heterogeneous CPS. This finding establishes a practical insight for the terminal operators to manage truck arrivals. When the batch size is large, for terminals that use high bays, encouraging the trucks to arrive at deterministic sub-time windows can significantly reduce the number of relocations.

When the batch size is 6, the impact is not significant in most cases. This finding complements the results of the corresponding experiments in Feng et al. (2020). Their study shows that the number of relocations is much lower under the CPSs of "100%" and "0%" when compared with those under the homogeneous CPS. This is because the "100%" CPS and "0%" CPS represent that all trucks will arrive at the first sub-time window and the second sub-time window, respectively, both of which provide the largest opportunity for optimizing the retrieval sequence.

5.3 Impact of information availability

In this section, we assess the value of the customer preference information. The information about the customer preference may not always be available. Truckers may not be able to provide their exact arrival probabilities for each sub-time window. To acquire the customer preference, terminals need to keep track of the historical arrival data of each truck. When the customer preference information is unavailable, the terminal operators will assume that the preference for each sub-time window is the same, which is equivalent to the homogeneous CPS. However, in reality, the trucks' arrival behaviour may cohere with the heterogeneous CPS or the exact CPS. Therefore, the decisions made under the assumption of homogeneous CPS may perform badly under the heterogeneous CPS or the exact CPS. In order to understand whether it is necessary to gather and utilize the customer preference information, we assess the value of such information by comparing the results of the SCRP-MFS that utilizes the preference information and that does not utilize the information. The results of the SCRP-MFS without customer information are obtained as follows. First, we make decisions under the assumption of the homogeneous CPS, and then we apply the decisions to the simulation environment where the trucks arrive under the heterogeneous CPS or exact CPS to obtain the total number of relocations.

We first discuss the results for small batch size (sub-section 5.3.1) and then for large batch size (sub-section 5.3.2). At the end of this section, we provide managerial insights (sub-section 5.3.3).

5.3.1 Small batch size

Table 7 and Table 8 respectively show the impacts of the availability of the heterogeneous information (Scenario 2) and the exact information (Scenario 3) on the SCRP-MFS when the average batch size is 6. Column "Gap" reports the relative difference of the total number of relocations obtained by utilizing the customer preference information (Inf [Rel]) and not utilizing the information (NoInf [Rel]), and it is calculated as Gap = (NoInf [Rel] – Inf [Rel]) / NoInf [Rel]×100%. Column "p-value" reports the p-value of the paired *t*-test of using and not using the information (significance level = 5%).

Table 7 Comparisons of the total number of relocations between utilizing the "exact" information and not utilizing such information (average batch size = 6).

Problem class			$W = 2$		$W = 3$		$W = 4$		$W = 5$		$W = 6$	
T	S	C	Gap	p-value	Gap	p-value	Gap	p-value	Gap	p-value	Gap	p-value
3	5	10	1.1%	0.00	1.4%	0.00	1.1%	0.00	1.2%	0.00	1.2%	0.00
	6	12	1.3%	0.00	1.9%	0.00	1.6%	0.00	2.0%	0.00	2.1%	0.00
	7	14	0.6%	0.00	0.9%	0.00	1.0%	0.00	1.0%	0.00	0.9%	0.00
	8	16	0.2%	**0.26**	0.8%	0.01	0.6%	0.02	0.9%	0.01	1.0%	0.01
	9	18	0.7%	0.05	0.6%	0.05	0.4%	0.01	0.8%	0.02	0.6%	0.01
	10	20	0.2%	0.03	0.3%	0.02	0.4%	0.03	0.5%	0.01	0.3%	0.04
4	5	13	2.3%	0.00	1.9%	0.00	2.0%	0.00	2.1%	0.00	2.1%	0.00
	6	16	1.2%	0.00	0.9%	0.00	1.5%	0.00	1.6%	0.00	1.7%	0.00
	7	19	0.7%	0.00	1.3%	0.00	1.2%	0.00	1.0%	0.00	1.1%	0.00
	8	21	0.8%	0.00	1.0%	0.00	0.9%	0.00	1.0%	0.00	1.0%	0.00
	9	24	0.5%	0.01	0.6%	0.00	0.5%	0.00	0.7%	0.00	0.5%	0.00
	10	27	0.4%	0.01	0.6%	0.02	0.6%	0.00	0.6%	0.00	0.5%	0.00
5	5	17	3.0%	0.00	2.8%	0.00	2.6%	0.00	3.2%	0.00	2.4%	0.00
	6	20	1.3%	0.00	1.5%	0.00	1.5%	0.00	1.7%	0.00	1.5%	0.00
	7	23	1.6%	0.00	1.8%	0.00	1.8%	0.00	2.2%	0.00	2.0%	0.00
	8	27	1.0%	0.00	0.9%	0.00	1.1%	0.00	1.3%	0.00	1.5%	0.00
	9	30	0.7%	0.00	0.5%	0.00	0.8%	0.00	0.6%	0.01	0.6%	0.00
	10	34	0.3%	0.01	0.4%	0.00	0.4%	0.00	0.5%	0.00	0.6%	0.00
6	5	20	2.4%	0.00	2.2%	0.00	2.5%	0.00	2.4%	0.00	2.1%	0.00
	6	24	2.2%	0.00	2.0%	0.00	1.7%	0.00	1.6%	0.00	1.7%	0.00
	7	28	0.7%	0.00	1.0%	0.00	0.9%	0.00	1.2%	0.00	1.1%	0.00
	8	32	0.6%	0.01	0.9%	0.00	0.9%	0.00	0.9%	0.00	0.7%	0.01
	9	36	0.5%	0.01	0.9%	0.00	0.9%	0.00	1.1%	0.00	1.1%	0.00
	10	40	0.4%	**0.11**	0.7%	0.01	0.7%	0.01	0.8%	0.02	0.6%	0.04

Note: The bold numbers represent not significantly differing at a 5% level.

In Table 7, the p-values show that, in most cases (118 out of 120 problem classes), the results of utilizing the exact information and not utilizing such information significantly differ at a 5% level. However, their relative differences (see Gap) are small, which vary between 0.2% and 3.2%. In addition, regarding whether or not to utilize the heterogeneous information, Table 8 shows that about one-fifth of problem classes (24 out of 120) do not show a significant difference (p-values >0.05). A common feature in these non-significant problem classes is that their bays generally have more stacks (e.g., S = 9, 10). Another observation from the two tables is that in both scenarios, the bays with fewer stacks have greater gap values, and in the exact scenario, the gap values show an increasing trend as the number of tiers increases. The reason is that in the problem classes with fewer stacks and higher tiers, the BIS and DIS indexes are used more frequently to determine the relocation stacks. The calculation of the two indexes relies on the preference information, but the assumed equal preference may lead to bad decisions on the relocation stacks and thus result in more relocations. Moreover, it is observed that the gap values in Table 8 are smaller than Table 7 and are no more than 1.3%. One explanation for this phenomenon could be that the differences in the values of the two indexes between the exact CPS and the homogeneous CPS are larger than that between the heterogeneous CPS and the homogeneous CPS, given that the information in the exact CPS is more accurate.

Table 8 Comparisons of the total number of relocations between utilizing the "heterogeneous" information and not utilizing such information (average batch size = 6).

Problem class			$W = 2$		$W = 3$		$W = 4$		$W = 5$		$W = 6$	
T	S	C	Gap	p-value	Gap	p-value	Gap	p-value	Gap	p-value	Gap	p-value
3	5	10	0.5%	0.00	0.4%	0.00	0.3%	0.00	0.3%	0.00	0.2%	0.00
	6	12	0.7%	0.00	0.7%	0.00	0.6%	0.00	0.5%	0.00	0.6%	0.00
	7	14	0.3%	0.00	0.3%	0.00	0.2%	0.00	0.2%	0.00	0.2%	0.00
	8	16	0.4%	0.02	0.3%	**0.07**	0.2%	**0.17**	0.3%	0.02	0.3%	**0.08**
	9	18	0.3%	**0.06**	0.2%	**0.09**	0.2%	0.03	0.2%	**0.06**	0.0%	**0.54**
	10	20	0.2%	0.01	0.2%	**0.06**	0.2%	0.01	0.2%	**0.06**	0.1%	0.04
4	5	13	0.7%	0.00	0.5%	0.00	0.4%	0.00	0.4%	0.00	0.3%	0.04
	6	16	0.7%	0.00	0.6%	0.00	0.5%	0.00	0.3%	0.00	0.3%	0.00
	7	19	0.5%	0.00	0.5%	0.00	0.4%	0.01	0.3%	0.03	0.3%	0.00
	8	21	0.5%	0.00	0.5%	0.00	0.2%	0.00	0.2%	0.02	0.3%	0.00
	9	24	0.3%	0.03	0.3%	0.01	0.3%	0.00	0.2%	0.00	0.1%	0.03
	10	27	0.1%	**0.31**	0.1%	**0.25**	0.2%	0.04	0.2%	0.02	0.2%	0.01
5	5	17	1.3%	0.00	0.6%	0.00	0.7%	0.00	0.5%	0.00	0.4%	0.00
	6	20	0.7%	0.00	0.6%	0.00	0.5%	0.00	0.4%	0.00	0.3%	0.02
	7	23	1.0%	0.00	0.7%	0.00	0.8%	0.00	0.5%	0.00	0.5%	0.00
	8	27	0.6%	0.00	0.5%	0.00	0.4%	0.00	0.5%	0.00	0.4%	0.00
	9	30	0.4%	0.00	0.3%	0.01	0.5%	0.00	0.3%	0.04	0.2%	**0.08**
	10	34	0.1%	0.01	0.1%	0.02	0.1%	**0.37**	0.1%	0.01	0.1%	**0.26**
6	5	20	0.9%	0.00	0.8%	0.00	0.7%	0.01	0.4%	0.00	0.3%	0.03
	6	24	1.1%	0.01	1.0%	0.00	0.6%	0.02	0.7%	0.02	0.3%	0.00
	7	28	0.4%	0.00	0.3%	0.00	0.0%	**0.89**	0.3%	0.05	0.2%	0.01
	8	32	0.3%	0.01	0.2%	**0.12**	0.2%	0.04	0.2%	**0.11**	0.2%	**0.08**
	9	36	0.4%	0.00	0.3%	0.01	0.2%	**0.06**	0.3%	0.00	0.3%	**0.06**
	10	40	0.4%	0.01	0.2%	**0.07**	0.1%	**0.06**	0.1%	**0.21**	0.1%	**0.49**

Note: The bold numbers represent not significantly differing at a 5% level.

5.3.2 Large batch size

The results for the average batch size being 12 are shown in Tables B.14 and B.15 in Appendix B.4. The p-values in the two tables show that in most cases, the results of utilizing information and not utilizing information significantly differ at a 5% level. This observation is the same as the experiments where the average batch size is 6 (Tables 7 and 8). A different observation is that some small-scale problem classes that are highlighted in bold do not show a significant difference (p-values >0.05) and the gap values are very minor that are around 0.0%. However, this observation does not change the conclusion on the value of information availability as it is caused only by insufficient instance size. This is because, in these problem classes, the total number of containers is fewer than the sum of two batches. After the sub-batch priority of the first batch is revealed, there is only one or zero batch of containers that have the same batch priority, which means that the uncertainty is low and the occurrence of ties is rare. Therefore, the availability of the customer arrival preference information does not make much difference.

In addition, in Tables B.14 and B.15 where the average batch size equals 12, it is observed that the gap values of the problem classes that have no fewer than two batches of containers (which have been highlighted in italic) are greater than those in Tables 7 and 8 where the batch size is 6. This observation indicates that when there are more trucks per batch, the value of customer preference information is greater.

5.3.3 Management insights

The above results indicate that whether or not utilizing the customer preference information can have quite a different impact on the number of relocations. The difference is minor for the heterogeneous CPS but is obvious for the exact CPS and is larger in the case with a larger batch size. This implies that, when each truck only prefers a specific sub-time window, it is worthy to gather and utilize such customer preference information, especially when the batch size is large.

6. Conclusions

This paper investigates the trade-off between the number of relocations and the service fairness and assesses the value of customer information in the SCRP under flexible service policies in the retrieval process of import containers. To handle the issue of service unfairness, two measures are adopted in two phases. In phase 1, each appointment time window is split into multiple sub-time windows equally and the flexible service policy is only applied to individual sub-time windows. In phase 2, we develop a stochastic dynamic programming model with two lexicographically ordered objective functions, in which minimizing the expected total number of relocations is the primary objective, which reflects relocation efficiency, and minimizing the maximum truck turnaround time is the secondary objective, which reflects service fairness. The model is tackled using a hierarchical iterative heuristic.

The computational experiments demonstrate that our model is effective in reducing the maximum truck turnaround time while incurring a moderate increase in the total number of relocations and only a slight increase in the average truck turnaround time. Meanwhile, the coefficient of variation of the truck turnaround time, which is another metric of service fairness, can be reduced greatly. This suggests that when applying the flexible service policy, the number of sub-time windows should be carefully determined to balance the number of relocations and the performance of individual trucks. In addition, it is found that the service fairness is not necessarily always improving as the number of sub-time windows increases. This implies that although the commonly used FCFS policy appears to be fair in terms of service sequence, it does not always ensure service fairness. Moreover, it is shown that the results between utilizing and not utilizing the customer arrival preference information are significantly different in most cases. This suggests that if the yard planning system is able to gather and utilize customer preference information, the number of relocations can be reduced.

As future work, apart from the maximum truck turnaround time, more measures can be proposed to describe the service fairness for the container retrieval process and considered in the multi-objective decision-making process. Besides, it deserves more research to investigate the specific relationship between the length of sub-time windows and time parameters (e.g., unit relocation time, unit retrieval time). In addition, the problem can be extended to consider dynamic container arrivals and retrievals in a three-dimensional (bays, rows, and tiers) storage area, which considers the yard crane working efficiency, the container stacking efficiency at the quayside and the container retrieval efficiency at the landside. Future research could also study the CRP for containers by trains which are usually collected in large batches and need to follow strict departure times of trains.

Acknowledgements:

The authors would like to thank three anonymous reviewers for their insightful comments and constructive suggestions. This study is partially supported by the Royal Society (Grant No. IEC\NSFC\170100) and the EU H2020 (Grant No. 777742, EC H2020-MSCA-RISE-2017).

Appendix. Supplementary materials at:

https://www.sciencedirect.com/science/article/abs/pii/S1366554522002988#!

References

Azab, A., & Morita, H. (2022a). The block relocation problem with appointment scheduling. European Journal of Operational Research, 297(2), 680-694.

Azab, A., & Morita, H. (2022b). Coordinating truck appointments with container relocations and retrievals in container terminals under partial appointments information. Transportation Research Part E: Logistics and Transportation Review, 160, 102673.

Bacci, T., Mattia, S., & Ventura, P. (2019). The bounded beam search algorithm for the block relocation problem. Computers & Operations Research, 103, 252-264.

Bacci, T., Mattia, S., & Ventura, P. (2020). A branch-and-cut algorithm for the restricted Block Relocation Problem. European Journal of Operational Research, 287(2), 452-459.

Bertsimas, D., Farias, V. F., & Trichakis, N. (2011). The price of fairness. Operations research, 59(1), 17-31.

Bertsimas, D., Farias, V. F., & Trichakis, N. (2013). Fairness, efficiency, and flexibility in organ allocation for kidney transplantation. Operations Research, 61(1), 73-87.

Borjian, S., Manshadi, V., Barnhart, C., & Jaillet, P. (2013). dynamic stochastic optimization of reshuffles in container terminals. In Manufacturing and Service Operations Management (MSOM) conference.

Borjian, S., Manshadi, V. H., Barnhart, C., & Jaillet, P. (2015). Managing relocation and delay in container terminals with flexible service policies. arXiv preprint 1503.01535.

Caserta, M., Schwarze, S., & Voß, S. (2012). A mathematical formulation and complexity considerations for the blocks relocation problem. European Journal of Operational Research, 219(1), 96-104.

Caserta, M., Schwarze, S., & Voß, S. (2020). Container rehandling at maritime container terminals: a literature update, in: Böse, J.W. (Ed.), Handbook of Terminal Planning. Operations Research/Computer Science Interfaces Series 64. Springer Nature Switzerland AG, pp.343-382.

Chen, G., Govindan, K., & Golias, M. M. (2013a). Reducing truck emissions at container terminals in a low carbon economy: Proposal of a queueing-based bi-objective model for optimizing truck arrival pattern. Transportation Research Part E: Logistics and Transportation Review, 55, 3-22.

Chen, G., Govindan, K., Yang, Z. Z., Choi, T. M., & Jiang, L. (2013b). Terminal appointment system design by non-stationary M(t)/Ek/c(t) queueing model and genetic algorithm. International Journal of Production Economics, 146(2), 694-703.

Covic, F. (2017). Re-marshalling in automated container yards with terminal appointment systems. Flexible Services and Manufacturing Journal, 29(3), 433-503.

de Melo da Silva, M., Erdoğan, G., Battarra, M., & Strusevich, V. (2018). The block retrieval problem. European Journal of Operational Research, 265(3), 931-950.

Expósito-Izquierdo, C., Melián-Batista, B., & Moreno-Vega, J. M. (2015). An exact approach for the blocks relocation problem. Expert Systems with Applications, 42(17-18), 6408-6422.

Feng, Y., Song, D. P., Li, D., & Zeng, Q. (2020). The stochastic container relocation problem with flexible service policies. Transportation Research Part B: Methodological, 141, 116-163.

Feng, Y., Song, D. P., & Li, D. (2022). Smart stacking for import containers using customer information at automated container terminals. European Journal of Operational Research, 301(2), 502-522.

Galle, V., Manshadi, V. H., Boroujeni, S. B., Barnhart, C., & Jaillet, P. (2018). The stochastic container relocation problem. Transportation Science, 52(5), 1035-1058.

Gharehgozli, A., & Zaerpour, N. (2018). Stacking outbound barge containers in an automated deep-sea terminal. European Journal of Operational Research, 267(3), 977-995.

Hakan Akyüz, M., & Lee, C. Y. (2014). A mathematical formulation and efficient heuristics for the dynamic container relocation problem. Naval Research Logistics (NRL), 61(2), 101-118.

Jang, D. W., Kim, S. W., & Kim, K. H. (2013). The optimization of mixed block stacking requiring relocations. International Journal of Production Economics, 143(2), 256-262.

Jia, S., Meng, Q., & Kuang, H. (2022). Equitable vessel traffic scheduling in a seaport. Transportation Science, 56(1), 162-181.

Jiang, T., Zeng, B., Wang, Y., & Yan, W. (2021, April). A new heuristic reinforcement learning for container relocation problem. In Journal of Physics: Conference Series (Vol. 1873, No. 1, p. 012050). IOP Publishing.

Jin, B., Zhu, W., & Lim, A. (2015). Solving the container relocation problem by an improved greedy look-ahead heuristic. European Journal of Operational Research, 240(3), 837-847.

Jovanovic, R., & Voß, S. (2014). A chain heuristic for the blocks relocation problem. Computers & Industrial Engineering, 75, 79-86.

Kang, J., Ryu, K. R., & Kim, K. H. (2006). Deriving stacking strategies for export containers with uncertain weight information. Journal of Intelligent Manufacturing, 17(4), 399-410.

Kim, K. H., & Hong, G. P. (2006). A heuristic rule for relocating blocks. Computers & Operations Research, 33(4), 940-954.

Kim, K. H., & Yi, S. (2021). Utilizing information sources to reduce relocation of inbound containers. Maritime Economics & Logistics, 1-24.

Knatz, G. (2017). How competition is driving change in port governance, strategic decision-making and government policy in the United States. Research in Transportation Business & Management, 22, 67-77.

Ku, D., & Arthanari, T. S. (2016a). Container relocation problem with time windows for container departure. European Journal of Operational Research, 252(3), 1031-1039.

Ku, D., & Arthanari, T. S. (2016b). On the abstraction method for the container relocation problem. Computers & Operations Research, 68, 110-122.

Law, A.M., & Kelton, W.D. (2000). Simulation Modeling and Analysis (3rd ed). Boston: McGraw-Hill, pp. 511-515.

Li, D., Zhang, T., Dong, X., Yin, Y., & Cao, J. (2019). Trade-off between efficiency and fairness in timetabling on a single urban rail transit line under time-dependent demand condition. Transportmetrica B: Transport Dynamics.

Maglić, L., Gulić, M., & Maglić, L. (2020). Optimization of container relocation operations in port container terminals. Transport, 35(1), 37-47.

Matl, P., Hartl, R. F., & Vidal, T. (2018). Workload equity in vehicle routing problems: A survey and analysis. Transportation Science, 52(2), 239-260.

Parreño-Torres, C., Alvarez-Valdes, R., Ruiz, R., & Tierney, K. (2020). Minimizing crane times in pre-marshalling problems. Transportation Research Part E: Logistics and Transportation Review, 137, 101917.

Phan, M. H., & Kim, K. H. (2015). Negotiating truck arrival times among trucking companies and a container terminal. Transportation Research Part E: Logistics and Transportation Review, 75, 132-144.

Qi, J. (2017). Mitigating delays and unfairness in appointment systems. Management Science, 63(2), 566-583.

Quispe, K. E. Y., Lintzmayer, C. N., & Xavier, E. C. (2018). An exact algorithm for the blocks relocation problem with new lower bounds. Computers & Operations Research, 99, 206-217.

Tanaka, S., & Takii, K. (2015). A faster branch-and-bound algorithm for the block relocation problem. IEEE Transactions on Automation Science and Engineering, 13(1), 181-190.

Tanaka, S., & Mizuno, F. (2018). An exact algorithm for the unrestricted block relocation problem. Computers & Operations Research, 95, 12-31.

Tang, L., Jiang, W., Liu, J., & Dong, Y. (2015). Research into container reshuffling and stacking problems in container terminal yards. IIE Transactions, 47(7), 751-766.

Teye, C., & Bell, M. (2016). Dynamic discrete berth allocation in container terminals under four performance measures. The University of Sydney, working paper.

Ting, C. J., & Wu, K. C. (2017). Optimizing container relocation operations at container yards with beam search. Transportation Research Part E: Logistics and Transportation Review, 103, 17-31.

Tong, X., Woo, Y. J., Jang, D. W., & Kim, K. H. (2015). Heuristic rules based on a probabilistic model and a genetic algorithm for relocating inbound containers with uncertain pickup times. International Journal of Industrial Engineering, 22, 93–101.

van Asperen, E., Borgman, B., & Dekker, R. (2013). Evaluating impact of truck announcements on container stacking efficiency. Flexible Services and Manufacturing Journal, 25(4), 543-556.

Wan, Y. W., Liu, J., & Tsai, P. C. (2009). The assignment of storage locations to containers for a container stack. Naval Research Logistics (NRL), 56(8), 699-713.

Wang, Y., Xiao, Z., Huang, Y., Hao, Y., & Gu, T. (2017). Study of continuous berth allocation algorithm based on fairness maximization. Journal of Engineering Science & Technology Review, 10(5), 116-127.

Wu, Y., Yang, H., Zhao, S., & Shang, P. (2021). Mitigating unfairness in urban rail transit operation: A mixed-integer linear programming approach. Transportation Research Part B: Methodological, 149, 418-442.

Xie, Y., & Song, D. P. (2018). Optimal planning for container prestaging, discharging, and loading processes at seaport rail terminals with uncertainty. Transportation Research Part E: Logistics and Transportation Review, 119, 88-109.

Yang, M., Allen, T. T., Fry, M. J., & Kelton, W. D. (2013). The call for equity: simulation optimization models to minimize the range of waiting times. IIE Transactions, 45(7), 781-795.

Zehendner, E., & Feillet, D. (2014). A branch and price approach for the container relocation problem. International Journal of Production Research, 52(24), 7159-7176.

Zehendner, E., Caserta, M., Feillet, D., Schwarze, S., & Voß, S. (2015). An improved mathematical formulation for the blocks relocation problem. European Journal of Operational Research, 245(2), 415-422.

Zehendner, E., Feillet, D., & Jaillet, P. (2017). An algorithm with performance guarantee for the online container relocation problem. European Journal of Operational Research, 259(1), 48-62.

Zeng, Q., Feng, Y., & Yang, Z. (2019). Integrated optimization of pickup sequence and container rehandling based on partial truck arrival information. Computers & Industrial Engineering, 127, 366-382.

Zhang, C., Guan, H., Yuan, Y., Chen, W., & Wu, T. (2020). Machine learning-driven algorithms for the container relocation problem. Transportation Research Part B: Methodological, 139, 102-131.

Zhang, X., Chen, X., Ji, M., & Yao, S. (2017). Vessel scheduling model of a one-way port channel. Journal of Waterway, Port, Coastal, and Ocean Engineering, 143(5), 04017009.

Zhao, W., & Goodchild, A. V. (2010). The impact of truck arrival information on container terminal rehandling. Transportation Research Part E: Logistics and Transportation Review, 46(3), 327-343.

Zhou, C., Wang, W., & Li, H. (2020). Container reshuffling considered space allocation problem in container terminals. Transportation Research Part E: Logistics and Transportation Review, 136, 101869.

Zhu, W., Qin, H., Lim, A., & Zhang, H. (2012). Iterative deepening A* algorithms for the container relocation problem. IEEE Transactions on Automation Science and Engineering, 9(4), 710-722.

Zuidwijk, R. A., & Veenstra, A. W. (2015). The value of information in container transport. Transportation Science, 49(3), 675-685.