Scheduling AGVs in ports with battery charging and swapping

Lu Zhen [1], Qian Zhang [2], Zheyi Tan [1], Shuaian Wang [3*]

[1] *School of Management, Shanghai University, Shanghai, China*
[2] *Logistics Engineering College, Shanghai Maritime University, Shanghai, China*
[3] *Faculty of Business, The Hong Kong Polytechnic University, Hung Hom, Hong Kong*
* **Corresponding author**: wangshuaian@gmail.com

Abstract: Efficient scheduling of automated guided vehicles (AGVs) in automated container terminals (ACTs) is crucial to their operations management under the initiative of green port and smart port development. Facing numerous containers, a port operator needs to assign container loading/unloading tasks to AGVs, sequence the tasks for each AGV, and arranges the battery charging or swapping activities for the AGVs in a high-efficient manner, which affects the operational performance of the port and further impacts its annual throughput. This study proposes a mathematical model for scheduling AGVs under the battery charging-swapping mixed mode. The model should be solved in a short time, otherwise the proposed methodology cannot apply in realistic port environment. Therefore, an efficient column generation based solution method is designed for solving the models; a novel label setting algorithm is also proposed and embedded in the above method for solving the pricing problems in a fast way. Based on two representative ACTs, i.e., the first ACT in China and the largest ACT in the world, numerical experiments are conducted to derive some managerial insights for ACTs' operations management on AGVs under different battery management modes.

Keywords: Port operations; automated guided vehicles; battery; automated container terminals; column generation.

1. Introduction

In the trend of digitalization in operations management, the denomination of "smart" is being associated with various industries and domains. As the nodes within the global supply chain network, container ports and their operational efficiency have a great influence on global business activities (Fransoo and Lee 2013; Meng et al. 2014; Lee and Song 2017). The initiative of "smart port" has been pursued worldwide by port operators to transform their traditional ports into ones with superior efficiency, unmanned operations, and sustainable operations, which are usually based on the digitalization, on use of ICT (information and communications technology) and on automation of port operations processes. Automated guided vehicles (AGVs) have been proposed as a key technology to help not only increase the intelligent operation level but also reduce the greenhouse gas (GHG) emissions during the port operations; the environmental pollution at maritime ports has also been highly concerned by governments, which have taken a series of measures for greening the port industry (Wang et al. 2018 and 2020; Wu et al. 2023). Thus, efficient scheduling of AGVs in container ports is crucial to their operations management under the initiatives of green port and smart port development (Roy et al. 2020).

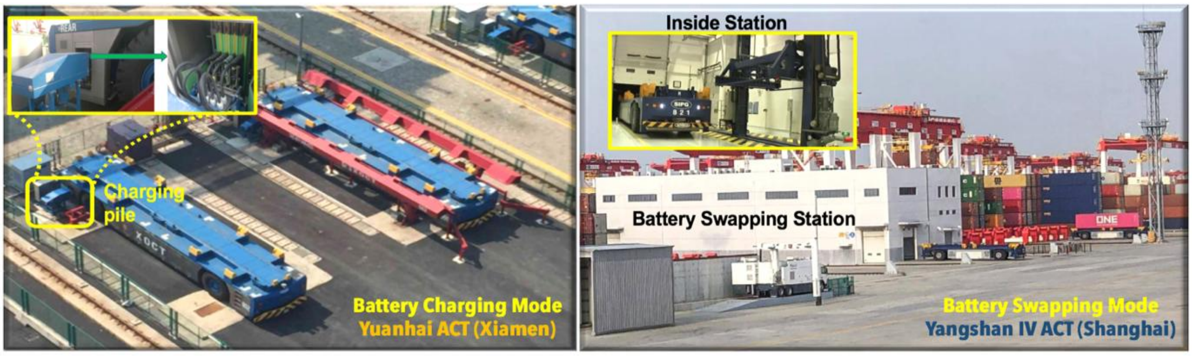


Figure 1: Battery charging and swapping modes in container ports

Scheduling the electric AGVs should not ignore the batteries' charging and swapping, as shown in Figure 1. Neglecting these aspects in the scheduling of electric AGVs suggests that, theoretically, the methods used for trucks scheduling in traditional ports could also be applicable to AGVs. Due to the limited capacity of batteries, an AGV can usually travel for about seven hours on a single charge and needs about two hours to be recharged; alternatively, the AGV's depleted battery can also be exchanged with a full-charged battery (Widrick et al. 2018). For the battery charging mode, the advantage mainly lies in the low-cost construction, small occupied area and flexible distribution of the charging piles. AGVs need to travel to their nearby and available charging piles before the electricity run out. If an AGV's battery is to be fully charged, it becomes available to undertake container transporting tasks for a relatively long time. Some ports may not enforce their AGVs to be fully charged every time so as to increase the operational efficiency; an AGV can make use of the free time slots between two consecutive tasks and charges for a while such as the waiting time for a yard crane's operation at the end of a yard block. As the first automated container terminal (ACT) in China, the Yuanhai port in Xiamen is equipped with 18 AGVs and eight charging piles under the battery charging mode; based on the port's estimation, it can save energy by about 25% and reduce GHG by 16% when comparing with traditional port system. For the battery swapping mode, its relative merit lies in short required time (about six minutes) of making an AGV become fully powered rather than waiting at a charging pile for about two hours. However, the disadvantage of this mode is the high-cost construction and large occupied area of the battery swapping stations in container yards. As the largest ACT in the world, the Yangshan Phase-IV port in Shanghai is equipped with 130 AGVs and two automatic battery swapping stations; it declares an effect of 40% energy saving by comparing with the traditional port and zero GHG emission during the charging process for the depleted batteries that are exchanged from AGVs. By combining the relative merits of the above two modes, an ACT could adopt a mixed mode, in which the AGVs with extremely low electric power or battery faults could be handled by the swapping mode, while the AGVs with relatively long free time slots and medium electric power could be handled by the charging mode so as to further increase their travel endurance. The above mixed mode may become an emerging trend under the context of the green and smart port initiative.

As a basic decision in an ACT's operations management, AGV scheduling mainly contains the assignment of container transporting tasks to the available AGVs and the sequencing of the tasks assigned to each AGV. Besides the above routine decisions on assigning and sequencing tasks for vehicles, which is the same as the yard trucks' scheduling in traditional ports, the timing and routing decisions for charging AGVs or swapping batteries for AGVs further complicate the vehicle scheduling problem in the ACTs. Moreover, the battery mixed mode leads to detailed model as well as algorithm for the above mentioned complex AGV scheduling problem with battery charging and swapping activities (Zhong et al. 2024). This study aims to propose a systematic methodology on AGV scheduling in ACTs under the mixed battery mode. A mixed integer linear programming (MILP) model is designed for the problem under specific contexts; a column generation (CG) based solution method is implemented for solving the proposed models efficiently. Some acceleration tactics are also proposed to facilitate the application of the proposed methodology in large-scale applications in reality. The experiments based on the ACTs in Shanghai and Xiamen ports are conducted to not only validate the proposed methodology's effectiveness but also provide some potentially useful managerial insights for practitioners. The implemented models as well as the tailored algorithms could pave the way for developing some decision support software on AGV scheduling for port operators.

The remainder of this paper is organized as follows. The related works are reviewed in the next section. Section 3 addresses the problem background. In Section 4, an MILP model is proposed for the AGV scheduling problem under the battery charging-swapping mixed mode. The CG based solution method with some acceleration tactics is designed in Section 5 for solving the model. Some experiments are performed in Section 6 to validate the proposed methodology and derive some managerial implications. Conclusions are then outlined in the last section.

2. Related works

As an important symbol of green ports, ACTs are leading a major change in port infrastructural development. The choice of the appropriate type of container transportation equipment is an important decision at strategic level for the design of green ports (Vis and de Koster, 2003). AGVs are responsible for the transportation of containers between the quay side and the yard side in ACTs. For a comprehensive overview on operations management of AGVs, see Vis (2006), Le-Anh and de Koster (2006). This paper investigates a joint optimization problem on AGV battery charging/swapping decisions and scheduling decisions. Specifically, this paper proposes an optimization model and algorithm for integrating the battery charging-swapping mode and assigning tasks to AGVs. Some thresholds of battery power in the charging and swapping modes are also taken into account. In the remainder of this section, the related works are mainly reviewed from the following two perspectives.

The first stream is about the studies on AGV scheduling problems in container ports. The scheduling optimization problems of AGVs include vehicle routing and task assignment according to time and space dimensional characteristics. Kim and Bae (2004) schedule AGVs by utilizing information about where and when future tasks will be delivered. An MILP model is constructed and solved using a heuristic algorithm. Similarly, Luo and Wu (2015) aim to minimize the ships' dwell time at berths, an MILP model is constructed to determine the dispatching rules of AGVs and container storage locations. Based on previous studies, Xiang et al. (2021) analyze container flows and adopt theoretical analysis to determine system capacity considering some factors such as traffic congestion, unbalanced task assignment, container batch arrival, and different berth and yard layouts. Besides the above AGV routing problems, the integrated configuration with other equipment in the yard is also an issue that needs to be addressed in the design of container terminals. Yang et al. (2018) propose a bi-level programming model for integrated scheduling of equipment coordination and AGV routing, and compare the rolling horizon procedure with bi-level genetic algorithm based on the congestion prevention rule (CPR-BGA), and the results show that CPR-BGA is highly effective. The above research focuses on the scheduling of AGVs, while our research is of higher complexity, and can be regarded as a variant of the pickup and delivery problem with time windows. The pioneering work on the problem started with Krishnamurthy et al. (1993). They solve this problem by using CG. Adamo et al. (2018) study the routing and scheduling problems of AGVs in ports. They also determine the vehicle paths and speeds on each arc of the path while meeting the time windows. For the split delivery vehicle routing problem (SDVRP), Zhu et al. (2023) proposed a forest-based solution representation for solving basic SDVRP, multidepot SDVRP, and the SDVRP with time windows. Apart from the above studies, we also consider the configuration of the AGVs' battery charging piles and swapping stations; in addition, the battery charging piles/swapping stations are allowed to be visited for multiple times in our study.

The second stream is about the studies on the operation optimization of AGVs considering battery charging or swapping activities. Mousavie et al. (2017) construct a multi-objective scheduling model intending to minimize the makespan and number of AGVs considering the AGVs' battery charging. They propose a hybrid particle swarm optimization and generic algorithm (GA) to solve the model, which is further evaluated and validated by the simulation with Flexsim software. Kabir and Suzuki (2018) develop simulation models to investigate how the duration of battery charging of AGVs can be varied to improve the flexibility of manufacturing systems. Then they further investigate the impact of different routine techniques of AGV battery management on the productivity of manufacturing facilities (Kabir and Suzuki, 2019). Liu et al. (2019) propose a model with the objective of minimizing the makespan, the number of AGVs used, and the power consumption by considering the charging tasks and changeable speed of AGVs. They integrate an adaptive GA with a multi-adaptive GA to solve the model. Ma et al. (2021) present a simulation approach to deploy the battery charging stations and battery-powered AGVs. However, simulation studies cannot deliver optimal solutions for specific scenarios. Dang et al. (2021) consider the AGV's limited battery capacity and battery threshold, an MILP model is formulated to minimize the tardiness costs and travel costs of AGVs. A hybrid adaptive large neighborhood search (ALNS) with local search method is developed to solve the model. The study of Singh et al. (2022) is similar to the problems studied by Dang et al. (2021), and they propose an ALNS algorithm and a linear program to solve industry-size instances. Boccia et al. (2023) study the AGV scheduling problem with battery constraints (ASP-BC) and propose a three-step heuristic algorithm to solve the problem. In this paper, the AGV battery charging time is fixed and does not depend on the remaining power. Li et al. (2023a) construct a two-stage stochastic programming model with a joint scheduling problem of battery swapping and task operation, and propose a simulation-based sample average approximation (SAA) combined with ant colony optimization (ACO) algorithm to solve the problem. Similar to the above, the charging time in this paper is set to a fixed value. Li et al. (2023b) model an integrated scheduling problem containing AGVs, automated quay cranes and yard cranes with the objective of minimizing the AGV charging cost and penalty cost associated with makespan of all containers. A decomposition iterative algorithm is designed to solve the problem. Unlike the previous studies, the charging time in this paper is no longer a fixed value but is set as a variable related to the remaining power. Lu et al. (2024) focus on how to replace traditional diesel-powered trucks with electric vehicles (EVs) in the seaport operations environment, and propose a two-stage energy-transport scheduling framework that includes vehicle task matching, EV dispatch, and energy management. There is a significant difference between our study and Lu's research in dealing with energy replenishment modes, that directly affects the complexity of the model and the practicality of the application. Lu et al. (2024) focus on the case of battery swapping, where the swapping time is fixed. In contrast, our study considers a more complex hybrid charging and swapping mode, where the charging time of AGVs is variable and is affected by factors such as remaining power and charging pile availability. Xiao et al. (2024) construct an MIP model for the AGV scheduling problem based on the consideration of hybrid modes of battery swapping and charging, aiming to minimize the energy and delay costs, and design an adaptive large neighborhood search (ALNS) algorithm to solve the model. This has some similarities with our work, but there are also significant differences. In contrast to Xiao's paper, where the variability of charging time, the queuing of AGVs at charging/swapping locations, and the layout and optimization of charging/swapping locations are not specifically mentioned, our study explicitly considers the variability of charging time and specifically considers the queuing of AGVs at charging/swapping locations, and focuses on the layout optimization of charging/swapping locations. In actual port operations, the availability of charging/swapping locations and the flexibility of charging time are crucial to the overall scheduling efficiency, and our study takes into account the complexities of the actual operation and is closer to actual situations. The aforementioned literatures investigate either battery charging or battery swapping for electric vehicles separately; while this paper considers a charging-swapping mixed mode. Electric vehicles can choose a more time-saving charging way according to their actual situation; and the configuration of charging piles and swapping stations is also investigated by experiments. Table 1 illustrates some representative works on AGV scheduling in ports and the comparisons with this study from some specific problem features.

Compared with the existing relevant literature, this study differs from them significantly. First, this paper conducts a comprehensive study on AGV task assignment and sequencing, battery charging/swapping activity scheduling, and battery charging piles/swapping stations deployment; an MILP model for the battery charging-swapping mixed mode is formulated for the first time, and this model involves several detailed factors like AGVs waiting at facilities. Second, this paper designs and implements an enhanced CG algorithm embedded with a novel label setting algorithm to solve the model efficiently. Unlike most of the existing label setting algorithm, our algorithm is more complex, and each vertex could be visited multiple times. Last but not the least, based on the real cases of Xiamen Port Yuanhai ACT and Shanghai Yangshan Deepwater Port Phase IV ACT, this paper conducts numerical experiments and sensitivity analysis to deliver some managerial insights for practitioners.

Table 1: Comparison with relevant research on scheduling optimization of AGVs in container ports

Papers	**Problem features**									**Algorithms**
	Task assignment	*Task sequence*	*Time window*	*Pickup and delivery*	*Threshold battery power*	*Battery charging*	*Battery swapping*	*Variable charging time*	*Waiting at charging pile*	
Krishnamurthy et al. (1993)	√	√		√						CG
Kim and Bae (2004)	√	√	√							Heuristic
Luo and Wu (2015)	√	√		√						GA
Mousavie et al. (2017)	√	√		√		√				Hybrid GA-PSO
Yang et al. (2018)	√	√	√							Bi-level GA
Adamo et al. (2018)	√	√	√	√						Branch-and-bound
Kabir and Suzuki (2018)	√			√		√		√		Simulation
Liu et al. (2019)	√	√				√				Multi-adaptive GA
Xiang et al. (2021)	√									N/A
Ma et al. (2021)	√				√	√			√	Simulation
Dang et al. (2021)	√	√	√	√	√	√				ALNS + local search
Singh et al. (2022)	√	√	√	√	√	√				ALNS
Boccia et al. (2023)	√	√				√				Three step matheuristic
Li et al. (2023a)	√	√			√		√			SAA+ACO
Li et al. (2023b)	√	√			√	√		√		Decomposition iterative
Lu et al. (2024)	√	√	√	√			√		√	Deep Q-network
Xiao et al. (2024)	√	√	√	√	√	√	√			ALNS
This paper	√	√	√	√	√	√	√	√	√	CG + label-setting

3. Problem background

Suppose there is a fleet of AGVs in an ACT for fulfilling a batch of tasks. The core decision of the problem is to assign tasks to AGVs and also sequence the tasks assigned to each AGV, as shown in Figure 2. For each AGV, we know its location and remaining battery power before starting the fulfillment of tasks in the batch. During the process of fulfilling the sequence of tasks, an AGV may go to a charging pile (or a battery swapping station) for charging (or swapping) its battery between the end time of fulfillment for a task and the start time of fulfillment for the next task; and AGV may perform the battery charging or swapping activities multiple times in the process. For each AGV, the start time and the facilities for charging (or swapping) its battery are also the decisions that should be made in the problem. The locations of these facilities (charging piles and battery swapping stations) are known in this problem.

In this problem, a task is either to load or unload a container between quay side and yard side of the ACT. The duration for fulfilling each task, the locations of each task's origin and destination are known parameters. For each task i, there is an expected time window with the earliest time a_i and the latest time b_i; and the task i's scheduled start and end time should be within the time window $[a_i, b_i]$. If the operation for a task is delayed and violates the expected time window, a penalty will be reflected in the model's objective because it may cause a delay in the departure of a vessel and may incur some potential economic loss.

Some details on the battery charging and swapping activities are taken into account in this problem. Each AGV's battery power (unit is A·h) is consumed at a constant rate (unit is A·h/t); the time duration to charge an AGV for a rise of one A·h battery power is also a known parameter. In addition, it is noted that the time duration to swap a battery for an AGV is a constant for the battery swapping mode. In this problem, the minimum and the maximum power for AGV battery are denoted by $\dot{q}$ and $\ddot{q}$, respectively. When an AGV completes a task, the battery power is checked whether it is less than $\dot{q}$; if so, the AGV will be charged or swapped with a full power battery. When an AGV i is arranged to be charged at a charging pile and there is another AGV j is being charged at the pile, the AGV i should wait there until the AGV j is charged to the maximum power $\ddot{q}$; then the AGV j leaves the pile and the AGV i starts its charging activity.

Besides $\dot{q}$ and $\ddot{q}$, another important parameter for the AGV battery is the threshold battery power for charging, denoted by $h^{\mathbb{C}}$; it means when remaining power is larger than $h^{\mathbb{C}}$, the battery should not be charged. For the battery swapping mode, the threshold parameter is denoted by $h^{\mathbb{S}}$; it means when remaining power is larger than $h^{\mathbb{S}}$, the battery should not be swapped. Overall, the above parameters imply that when a battery power is in the ranges $(0, \dot{q}]$, $(\dot{q}, h^{\mathbb{C}}]$, and $(h^{\mathbb{C}}, \ddot{q}]$, the battery must be charged, could be charged or not, and should not be charged, respectively. In the battery swapping mode, when a battery power is in the ranges $(0, \dot{q}]$, $(\dot{q}, h^{\mathbb{S}}]$, and $(h^{\mathbb{S}}, \ddot{q}]$, the battery must be swapped, could be swapped or not, and should not be swapped, respectively. The reason for establishing the parameters $h^{\mathbb{C}}$ and $h^{\mathbb{S}}$ in this problem lies in that these thresholds exist in the realistic battery management. The battery life is influenced by the charge and discharge times of battery, thus setting proper thresholds could increase the battery life for AGVs and further reduce the operation cost for the ACTs' port operators.

Last but not the least, the time is discretized into time steps in this problem. Then the model for the problem is formulated on the basis of discrete time. For example, we could set one time step as one minute. Theoretically speaking, the setting of one time step depends on not only the size of the model's solution space under the problem' realistic scale, but also the computation capacity of the used computer.

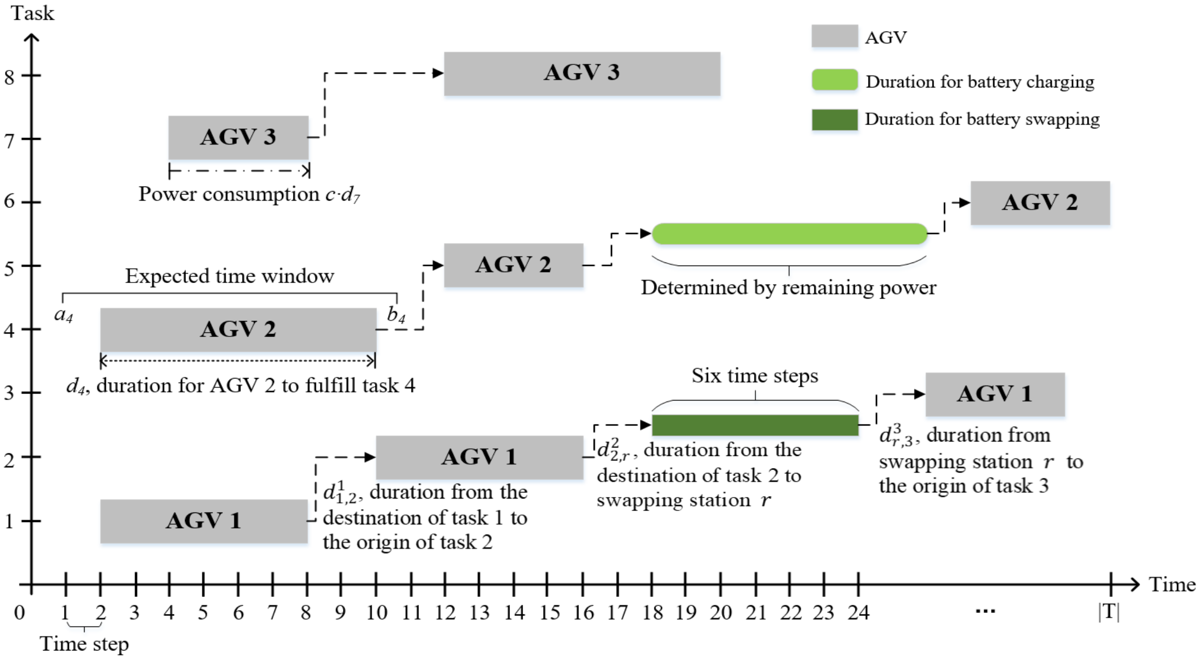


Figure 2: Example of AGV task assignment considering battery charging and swapping

Before presenting the mathematical models, the assumptions for this study are outlined as follows. First, we assume an AGV's travel speed is a constant, which is not influenced by the weight of container that the AGV is carrying. Second, we assume an AGV's battery is charged to the maximum power (i.e., $\ddot{q}$) every time (Schneider et al., 2018; Zou et al., 2018; Xiang and Liu, 2021). We can relax the second assumption by further defining another decision variable to denote the battery power, to which an AGV is charged in one of its charging activities. For the interest of simplicity of the modeling and ease of explanation, we make that assumption.

4. Model formulation

An MILP model on AGV scheduling under the battery charging-swapping mixed mode is proposed in this section. Before formulating the model, the sets, parameters, and decision variables are defined first. For the ease of understanding, this section uses the Latin alphabet and the Greek alphabet to denote parameters and decision variables, respectively. In the battery charging-swapping mixed mode, AGVs with low power could be scheduled to charging piles to charge the batteries or be scheduled to swapping stations to exchange the batteries.

Indices and sets

- V set of AGVs, indexed by v
- I set of tasks, indexed by i and j
- $I^{+}, I^{-}, \hat{I}$ sets of tasks and dummy tasks; $x(v)$ and $x'(v)$ are defined as the dummy start task and end task of AGV v, respectively; $I^{-} = I \cup \{x(v)\}$, $I^{+} = I \cup \{x'(v)\}$, $\hat{I} = I \cup \{x(v), x'(v)\}$
- T set of time steps, indexed by t
- $R^{\mathbb{c}}, R^{\mathbb{s}}$ set of charging piles or swapping stations. $R = R^{\mathbb{c}} \cup R^{\mathbb{s}}$, indexed by r

Parameters

- $d_{i,j}^{1}$ duration (in time steps) for an AGV travelling from task i's destination to task j's origin
- $d_{i,r}^{2}$ duration for an AGV travelling from task i's destination to charging or swapping location r
- $d_{r,j}^{3}$ duration for an AGV travelling from charging or swapping location r to task j's origin
- d_i duration for an AGV fulfilling task i
- $[a_i, b_i]$ expected time window for fulfilling task i; a_i and b_i are the earliest and latest time for task i
- q_v initial power of AGV v; the unit is A·h (Ampere-Hour)
- $\dot{q}, \ddot{q}$ minimum, maximum power for AGV battery
- $h^{\mathbb{c}}$ threshold battery power; when remaining power is larger than $h^{\mathbb{c}}$, the battery should not be charged
- $h^{\mathbb{s}}$ threshold battery power; when remaining power is larger than $h^{\mathbb{s}}$, the battery should not be swapped
- c consumption rate of AGV battery power; the unit is A·h/timestep
- f duration to charge an AGV for a rise of one A·h battery power; the unit is timestep/A·h
- s duration for swapping battery for an AGV v
- M a sufficiently large positive number

Decision variables

$\alpha_{i,v}$ binary, equals one if task i is assigned to AGV v , otherwise zero

$\mu_{i,j,v}$ binary, equals one if tasks i, j are assigned to AGV v and fulfilled consecutively, otherwise zero. It should be noted that if the AGV fulfills task i, and then goes to a charging pile for charging, then fulfills task j, the variable $\mu_{i,j,v}$ also equals one

$\beta_{i,v}$ binary, equals one if AGV v's battery needs be charged or swapped after completing task i, otherwise zero

$\delta_{i,v,r}$ binary, equals one if AGV v completes task i and then go to location r for charging or swapping, otherwise zero

$\tau_{i,j,r}$ binary, equals one if both AGVs for fulfilling tasks i, j are charged or swapped at location r and the charging or swapping for the tasks i's AGV is immediately followed by the charging or swapping for the tasks j's AGV, otherwise zero

$\gamma_{i,v}^{\mathbb{C}}$ binary, equals one if AGV v's battery needs be charged after completing task i, otherwise zero

$\gamma_{i,v}^{\mathbb{S}}$ binary, equals one if AGV v's battery needs be swapped after completing task i, otherwise zero

$\theta_{i,v}$ integer, duration for charging or swapping after AGV v's completion of task i

$\varepsilon_{i,v}$ integer, start time step when AGV v begins fulfilling task i

$\omega_{i,v}$ integer, end time step when AGV v completes fulfilling task i

$\zeta_{i,r}$ integer, start time step when an AGV begins charging or swapping after the AGV's completion of task i

$\lambda_{i,v}$ nonnegative, AGV v's remaining battery power (A·h) after AGV v's completion of task i

ρ_i nonnegative, delay of task i's completion from the latest time in its expected time window, i.e., b_i

Mathematical model

$[\boldsymbol{\mathcal{M}_{Mix}}]$ Minimize $\sum_{i\in I}\rho_i$ (1-1)

subject to:

$\rho_i \geq \sum_{v\in V}\omega_{i,v} - b_i$ $\forall i \in I$ (1-2)

$\sum_{v\in V}\varepsilon_{i,v} \geq a_i$ $\forall i \in I$ (1-3)

$\sum_{v\in V}\alpha_{i,v} = 1$ $\forall i \in I$ (1-4)

$\varepsilon_{i,v} + \omega_{i,v} \leq \alpha_{i,v}M$ $\forall i \in I, \forall v \in V$ (1-5)

$\sum_{j\in I^+}\mu_{i,j,v} = \sum_{j\in I^-}\mu_{j,i,v} = \alpha_{i,v}$ $\forall i \in I, \forall v \in V$ (1-6)

$\sum_{i\in I^+}\mu_{x(v),i,v} = \sum_{i\in I^-}\mu_{i,x'(v),v} = 1$ $\forall v \in V$ (1-7)

$$\varepsilon_{j,v} \geq \omega_{i,v} + d^1_{i,j} + 1 - M(1 - \mu_{i,j,v}) \qquad \forall i, j \in \hat{I}: i \neq j, \forall v \in V \qquad (1\text{-}8)$$

$$\varepsilon_{j,v} \geq \zeta_{i,r} + \theta_{i,v} + d^3_{r,j} - M\big(2 - \mu_{i,j,v} - \delta_{i,v,r}\big) \qquad \forall i, j \in \hat{I}: i \neq j, \forall v \in V, \forall r \in R \qquad (1\text{-}9)$$

$$\textstyle\sum_{v \in V} \omega_{i,v} = \sum_{v \in V} \varepsilon_{i,v} + d_i - 1 \qquad \forall i \in \hat{I} \qquad (1\text{-}10)$$

$$\theta_{i,v} + M\big(1 - \beta_{i,v}\big) \geq f\big(\ddot{q} - \lambda_{i,v} + \textstyle\sum_{r \in R} c d^2_{i,r} \delta_{i,v,r}\big) - M(1 - \gamma^{\mathbb{C}}_{i,v}), \qquad \forall i \in I, \forall v \in V \qquad (1\text{-}11)$$

$$\theta_{i,v} + M\big(1 - \beta_{i,v}\big) \geq s - M(1 - \gamma^{\mathbb{S}}_{i,v}) \qquad \forall i \in I, \forall v \in V \qquad (1\text{-}12)$$

$$\gamma^{\mathbb{C}}_{i,v} + \gamma^{\mathbb{S}}_{i,v} = \beta_{i,v} \qquad \forall i \in I, \forall v \in V \qquad (1\text{-}13)$$

$$\gamma^{\mathbb{C}}_{i,v} \leq \textstyle\sum_{r \in R^{\mathbb{C}}} \delta_{i,v,r} \qquad \forall i \in I, \forall v \in V \qquad (1\text{-}14)$$

$$\gamma^{\mathbb{S}}_{i,v} \leq \textstyle\sum_{r \in R^{\mathbb{S}}} \delta_{i,v,r} \qquad \forall i \in I, \forall v \in V \qquad (1\text{-}15)$$

$$\lambda_{j,v} \leq \ddot{q} - c(d^3_{r,j} + d_j) + M(2 - \delta_{i,v,r} - \mu_{i,j,v}) \qquad \forall i, j \in \hat{I}: i \neq j, \forall v \in V, \forall r \in R \qquad (1\text{-}16)$$

$$\lambda_{j,v} \geq \ddot{q} - c(d^3_{r,j} + d_j) - M(2 - \delta_{i,v,r} - \mu_{i,j,v}) \qquad \forall i, j \in \hat{I}: i \neq j, \forall v \in V, \forall r \in R \qquad (1\text{-}17)$$

$$\lambda_{j,v} \leq \lambda_{i,v} - c(d^1_{i,j} + d_j) + M\beta_{i,v} + M(1 - \mu_{i,j,v}) \qquad \forall i, j \in \hat{I}: i \neq j, \forall v \in V \qquad (1\text{-}18)$$

$$\lambda_{j,v} \geq \lambda_{i,v} - c\big(d^1_{i,j} + d_j\big) - M\beta_{i,v} - M(1 - \mu_{i,j,v}) \qquad \forall i, j \in \hat{I}: i \neq j, \forall v \in V \qquad (1\text{-}19)$$

$$M\beta_{i,v} \geq \dot{q} - \lambda_{i,v} - M(1 - \alpha_{i,v}) \qquad \forall i \in I^-, \forall v \in V \qquad (1\text{-}20)$$

$$M(1 - \gamma^{\mathbb{C}}_{i,v}) \geq \lambda_{i,v} - h^{\mathbb{C}} \qquad \forall i \in I^-, \forall v \in V \qquad (1\text{-}21)$$

$$\beta_{i,v} \leq \alpha_{i,v}, \qquad \forall i \in I^-, \forall v \in V \qquad (1\text{-}22)$$

$$\zeta_{i,r} + M\big(1 - \delta_{i,v,r}\big) \geq \omega_{i,v} + d^2_{i,r}, \qquad \forall i \in I^-, \forall v \in V, \forall r \in R \qquad (1\text{-}23)$$

$$\zeta_{j,r} \geq \zeta_{i,r} + \theta_{i,v} - M\big(1 - \tau_{i,j,r}\big), \qquad \forall i, j \in I^-: i \neq j, \forall r \in R \qquad (1\text{-}24)$$

$$\zeta_{i,r} \leq M \textstyle\sum_{v \in V} \delta_{i,v,r}, \qquad \forall i \in I^-, \forall r \in R \qquad (1\text{-}25)$$

$$\beta_{i,v} = \textstyle\sum_{r \in R} \delta_{i,v,r}, \qquad \forall i \in I^-, \forall v \in V \qquad (1\text{-}26)$$

$$\tau_{i,j,r} + \tau_{j,i,r} \leq \textstyle\sum_{v \in V} \delta_{i,v,r}, \qquad \forall i, j \in I^-: i \neq j, \forall r \in R \qquad (1\text{-}27)$$

$$\tau_{i,j,r} + \tau_{j,i,r} \geq \textstyle\sum_{v \in V} \delta_{i,v,r} + \sum_{v \in V} \delta_{j,v,r} - 1, \qquad \forall i, j \in I^-: i \neq j, \forall r \in R \qquad (1\text{-}28)$$

$$M\big(1 - \gamma^{\mathbb{S}}_{i,v}\big) \geq \lambda_{i,v} - h^{\mathbb{S}}, \qquad \forall i \in I, \forall v \in V \qquad (1\text{-}29)$$

$$\alpha_{i,v}, \mu_{i,j,v}, \beta_{i,v}, \delta_{i,v,r}, \tau_{i,j,r}, \gamma^{\mathbb{C}}_{i,v}, \gamma^{\mathbb{S}}_{i,v} \in \{0,1\}, \qquad \forall i, j \in \hat{I}: i \neq j, \forall v \in V, \forall r \in R \qquad (1\text{-}30)$$

$$\theta_{i,v}, \varepsilon_{i,v}, \omega_{i,v}, \zeta_{i,r} \in T, \qquad \forall i \in \hat{I}, \forall v \in V, \forall r \in R \qquad (1\text{-}31)$$

$$\lambda_{i,v}, \rho_i \geq 0, \lambda_{x(v),v} = q_v \qquad \forall i \in \hat{I}, \forall v \in V. \qquad (1\text{-}32)$$

Objective (1-1) minimize the sum of delay time for all the tasks. Constraints (1-2) calculate the delay time for each task. Constraints (1-3) state each task's start time should be no earlier than its time window's earliest time. Constraints (1-4) guarantee each task is assigned to one AGV. Constraints (1-5) ensure that if a task is not assigned to an AGV, the variables that denote the task's start and end time by the AGV are set as zero. Constraints (1-6) connect the two consecutive tasks undertaken by one AGV. Constraints (1-7) address the start and end of the task sequence for each AGV. Constraints (1-8) link task i's end handling time and the start handling time of task j that immediately follows task i in the task sequence of AGV v. It is noted that the part "$+1$" in the constraints is due to the discrete time steps (Geismar et al. 2008). Constraints (1-9) link the start charging time after task i and the start handling time of task j that follows task i and charging duration of AGV v. Constraints (1-10) connect each task's start and end handling time. It is noted that the part "-1" in the constraints is also due to the discrete time steps. Constraints (1-11) and (1-12) denote the battery charging duration and swapping duration, respectively. Constraints (1-13) connect and involve the two binary variables ($\gamma_{i,v}^{\mathbb{C}}$, $\gamma_{i,v}^{\mathbb{S}}$) into one binary variable ($\beta_{i,v}$). Constraints (1-14) and (1-15) link the binary variables in charging mode and swapping mode to the previously defined variable $\delta_{i,v,r}$ within the charging facility set $R^{\mathbb{C}}$ and the swapping facility set $R^{\mathbb{S}}$, respectively. Constraints (1-16) and (1-17) calculate an AGV's remaining power after completing a task if the AGV is charged before executing the task. Constraints (1-18) and (1-19) calculate an AGV's remaining power after completing a task if the AGV is not charged before executing the task. Constraints (1-20) ensure that an AGV must be charged if its remaining power is lower than the minimum power. Constraints (1-21) ensure that AGV should not be charged if its remaining power is higher than the threshold. Constraints (1-22) link two binary decision variables. Constraints (1-23) state that an AGV's start charging time equals to the end handling time for a task plus the travel time from the task's destination to the charging pile's location. Under conditions where charging/swapping locations are limited, it is common for two or more vehicles to queue up for charging at the same charging/swapping locations. Constraints (1-24) connect the start charging time of two AGVs that are arranged to be charged at the same pile successively. Constraints (1-25) ensure that the binary variable $\zeta_{i,r}$ is zero if the AGV that executes task i is not assigned to pile r for charging. Constraints (1-26) guarantee that if the binary variable $\beta_{i,v}$ equals one that means AGV v should be charged after completing task i, one charging pile should be assigned to the AGV. Constraints (1-27) and (1-28) state the successive relationship between two AGVs for their charging activities at the same pile. More specifically, Constraints (1-27) ensure that when an AGV fulfilling task i does not go to charging pile r after completing the task, both the binary variables $\tau_{i,j,r}$ and $\tau_{j,i,r}$ should be zero; Constraints (1-28) guarantee that, when both tasks i and j are completed, the two AGVs fulfilling them are assigned to the same pile r for charging, one of binary variables $\tau_{i,j,r}$ and $\tau_{j,i,r}$ should be one. Constraints (1-29) ensure that when the remaining power is higher than $h^{\mathbb{S}}$, a battery cannot be swapped. Constraints (1-30), (1-31), and (1-32) define the domains of binary, integer, and nonnegative variables, respectively; $\lambda_{x(v),v} = q_v$ indicates that the initial power of an AGV is $\lambda_{x(v),v} = q_v$.

Proposition 1: Finding an optimal solution for the problem is NP-hard.

Proof: See Appendix A.1. ■

5. CG based solution method

For solving the above proposed model $\mathcal{M}_{Mix}$ in large-scale instances, a CG based solution method is designed and implemented in this section.

5.1. Set-partitioning based model formulation

Before formulating the set-partitioning based model, the plans (columns) for AGVs are defined first. The index of a plan for an AGV (e.g., AGV v) is denoted by $\wp$; here $\wp \in \mathbb{P}$, i.e., the set of all the plans for all the AGVs.

A plan $\wp$ mainly contains information on the assignment of tasks to AGV v, choosing which facility for charging or swapping its battery as well as the duration at the chosen facility (a charging pile or a swapping station). The cost of the plan $\wp$ is denoted by $\mathcal{C}_{\wp,v}$, which equals to the sum of the delays for all the tasks assigned to AGV v. Each plan's cost can be calculated in advance according to the information contained in the plan, which is specified by the following parameters.

Parameters in columns

$x_{i,\wp}$ binary, equals one if task i is assigned to an AGV in the plan $\wp$, otherwise zero

$y_{i,r,t,\wp}$ binary, equals one if an AGV completes task i and then goes to facility r for starting battery charging or swapping at time step t in the plan $\wp$, otherwise zero

Decision variable

$\pi_{\wp,\mathrm{v}}$ binary, equals one if plan $\wp$ is chosen for AGV v, otherwise zero

Mathematical model

$$[\boldsymbol{\mathcal{MP}}] \qquad Minimize \ \sum_{v \in V} \sum_{\wp \in \mathbb{P}} \mathcal{C}_{\wp,v} \pi_{\wp,\mathrm{v}} \tag{2-1}$$

subject to:

$$\sum_{\wp \in \mathbb{P}} \pi_{\wp,\mathrm{v}} \le 1 \qquad \forall v \in V \tag{2-2}$$

$$\sum_{v \in V} \sum_{\wp \in \mathbb{P}} x_{i,\wp} \pi_{\wp,\mathrm{v}} = 1 \qquad \forall i \in I \tag{2-3}$$

$$\sum_{i \in I} \sum_{v \in V} \sum_{\wp \in \mathbb{P}} y_{i,r,t,\wp} \pi_{\wp,\mathrm{v}} \le 1 \qquad \forall r \in R, \forall t \in T \tag{2-4}$$

$$\pi_{\wp,\mathrm{v}} \in \{0,1\} \qquad \forall v \in V, \wp \in \mathbb{P}. \tag{2-5}$$

Objective (2-1) minimizes the sum of the delays for all the fulfilled tasks. Constraints (2-2) ensure each AGV is assigned with at most one plan. Constraints (2-3) state that each task is included in a plan which has been chosen. Constraints (2-4) guarantee each battery charging or swapping facility is occupied by at most one AGV all the time. Constraints (2-5) define the decision variable.

In addition, Jepsen et al. (2008) introduced subset-row inequalities in the VRPTW for the first time and proved that subset-row inequalities are valid for the set partitioning model. Luo et al. (2021) also used the inequalities in a cable-routing problem in solar power plants. For any $B \subseteq I$ and an integer $p \in \mathbb{N}$ such that $0 < p \leq |B|$, the valid inequalities obtained by the Chvatal-Gomory rounding of the partitioning constraints are as follows, also called Subset Rows Inequalities (SRI):

$$\sum_{v \in V} \sum_{\wp_v \in \mathcal{P}_v} \left\lfloor \frac{\sum_{i \in B} x_{i,\wp_v}}{p} \right\rfloor \pi_{\wp_v} \leq \left\lfloor \frac{|B|}{p} \right\rfloor, \qquad \forall B \subseteq I, |B| < |I|. \tag{2-6}$$

We enumerated all the subset-row inequalities and added them to the model since $|B|$ and p are small. In order to have a good tradeoff between the quality of the lower bound obtained by adding inequality (2-6) to the RMP and the complexity of the resulting pricing problem, we decided, through extensive experimentation, to use the subset row inequalities with $|B| = 5$ and $p = 5$ (See Appendix E.2), i.e.:

$$\sum_{v \in V} \sum_{\wp_v \in \mathcal{P}_v} \left\lfloor \frac{\sum_{i \in B} x_{i,\wp_v}}{5} \right\rfloor \pi_{\wp_v} \leq 1, \qquad \forall B \subseteq I,\ |B| = 5. \tag{2-7}$$

As the usually practice of the CG, we relax the binary variable $\pi_{\wp_v}$ in the above master problem $\mathcal{MP}$, and restrict the problem by constructing a subset of columns rather than using the full set of all the possible columns. This study proposes an adaptive large neighborhood search (ALNS) based method for generating initial set of columns. The details on this ALNS based initialization method are elaborated in Appendix B.1 and the pseudocode is provided in Appendix C.1. After solving the above linear-relaxed restricted master problem (LR-RMP), the dual variables for the constraints (2-2), (2-3), (2-4) and (2-6) are obtained for constructing the pricing problem that generate new columns iteratively:

η_v^1 dual variables for Constraints (2-2), $\forall v \in V$

η_i^2 dual variables for Constraints (2-3)，$\forall i \in I$

$\eta_{r,t}^3$ dual variables for Constraints (2-4)，$\forall r \in R, \forall t \in T$

η_B^4 dual variables for Constraints (2-7)，$\forall B \subseteq I, |B| = 5.$

5.2. Pricing problem

Based on the above dual variables obtained from the LR-RMP, the objective of dual linear restricted master problem (DL-RMP) should be formulated as follows.

$$\text{Maximize } \sum_{v \in V} \eta_v^1 + \sum_{i \in I} \eta_i^2 + \sum_{r \in R} \sum_{t \in T} \eta_{r,t}^3 + \sum_{B \subseteq I, |B|=5} \eta_B^4 \tag{2-8}$$

$$s.t.\ \eta_v^1 + \sum_{i \in I} \eta_i^2 x_i + \sum_{i \in I} \sum_{r \in R} \sum_{t \in T} \eta_{r,t}^3 y_{i,r,t} + \sum_{B \subseteq I, |B|=5} \eta_B^4 \left\lfloor \frac{\sum_{i \in B} x_i}{5} \right\rfloor \leq \mathcal{C}_{\wp_v},$$

$$v \in V, \wp_v \in \mathcal{P}_v. \tag{2-9}$$

As aforementioned, the index for a plan (or column) is $\wp_v$; and $\wp_v \in \mathcal{P}_v$. The set of all the columns

$\mathbb{P}$ is partitioned into $|V|$ subsets $\mathcal{P}_v$, $\mathbb{P} = \cup_{v \in V} \mathcal{P}_v$. Therefore, the pricing problem (PP) could also be divided into $|V|$ pricing subproblems, denoted by $\mathcal{PP}_v$, each of which corresponds to one AGV and generates columns for the set $\mathcal{P}_v$. For the interest of simplicity, we remove the index v and p_v from the subscripts of variables (x_i, $y_{i,r,t}$, $\mu_{i,j}$, β_i, $\gamma_i^{\mathbb{C}}$, $\gamma_i^{\mathbb{S}}$, θ_i, λ_i, ε_i, ω_i, $\delta_{i,r}$) in the following defined pricing subproblem $\mathcal{PP}_v$; we redefine variable C as the objective value of a plan (column), i.e., the sum of the delays for all the tasks assigned to the AGV v. The parameters are the same as the previously defined ones; and above-mentioned solved values for the dual variables $\eta^1, \eta_i^2, \eta_{r,t}^3, \eta_B^4$ are also the parameters in the $\mathcal{PP}_v$.

The details on the pricing subproblem for AGV v are elaborated in Appendix B.2.

5.3. Label-setting algorithm for solving pricing subproblems

For some large-scale instances, the above formulated pricing subproblem $\mathcal{PP}_v$ cannot be solved by CPLEX within a short time, which makes the solution process by the CG based solution method be very time consuming. Therefore, this subsection proposes a label-setting algorithm to solve the pricing subproblem $\mathcal{PP}_v$ efficiently so as to accelerate the solution process. The label-setting algorithm is a dynamic programming based algorithm which is usually used to solve the shortest path problems. Loui (1983) combine dynamic programming and dominance, which paves the way for the label-setting algorithm. Based on the above work, Desrochers and Soumis (1988) consider the time window constraint and propose the generalized label-setting algorithm. Murthy and Sarkar (1996) improve the label-setting procedure to solve the shortest path problem. In recent years, the label-setting algorithms are widely used in the shortest path problems with capacity or time window constraints (Prescott-Gagnon et al., 2010; Zhu & Wilhelm, 2012; Xue et al. 2016; Luo et al., 2017 and 2019; Capelle et al. 2019; Dabia et al., 2019; Koyuncu and Yavuz, 2019; Huang et al., 2021). Here we elaborate the three main components contained in our proposed label setting algorithm, i.e., the label definition, label extension function, and the dominance rules. The algorithmic flowchart as well as its contributions are summarized at the end of this subsection.

The label-setting algorithm is an accurate algorithm used to solve the shortest path problem, which is essentially similar to dynamic programming algorithm. Dynamic programming is commonly used to solve optimization problems by decomposing the original problem into several subproblems and storing the solutions of the subproblems to avoid repeated calculations, thereby efficiently finding the global optimal solution. The label-setting algorithm also follows this idea, gradually expanding the path and recording the current optimal solution to ensure termination after finding the optimal solution. The correctness of the label-setting algorithm is based on its ability to systematically explore all possible paths and update the shortest path estimate to each node at each step. The algorithm starts from the starting node and assigns a label to each node, which records the estimated shortest path to that node and the information of the previous nodes.

During the iteration process, the algorithm selects node i with the shortest distance label from the unprocessed node set X and transfers it to the processed node set S. According to the definition of the algorithm, the distance label $d(i)$ of node i represents the shortest path length from the initial node to node i, and this path does not contain any nodes in set S as internal nodes. Assuming there exists a path P from the initial node to node i, which includes some nodes in set S as internal nodes. Path P can be decomposed into two parts: path P^1 does not contain any nodes in set S as internal nodes, but terminates at a node k in set S; path P^2 is the path from node k to node i. According to the inductive assumption, the length of path P^1 is no shorter than $d(k)$, and since node i is the node with the shortest distance label selected from set X, $d(k) \geq d(i)$. This means that the length of path P^1 is no shorter than $d(i)$. Meanwhile, since all arc lengths are nonnegative, the length of path P^2 is also nonnegative. Therefore, the length of the entire path P is no shorter than $d(i)$, which proves that $d(i)$ is indeed the shortest path length from node i to the initial node. After the algorithm permanently marks node i, the distance labels of some nodes in set S may change because node i may become an internal node in the tentative shortest path of these nodes. After node i is permanently marked, the algorithm will check all arcs (i, j) starting from node i. If the distance label $d(j)$ of a node j is found to be longer than the path length $d(i) + c_{ij}$ from node i to node j, the algorithm will update the distance label of node j to $d(i) + c_{ij}$ and set node i as the predecessor node of node j. This update operation ensures that after each iteration, the distance label of each node in set S still represents the shortest path length from the initial node to that node, and the internal nodes of this path only contain nodes in set S. As the algorithm continues to iterate, the set S gradually expands until it includes all nodes. At this point, the distance label of each node represents the shortest path length from the initial node to that node, and the algorithm terminates and outputs the optimal solution (Ahuja et al. 1995). Due to the algorithm selecting the current optimal label for expansion at each step, and the label values continuously approaching the true shortest path length during the iteration process, it can ensure that the algorithm finds the global optimal solution at the end.

5.3.1. Label definitions

Before defining a label, the vertex in the label setting algorithm is defined first. Different from the traditional label setting algorithm, there are two types of vertices in our algorithm: one is named by task vertex, which denotes a loading/unloading task for AGVs (i.e., the original meaning of tasks in the problem description); the other is named by battery vertex, which denotes an activity of battery charging or swapping for AGVs at one certain charging pile or one certain battery swapping station. In this problem background, each task is executed (task vertex is visited) for only once, while a charging pile (or a battery swapping station) could be visited by one AGV for more than once. Because there are two types of vertices with different features, our label setting algorithm is more complex than the traditional one.

Our label setting algorithm is to solve the pricing subproblem $\mathcal{PP}_v$, which is oriented to the AGV v. Here each label is defined for a partial path for the AGV from its dummy start vertex to the current vertex. More specifically, a label is denoted by $L_{i,l,n} = (time_{i,l,n}, cost_{i,l,n}, q_{i,l,n}, pre_{i,l,n})$; the label $L_{i,l,n}$ corresponds a partial path indexed by n, which originates from the AGV's dummy start vertex, goes through l vertices, and ends at the current vertex indexed by i. Each label mainly contains four aspects of information, i.e., the current time, the cost related to the objective of the pricing subproblem $\mathcal{PP}_v$, the remaining power of the AGV's battery, and the vertex i's previous vertex. The details on the label definition are listed as follows.

i or j	index of the current or the next vertex; $i, j \in Z_+, i, j \in \left[1, \left|\hat{I}\right| + |R| \times |I|\right]$. If $i \leq \left|\hat{I}\right|$, it is an aforementioned task vertex; if $i > \left|\hat{I}\right|$, it is a battery vertex
l	number of vertices contained in the p artial path that corresponds to the label; $l \in Z_+, l \in \left[1, \left|\hat{I}\right| + |R| \times |I|\right]$
n or m	index of the current partial path or the next path; $m, n \in Z_+, m, n \in [1, |I| - 1 + 2 \times |R|]$
$time_{i,l,n}$	time after visiting vertex i in the partial path n that contains l vertices; the time corresponds to the variable ω_i in the pricing subproblem $\mathcal{PP}_v$
$cost_{i,l,n}$	reduced cost after visiting vertex i in the partial path n that contains l vertices; it corresponds to the objective of the pricing subproblem $\mathcal{PP}_v$
$q_{i,l,n}$	remaining power of the AGV's battery after visiting vertex i in the partial path n; it corresponds to the variable λ_i in the pricing subproblem $\mathcal{PP}_v$
$pre_{i,l,n}$	vertex i's previous vertex in the partial path n

5.3.2. Label extension function

As the usual practice of the traditional label setting algorithm, the label extension functions should be defined. However, the extension functions in our label setting algorithm need be defined from two aspects, one is for the extension of a task vertex and the other is for the extension of a battery vertex. Given a type of vertex, the next vertex in the extension may also have two options. Moreover, if the next vertex is a battery vertex, the extension functions for battery charging are different from the functions for battery swapping, which implies two more cases need be further investigated separately. Thus the label extension in our algorithm is more complex than the traditional way significantly.

5.3.2.1 Label extension for task vertices

The current vertex i is a task vertex, which means $i \leq |\hat{I}|$. The current label is $L_{i,l,n} = (time_{i,l,n}, cost_{i,l,n}, q_{i,l,n}, pre_{i,l,n})$. Suppose $p(L_{i,l,n})$ denotes the label $L_{i,l,n}$'s partial path; $P(L_{i,l,n})$ denotes the set of all the possible partials path from the current vertex i to the AGV's dummy end vertex; the partial path $p(L_{i,l,n})$ combined with one element (partial path) inside the set $P(L_{i,l,n})$ constitutes a complete path for the AGV.

For its extension (the next vertex in the partial path), there are also two options, which depend on the remaining power of the AGV's battery. More specifically, when the remaining power is larger than the minimum value, i.e., $q_{i,l,n} \geq \dot{q}$, the next extended vertex is still a task vertex; otherwise, the next vertex should be a battery vertex. The label extension functions in two cases are listed Appendix B.3.

5.3.2.2 Label extension for battery vertices

The current vertex i is a battery vertex, which means $|\hat{I}| \leq i \leq |\hat{I}| + |R| \times |I|$. In the current label $L_{i,l,n} = (time_{i,l,n}, cost_{i,l,n}, q_{i,l,n}, pre_{i,l,n})$, $q_{i,l,n} = \ddot{q}$, i.e., the battery power is full, which implies the next vertex j can be only a task vertex.

For all vertex j, $j \leq |\hat{I}|$, and the condition "$\ddot{q} - c(d^3_{pile_{i},j} + d_j) > 0$" holds, which implies the AGV's battery power can support it to travel to vertex j's location and complete the task j. From the above candidate vertices, we select the vertex j in the label extension, $j = \arg\min_{j \leq |\hat{I}|}((\omega_j - b_j)^+ - \eta_j^2)$, which implies if the vertex j is the next one, the reduced cost is minimized. A new label $L_{j,l+1,m}$ is created by extending from the label $L_{i,l,n}$. The index of the above newly created partial path is denoted by $\overline{m}_{j,l+1}$. The extension functions are listed as follows.

$$time_{j,l+1,\overline{m}_{j,l+1}} = \max\{time_{i,l,n} + d^3_{pile_{i},j} + 1, a_j\} + d_j - 1 \quad (2\text{-}10)$$

$$cost_{j,l+1,\overline{m}_{j,l+1}} = cost_{i,l,n} + (\omega_j - b_j)^+ - \eta_j^2 \quad (2\text{-}11)$$

$$q_{j,l+1,\overline{m}_{j,l+1}} = \ddot{q} - c(d^1_{i,j} + d^3_{pile_{i},j}) \quad (2\text{-}12)$$

$$pre_{j,l+1,\overline{m}_{j,l+1}} = L_{i,l,n} \quad (2\text{-}13)$$

5.3.3. Dominance rules

When the scale of problem instances grows, the number of labels increases significantly. Here we propose some dominance rules to eliminate the labels that cannot generate the optimal solution, which can avoid the time-consuming enumeration and accelerate the solution process of the label setting algorithm.

Proposition 2: For two partial paths m and n, both end at vertex i, with labels denoted by $L_{i,l,m} = (time_{i,l,m}, cost_{i,l,m}, q_{i,l,m}, pre_{i,l,m})$ and $L_{i,l,n} = (time_{i,l,n}, cost_{i,l,n}, q_{i,l,n}, pre_{i,l,n})$, respectively, if the following conditions hold, then label $L_{i,l,n}$ dominates label $L_{i,l,m}$, and so $L_{i,l,m}$ is a discard label, because solutions generated by $L_{i,l,m}$ can also be generated by $L_{i,l,n}$.

$$time_{i,l,n} \leq time_{i,l,m} \tag{2-14}$$

$$cost_{i,l,n} \leq cost_{i,l,m} \tag{2-15}$$

$$q_{i,l,n} \geq q_{i,l,m} \tag{2-16}$$

Proof: See Appendix A.2. ■

5.3.4. Summary of the label setting algorithm

Based on the above three components, a brief flow of the label setting algorithm is stated as follows.

Step 1: Initialization, $time_{i,l,n} = +\infty$, $q_{i,l,n} = \ddot{q}$, $pre_{i,l,n} = -1$, $cost_{i,l,n} = +\infty$, $\mathcal{C}_i = +\infty$, $l = 0$, $\forall i \in I$;

Step 2: If $l = |I| \times |R| + 2$, $\mathcal{C}_i^n$ is the minimal reduced cost, the algorithm stops; otherwise, go to Step3;

Step 3: According to the dominance rules in Proposition 2, we judge all the labels $L_{i,l,n}$ related to vertex i. If a label is dominated, it is discarded, and then go to Step 2; otherwise, go to Step 4.

Step 4: For all the vertices j, which are accessible from all vertices i, we create new labels $L_{j,l+1,m}$ according to the label extension functions; $time_{j,l+1,m}$, $cost_{j,l+1,m}$, $q_{j,l+1,m}$ and $pre_{j,l+1,m}$ are updated. Then $l = l + 1$, go to Step2.

A detailed procedure of the algorithm is illustrated in Appendix C.2 due to the limitation of space.

An efficient solution method for solving the pricing problem is the key for the CG-based solution methods. This study proposes a novel label setting algorithm for solving the pricing subproblem $\mathcal{PP}_v$ efficiently. Its advantage is validated by experiments which will be addressed in the next section. Here the contributions of our label setting algorithm could be summarized as follows.

There are two types of vertices in this algorithm, while the traditional label setting algorithms usually contain one type of vertex. For the battery vertices, i.e., one of the two types, there are further two options (battery charging or swapping activities). All of these bring more complexities for label definition (e.g., more types of vertices, and multi-dimensional information contained in a label), extension functions' formulation (e.g., more possible combinations for the extension between two vertices with respect to their types), and analysis on dominance rules (e.g., more conditions).

The majority of the existing label setting algorithms are oriented to the traveling salesman problem or the shortest path problem, in which each vertex is visited for once. While in our problem, each charging pile or battery swapping station can be visited by one AGV for more than once. Thus our problem may be more complex than the existing related problems.

The time complexity of the label-setting algorithm mainly depends on the number of nodes and edges in the graph. In the worst case, the algorithm needs to traverse all nodes and edges multiple times to update the label values of each node. Specifically, if there are n nodes and m edges in the graph, the time complexity of the algorithm is usually $O(n^2 + mn)$ (the n^2 term comes from the comparison and selection operations between nodes, and mn term comes from the traversal and label update operations of edges). According to the settings in this study, there are $|\hat{I}| + |R| \times |I|$ nodes and $|I| - 1 + 2 \times |R|$ edges in the graph, so its time complexity can be simplified as $O\left(|\hat{I}|^2(|R| + 1)^2\right)$.

5.4. Strategies for selecting plans for AGVs

After the above procedure of CG, the column pool (i.e., plan pool) with feasible AGV plans is obtained. Because the solution for the LR-RMP (i.e., π_{p_v}) may be fractional rather than integer, we propose a process that includes the following five steps to obtain a feasible integer solution for the RMP on the basis of the above obtained column pool and the LR-RMP's solution. In other words, we need to select a plan for each AGV from the pool.

Step 1: The values in the right-hand side of Constraints (2-3) and (2-4) are defined as AA_i and $BB_{r,t}$, respectively; and we initialize them as: $AA_i = 1, BB_{r,t} = 1$. The set of plans is defined as PP, which is initialized as a null set. The set of tasks that have not been assigned to AGVs is defined as RR, which is initialized as a full set of all the tasks.

Step 2: We execute the column generation procedure, which is elaborated in Sections 5.1 ~ 5.3, and obtain the solution, i.e., values of decision variable π_{p_v}. If all the values of π_{p_v} are integers, the whole algorithm terminates; otherwise, we construct a plan pool that include the plans p_v whose values of π_{p_v} are not zero.

Step 3: We select a plan from the above pool according to one strategy (Strategy 2 or Strategy 3 elaborated later), and add it into the set PP; and then we delete the tasks, which are related to the plan, from the set RR.

Step 4: We update the values of AA_i and $BB_{r,t}$ according to the above selected plan. More specifically, if task i in the plan has been assigned, we update $AA_i = 0$; if the AGV begins charging or swapping battery at the facility r in time step t, and the duration for the battery charging and swapping is θ_i, we update $BB_{r,tt} = 0$ for $\forall tt \in [t, t + \theta_i]$.

Step 5: We repeat the above Step 2 ~ Step 4 until the set RR is null. Then the plans in the set PP is the solution for the problem.

In the above process, two strategies (Strategy 2 or Strategy 3) are mentioned in Step 3 (Wang et al., 2018). Besides the two strategies, another widely used strategy (named Strategy 1 in the remainder of this study) is also adopted. The three strategies are outlined as follows. Some experiments will also be conducted to compare the performance among these three strategies.

Strategy 1: We use the CPLEX, which evokes a branch and cut method, to solve the integer programming model of the RMP rather than the model of LR-RMP, which is used in the CG.

Strategy 2: The plan $\wp_v$ with the highest value of $\pi_{\wp_v}$, which is solved in the CG and may be fractional, is selected for AGV v. If two plans' values of $\pi_{\wp_v}$ are equal to each other, the plan with the lower value of $C_{\wp_v}$ is selected for AGV v.

Strategy 3: The plan $\wp_v$ with the lowest value of $C_{\wp_v}$ is assigned to AGV v. If two plans' values of $C_{\wp_v}$ are equal to each other, the plan with the higher value of $\pi_{\wp_v}$ is selected for AGV v.

6. Numerical experiments

To verify the effectiveness of the proposed algorithm, numerical experiments including performance analysis and sensitivity analysis are conducted in this section. The solution quality and processing time of various feasible strategies, solving pricing subproblems by CPLEX or label-setting, with or without using the inequalities are compared. The sensitivity analysis is used to reveal the impact of the six parameters on the optimal solutions under different modes. The Central Processing Unit (CPU) of the experimental platform used in this study is Intel Xeon E5-2643 v4 3.4Ghz. The models are programmed in C# (Visual Studio 2019); the CPLEX's version is 12.6.1.

6.1. Experimental settings

In the experiments, we set $\ddot{q} = 500$ A·h as the maximum battery capacity of the fully charged AGV. In this state, the AGV can be used for eight hours. The consumption rate of AGV battery power c is about 1.04 A·h/timestep (timestep is measured in minutes). When the remaining power is lower than the minimum safe operating power $\dot{q}$, assuming $\dot{q} = 0.2 \times \ddot{q}$, battery charging or swapping must be considered. It takes about two hours to fully charge the AGV, so the duration f to charge an AGV for a rise of one A·h battery power is about 0.24 timestep. The duration for swapping a battery for an AGV s is six timesteps. The parameters $d^1_{i,j}, d^2_{i,r}, d^3_{r,j}$, and $d_{i,v}$ are determined according to the distances between two locations, which could be tasks' origins and destinations, the locations of charging piles and swapping stations, and the speed of AGVs. For each task, the origin and the destination are determined according to the locations of the related yard block and the related quay crane in a realistic port; for a loading (or an unloading) task, the origin (or destination) and the destination (or origin) are its related yard block and its related quay crane, respectively. For the expected time windows of tasks (i.e., $[a_i, b_i]$), the earliest time (i.e., a_i) is generated according to the data pattern of the task arrival time at the port operations system in a realistic port; the length of the time window (i.e., $b_i - a_i$) is set as the travel time from a task's origin to its destination and plus a certain amount of slack time.

6.2. Evaluation of the components embedded in the algorithm

The previous section proposes some components that are embedded in the CG-based algorithm so as to improve the algorithmic performance. More specifically, Section 5.4 proposes three strategies to construct feasible solutions; Section 5.3 proposes a label-setting algorithm to solve the PPs in more efficient way than using the CPLEX to solve the PPs; Section 5.1 also proposes some valid inequalities for algorithmic acceleration. For evaluating effectiveness of the above three components, the experimental investigations are conducted in the following.

Table 2: Comparison among three strategies of constructing feasible solutions

Instances		**CG + Strategy 1**		**CG + Strategy 2**		**CG + Strategy 3**	
Scale	ID	F_{S1}	t_1(s)	F_{S2}	t_2(s)	F_{S3}	t_3(s)
$\lvert I\rvert = 6, \lvert V\rvert = 2$	1-1	8	53	8	31	**8**	**25**
	1-2	7	33	7	32	**7**	**30**
	1-3	**18**	**33**	18	42	18	81
	1-4	**16**	**41**	16	46	16	52
	1-5	5	39	**5**	**26**	5	39
	1-6	**9**	**30**	9	46	9	41
	1-7	**13**	**36**	13	99	13	88
	1-8	**4**	**24**	4	45	4	38
$\lvert I\rvert = 8, \lvert V\rvert = 2$	2-1	**21**	**2109**	21	4662	21	5094
	2-2	**17**	**1173**	17	5430	17	4181
	2-3	**14**	**2418**	14	3202	14	3841
	2-4	**34**	**1610**	34	5297	34	4436
	2-5	**21**	**3522**	21	7296	21	5464
	2-6	**15**	**2340**	15	2441	15	3945
	2-7	**16**	**1654**	16	2122	16	2103
	2-8	**19**	**5298**	19	6283	19	6290
Average		**15**	**1276**	15	2319	15	2234

In Section 5.4, we propose three strategies of constructing feasible solutions after the CG procedure. Experiments are conducted to compare the performance of the three strategies with respect to the objective value and the solution time. The results are presented in Table 2, which demonstrates that the objective values in all three columns of F_{S1}, F_{S2} and F_{S3} are identical for each instance. As the instance scale increases, the solution time of the algorithm increases significantly. In the most instances, the strategy 1 uses the shortest solution time. Besides, strategy 3 performs better than strategy 2 with respect to the solution time. Therefore, in the remaining experiments of this study, our algorithm adopts the strategy 1 to construct feasible solution (i.e., select plans for AGVs) in the CG-based solution method. It is noted that CPLEX rather than the label-setting algorithm is used to solve the $\mathcal{PP}_v$ in the CG-based solution method in Section 6.2.

Furthermore, considering a label-setting algorithm proposed in Section 5.3, the performance of solving the pricing problems $\mathcal{PP}_v$ by label-setting algorithm and CPLEX is compared (See Appendix E.1). The results show that the solutions obtained by the above methods for solving $\mathcal{PP}_v$ are consistent, while compared with solving $\mathcal{PP}_v$ by CPLEX, using label-setting algorithm can effectively shorten the solution time of the CG method by about 90.08% on average.

Finally, considering some valid inequalities for the RMP proposed in Section 5.1, the performance of the two CG-based methods with or without using inequalities are compared (See Appendix E.2). The results show that the use of inequalities does not affect the objective value of solutions, but can further shorten the solution time of the CG-based method.

By summarizing the above three series of experiments, the most efficient combination of the components embedded in the algorithm is the strategy 1 for constructing feasible solution, the label-setting algorithm for solving the pricing problems, and the inequalities used in the RMP. Then, the algorithm used in the remainder of this study will be equipped with this setting.

Last but not the least, the proposed the CG-based method is applicable for real world problems although the problem scale of the above experiments is a bit small, which contains just eight tasks, i.e., $|I| = 8$. For example of the instances with $|I| = 8$ in the lower half part of Table 7 (See Appendix E.2), the average solution time is about 16 seconds. Taking one hour as a time span, an average of 651 TEU needs to be completed in Shanghai port and 108 TEU in Xiamen port (for detailed calculation process please see section 6.4). Taking eight tasks as one batch, 651 TEUs should be divided into 81 batches for sensitivity analysis, which takes about 1296 seconds (1296=81×16) for total computation time; and 108 TEUs should be divided into 14 batches requiring around 224 seconds (224=14×16) for total computation time. Both the cases could be solved within one hour, which validates the algorithm can handle the two ports' realistic workloads of one hour. It implies our proposed algorithm is applicable for the real-world context of some container mega-ports.

6.3. Evaluation of solution quality by comparing with CPLEX

After validating the best algorithmic setting, the quality of the solutions solved by the algorithm should be evaluated by comparing with the optimal solution, which can be obtained by using the CPLEX to solve the original model directly. The results in Table 3 further validate the advantages of the CG-based method on the basis of comparison of objective value and solution time.

Table 3: Comparison of the CG-based solution method with CPLEX

Instance Scale	ID	CPLEX F_{CPLEX}	t_{CPLEX}(s)	CG-based method F_{CG}	t_{CG}(s)	$\Delta_{F_{CG}}$	$\Delta_{t_{CG}}$
$\lvert I\rvert = 6, \lvert V\rvert = 2$	7-1	8	4	8	6	0.00%	50.00%
	7-2	7	3	7	6	0.00%	100.00%
	7-3	18	4	18	7	0.00%	75.00%
	7-4	16	3	16	6	0.00%	100.00%
	7-5	5	3	5	5	0.00%	66.67%
	7-6	9	4	9	4	0.00%	0.00%
	7-7	13	3	13	6	0.00%	100.00%
	7-8	4	3	4	4	0.00%	33.33%
$\lvert I\rvert = 8, \lvert V\rvert = 2$	8-1	21	75	21	15	0.00%	-80.00%
	8-2	17	91	17	17	0.00%	-81.32%
	8-3	14	55	14	15	0.00%	-72.73%
	8-4	34	570	34	18	0.00%	-96.84%
	8-5	21	273	21	18	0.00%	-93.41%
	8-6	15	246	15	17	0.00%	-93.09%
	8-7	16	99	16	11	0.00%	-88.89%
	8-8	19	478	19	20	0.00%	-95.82%
$\lvert I\rvert = 10, \lvert V\rvert = 2$	9-1	31	>7200	32	42	3.23%	——
	9-2	35	>7200	36	91	2.86%	——
	9-3	48	>7200	48	34	0.00%	——
	9-4	52	>7200	55	40	5.77%	——
	9-5	43	>7200	44	38	2.33%	——
	9-6	30	>7200	31	159	3.33%	——
	9-7	40	>7200	42	30	5.00%	——
	9-8	37	>7200	39	144	5.41%	——
Average		23	——	23.5	31	1.16%	

Notes: F_{CPLEX} and F_{CG} denote the objective value of the solution obtained by CPLEX and the CG-based method, respectively. t_{CPLEX} and t_{CG} denote the solution time of CPLEX and the CG-based method, respectively. $\Delta_{F_{CG}} = (F_{CG} - F_{CPLEX})/F_{CPLEX}$, $\Delta_{t_{CG}} = (t_{CG} - t_{CPLEX})/t_{CPLEX}$.

In the former two scales of Table 3, the CG-based method can obtain the solutions with a gap of 0.00% from the optimal solutions in a short time. With the increase of the scale, the solution time of CPLEX become very long; for the third scale of Table 3, the CPLEX cannot solve the instances within two hours, while our proposed CG-based method can obtain the solutions within 160 seconds. The average gap of our method's solution from the solution obtained by CPLEX at two hours is about 1.16% on average; but our method's solving process is very fast, the solution time is significantly shorter than two hours. The results show that our proposed CG-based method is effective and fast for solving the original model.

6.4. Sensitivity analysis

For delivering some managerial insights, sensitivity analysis is conducted on the basis of some real-world ports' environment so as to investigate the influence of different parameters on the solutions. In addition, two simplified models containing only single mode (i.e., battery charging mode or battery swapping mode) are presented in Appendix D. The sensitivity analysis considers three different modes, i.e., battery charging mode (e.g., Xiamen Port Yuanhai Terminal), battery swapping mode (e.g., the Fourth Phase Terminal of Shanghai Yangshan Deepwater Port), and battery charging-swapping mixed mode (e.g., deploying battery swapping stations in Xiamen Port, or deploying battery charging piles in Shanghai Port).The sensitivity analysis is performed on the following five parameters, i.e., duration to charge an AGV for a rise of one A·h battery power (f), duration for swapping battery for an AGV (s), threshold battery power for charging (h^{c}), threshold battery power for swapping (h^{s}), consumption rate of AGV battery power (c). Our objective is to explore the potential of these models in improving the operational efficiency of AGVs and optimizing the overall logistics efficiency of the port through in-depth analysis of the actual situation of Xiamen Port and Shanghai Port under different parameter adjustments. The aim is to gain management insights on how to balance investment in charging/swapping facilities, layout of battery charging/swapping facilities, and AGV energy management strategies through these specific case studies, so as to provide practical decision support for port managers and help them develop more efficient and economical solutions based on the operational characteristics and future development needs of the ports.

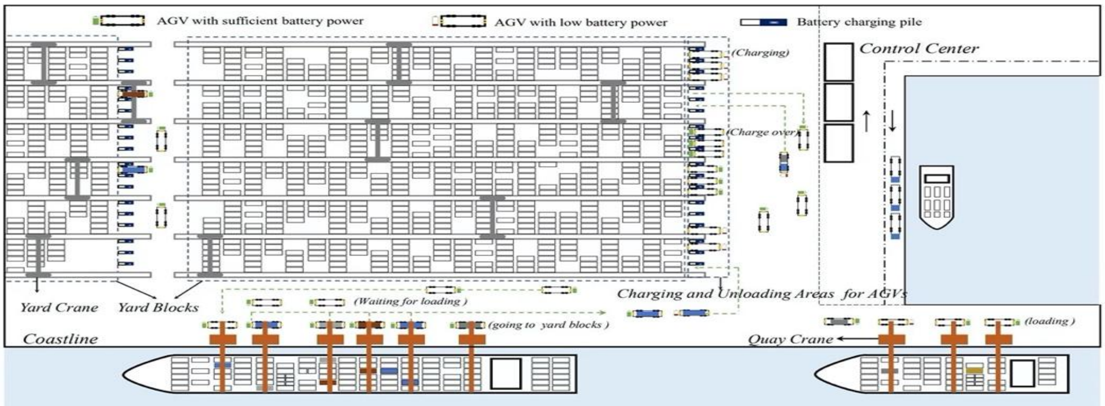


Figure 3: Layout diagram of Xiamen Port Yuanhai ACT

The Xiamen Port Yuanhai ACT is taken as an example to conduct sensitivity analysis for the battery charging mode. The Xiamen Port Yuanhai ACT is the first ACT in China. In the world, it is also the first ACT in which the yard blocks are parallelly arranged along the shoreline. The shoreline of the Yuanhai ACT is 447 meters long, with a designed annual throughput of 950,000 TEUs and one deep-water berth. There are 18 AGVs in the yard, and 24 battery charging piles on the right side of the storage area. The AGVs can be charged to improve their endurance while waiting for the yard cranes' operations. The layout diagram is shown in Figure 3. To simplify the calculation, we take one hour as a time span when designing the experiment. About 110 TEUs should be handled within one hour (108 TEU ≈ 950,000 TEU/year ÷ 365 days ÷ 24 hours). In many real-world transportation situations, scheduling involves batching decisions to provide faster work processing (Cakici et al. 2014). We take eight tasks (TEUs) as one batch for sensitivity analysis, and set the average delay of each task's completion from its expected time window in the 14 batches (14 batch ≈ 108 TEU ÷ 8 TEU/batch) as the objective value; we set the variation of one of the above five parameters (f, s, $h^{\mathbb{C}}$, $h^{\mathbb{S}}$, c) as the horizontal coordinate. The reason for setting eight tasks as one batch is explained at the end of Section 6.2; in short, we need to ensure the solution time for one batch is less than the fulfillment time for one batch of tasks.

Shanghai Yangshan Deepwater Port Phase IV ACT is taken as an example to conduct sensitivity analysis for the battery swapping mode. The Shanghai Yangshan ACT is the largest ACT in the world. The shoreline is 2,350 meters long with 26 quay cranes, and its annual throughput was 5.7 million TEUs in 2021. Currently, there are 50 AGVs in the yard, and it is expected to increase to 130 AGVs in the future. Restricted by the layout of the yard, no more than two battery swapping stations could be located in one yard. The schematic diagram of the layout is shown in Figure 4. Similar to the experiment design of Xiamen port, we also take one hour as a time span. About 660 TEUs should be handled within one hour (651 TEU ≈5,700,000 TEU/year ÷ 365 days ÷ 24 hours). We also take eight tasks as one batch for sensitivity analysis, and set the average delay of each task's completion from its expected time window in 81 batches (81 batch ≈ 651 TEU ÷ 8 TEU/batch) as the vertical coordinate; we also use one of the previous mentioned five parameters as the horizontal coordinate.

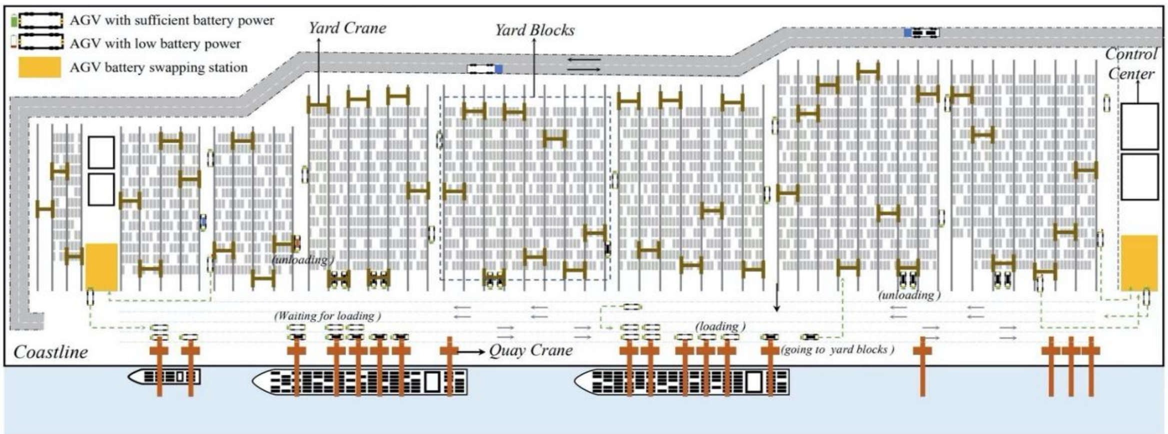


Figure 4: Layout diagram of Shanghai Yangshan Deepwater Port Phase IV ACT

In addition, the mixed mode of battery charging and swapping is also investigated here. More specifically, we construct two scenarios: one is to build battery swapping stations in Xiamen Port, which is named as *mixed mode 1* in the following experiments; the other is to build battery charging piles in Shanghai Port, which is named as *mixed mode 2* in the following experiments. Based on the realistic schematic diagram of the two ports' layouts, we suppose two new battery swapping stations are built at the southeast corner of the yard in Xiamen Port, and 40 battery charging piles with around 60 meters interval on average are built at the south side of the yard in Shanghai Port.

6.4.1. Results on the influence of battery power consumption rate (reflected by c)

The first series of sensitivity analysis is to investigate the impact of battery power consumption rate (c), which is an important performance indicator for the batteries used by an ACT's AGVs. Figure 5 describes the impact of consumption rate c on average delay of one task's completion from its expected time window under four different modes. All the curves in the four modes reveal an overall upward trend with the increase of c. According to Constraints (1-16) ~ (1-19), c has a negative effect on an AGV's remaining power $\lambda_{j,v}$ before the AGV executes its next task. Hence, as c increases, $\lambda_{j,v}$ decreases; then more charging/swapping times will be incurred, which will further cause the increase of the average delay. Although all the 4 modes show an overall upward trend as depicted in Figure 5, the average delay of battery charging-swapping mixed mode 1 (building battery swapping stations in Xiamen Port) is shorter than that of the battery charging mode under the same c, and the average delay of the battery swapping mode is shorter than that of battery charging-swapping mixed mode 2 (building battery charging piles in Shanghai Port).

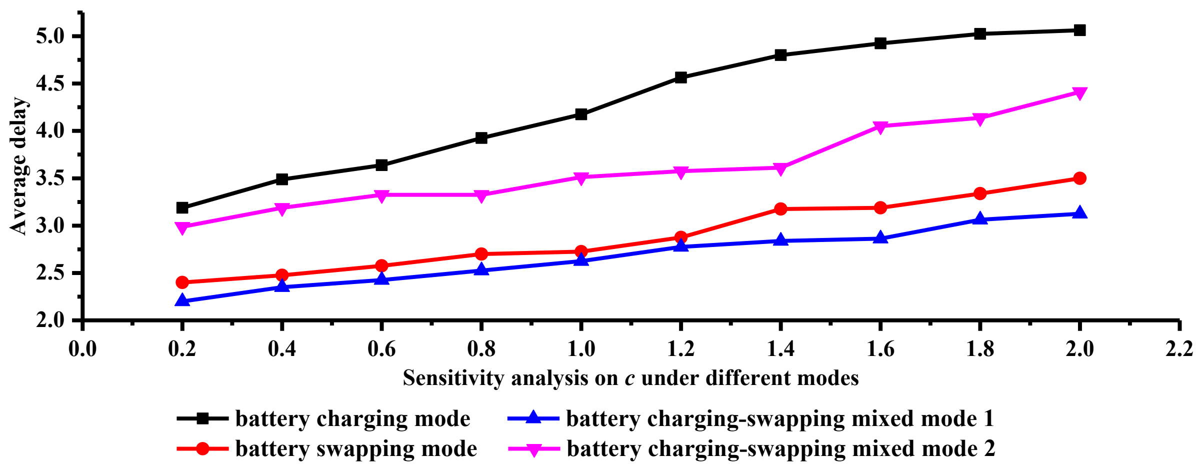


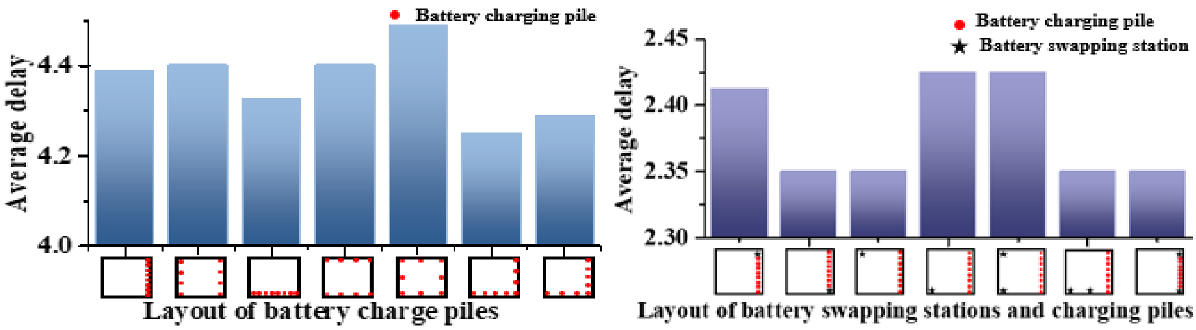
Figure 5: Sensitivity analysis on c under different modes

In addition, the sensitivity analysis on parameters $f, s, h^{\mathbb{C}}, h^{\mathbb{S}}$ are provided in Appendix E.3~E.6.

6.4.2. Results on the influence of battery swapping stations/charging piles layout in Xiamen Port

The second series of sensitivity analysis is to investigate the impact of battery charging piles layout in Xiamen Port. The two considered cases are layout of battery charging piles at different locations and establishing new battery swapping stations at specific locations in Xiamen Port (Geismar et al. 2020).

Boxes are used to represent the Xiamen Port yard. The red dots and the black five-pointed stars indicate the distribution locations of the charging stations and the newly establish battery swapping stations, respectively. Figure 6(a) illustrates the impact of different layouts of charging piles in Xiamen Port on the average delay of one task's completion from its expected time window. The first column represents the original distribution map of charging piles in Xiamen Port (distributed on the right side of the yard), while the other columns represent the distribution maps of charging piles on the left and right sides, bottom sides, top and bottom sides, surrounding areas, and bottom and right sides of the yard. This change allows AGVs to reduce charging distances while completing tasks, thereby reducing task delay time. From Figure 6(a), it can be seen that the charging piles are distributed according to the sixth column, that is, on the bottom and right sides of the yard, and there are more charging piles on the bottom side than on the right side, which can ensure the shortest average delay. Figure 6(b) illustrates the impact of adding a swapping station under the existing charging pile layout in Xiamen Port on the average delay of one task's completion from its expected time window. And it can be seen that building new battery swapping stations according to the 2nd, 3rd, 6th, and 7th columns can minimize the average delay. However, the latter two are costly since adding two new battery swapping stations is required. Therefore, using the first two layout schemes is recommended.

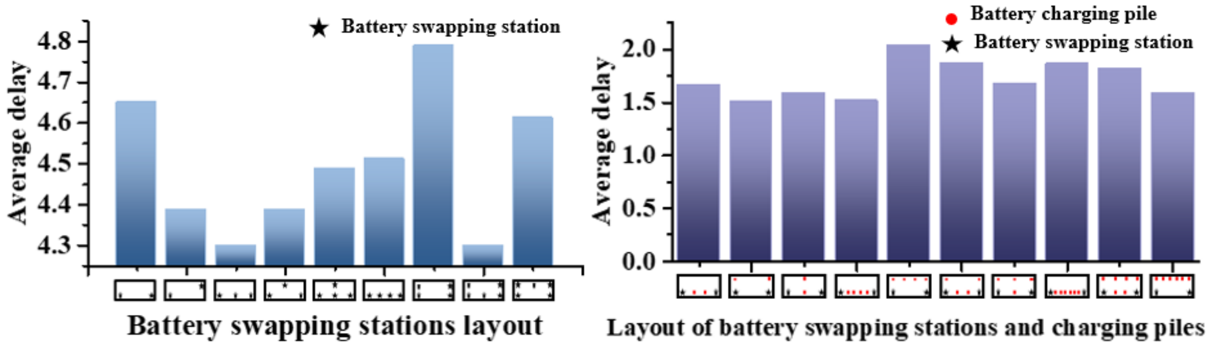


(a) battery charging piles layout at different locations (b) establishing new facilities at specific locations

Figure 6: Sensitivity analysis on battery swapping stations/charging piles layout in Xiamen Port

6.4.3. Results on the influence of battery swapping stations/charging piles layout in Shanghai Port

The third series of sensitivity analysis is to investigate the impact of battery swapping stations layout in Shanghai Port. The two considered cases are layout of swapping stations at different locations and establishing new battery charging piles at specific locations in Shanghai Port. Figure 7 (a) illustrates the impact of different layouts of existing and newly established battery swapping stations in Shanghai Port on the average delay of one task's completion from its expected time window. The first column represents the original distribution map of battery swapping stations in Shanghai Port (distributed at the left and right ends of the bottom of the yard), while the other columns represent the distribution maps of 2-5 battery swapping stations distributed at different positions on the yard. From Figure 7(a), it can be seen that according to the third column distribution of the battery swapping stations, uniformly distributing three battery swapping stations on the bottom side of the yard can ensure the shortest average delay. Figure 7(b) illustrates the impact of adding charging piles on the average delay of one task's completion from its expected time window under the existing layout of battery swapping stations in Shanghai Port. And it can be seen that the impact of adding new charging piles in Shanghai Port on the average delay is not significant. Compared to the second column, adding two charging piles at the left and right ends on the top side of the yard can minimize the average delay.



(a) battery swapping stations layout at different locations (b) establishing new facilities at specific locations

Figure 7: Sensitivity analysis on battery swapping stations/charging piles layout in Shanghai Port

7. Conclusions

For the purpose of realizing efficient operations management of AGVs in ACTs, this paper conducts a comprehensive study on AGV scheduling under battery charging-swapping mixed mode. An MILP model as well as solution method are designed and implemented for solving the model in a fast way. Numerical experiments based on two real-world ACTs are performed to validate the efficiency of the proposed methodology; some managerial implications are also obtained on the basis of the experimental results. As the AGV technology is important for reducing both the emission and the human labor works for ports, this study may be useful for practitioners under the initiatives of smart port and green port development. The main contributions of this paper are summarized as the following three perspectives.

From the perspective of problem modeling, this paper may be the first to propose a comprehensive mathematical model on AGV scheduling under the battery charging and swapping mixed mode. The assignment and sequence decisions of container loading/unloading tasks to AGVs, and arrangement of battery charging and swapping activities into the AGVs' execution of their tasks are considered by the proposed MILP models. Moreover, the models also involve some detailed factors such as the sequence decision of multiple AGVs at one charging pile or one battery swapping station, which implies the waiting phenomenon of AGVs at facilities is considered in the model. The consideration of the mixed mode is not only helpful for comparing the widely used charging and swapping modes in a quantitative way, but also potentially useful for investigating the applicability of the mixed mode in reality.

From the perspective of algorithm design, this paper designs a CG based solution method for solving the proposed MILP models. For solving the PPs efficiently, a novel label setting algorithm is proposed. Different from majority of the existing label setting algorithms that are oriented to the traveling salesman problem, our algorithm handles a more complex context, in which each vertex could be visited for more than once and two types of vertices are considered. The comparative experiment with using the CPLEX to solve the PPs demonstrates that our label setting algorithm can help the CG based method save the computation time by about 90%. In addition, some valid inequalities are also used for the RMP with the purpose of algorithmic acceleration. Experiments show that the inequalities can save about 14.35% solution time by comparing with the case without using the inequalities.

From the perspective of managerial insights, this paper conducts a series of sensitivity analysis on the basis of the realistic layouts of two representative ACTs, i.e., the first ACT in China and the largest ACT in the world; the former one adopts the battery charging mode while the latter one adopts the swapping mode. The sensitivity analysis reveals some nonintuitive insights. For example, by analyzing the re-layout of the charging and swapping facilities in Xiamen Port and Shanghai Port (including changing the location of the original charging/swapping facilities or adding new charging/swapping facilities at specific locations), it concludes that at the Xiamen Port yard, distributing charging piles on the bottom and right sides or establishing new swapping stations on the bottom-right side of the original layout, and at the Shanghai Port yard, distributing three swapping stations uniformly on the bottom side or establishing two additional charging piles on the left and right ends of the top side of the original layout appears to minimize the average delay. Under the same battery charging speed, or charging threshold, or swapping duration, the average delay of the battery charging-swapping mixed mode is always shorter than that of the single charging or swapping mode. While the average delay of battery charging-swapping mixed mode 1 (building battery swapping stations in Xiamen Port) is shorter than that of the battery charging mode under the same battery power consumption rate, and the average delay of the battery swapping mode is shorter than that of battery charging-swapping mixed mode 2 (building battery charging piles in Shanghai Port). Therefore, it is recommended that Xiamen Port operators could consider building a battery swapping station located at the bottom-right side, which can effectively reduce the task average delay and improve operational efficiency. For operators of Shanghai Port, the current battery swapping mode is more suitable under the actual situation, and there is no need to construct charging pile facilities, but a new battery swapping station in the middle of the bottom side of the yard could play a better role in reducing average delay.

In the future, the algorithmic performance could be further improved so as to solve much larger scale instances within a short solution time. More specifically, the solution time for a batch of tasks should be shorter than the time during which the batch of tasks is handled in a realistic port operation context.

Acknowledgments

This research is supported by the National Natural Science Foundation of China (Grant numbers 72025103, 72394360, 72394362, 72361137001, 72371221, and 72401179). The authors would like to express gratitude to the PolyU Maritime Data and Sustainable Development Centre (PMDC) and Research Centre for Digital Transformation of Tourism for their invaluable support in providing data and equipment. Dr. Lu Zhen and Dr. Qian Zhang contributed equally to this work and are regarded as co-first authors.

Notes on contributors

Lu Zhen is a Professor and Dean at School of Management, Shanghai University, Shanghai, China. His research interests include operations management and optimization; mixed-integer linear programming and algorithms; port operations and maritime transportation; urban logistics and supply chain management. He has served as an associate editor or an editorial board member of five journals such as Transportation Research Part B, Journal of the Operational Research Society, Computers & Operations Research; and he is also the Fellow of the Operational Research Society (U.K.). He has been awarded the National Funds for Distinguished Young Scientists, for Outstanding Young Scientists in China, the Changjiang Young Scholar in China, and etc.

Qian Zhang is an associate professor at Logistics Engineering College, Shanghai Maritime University, Shanghai, China. Her research interests include green port planning and operations, integrated energy systems, mixed integer linear programming and algorithms.

Zheyi Tan is a laboratory technician at Shanghai University. His research interests include operations management and optimization; mixed-integer linear programming and algorithms; intelligent warehouse scheduling; port operations and maritime transportation.

Shuaian Wang is a Professor at Faculty of Business, The Hong Kong Polytechnic University, Hong Kong, China. His research interests include big data in shipping, green shipping, shipping operations management, port planning and operations, urban transport network modeling, and logistics and supply chain management. He is an editor-in-chief of Cleaner Logistics and Supply Chain and Communications in Transportation Research, a co-editor-in-chief of Transportation Research Part E, an associate editor of Flexible Services and Manufacturing Journal, Transportmetrica A, and Transportation Letters, a handle editor of Transportation Research Record, an editorial board editor of

Transportation Research Part B, and an editorial board member of Maritime Transport Research.

References

Boccia M, Masone A, Sterle C, Murino T (2023) The parallel AGV scheduling problem with battery constraints: A new formulation and a matheuristic approach. *European Journal of Operational Research* 307(2): 590–603.

Cakici E, Mason SJ, Geismar HN, Fowler JW (2014) Scheduling parallel machines with single vehicle delivery. *Journal of Heuristics* 20(5): 511–537.

Capelle T, Cortes CE, Gendreau M, Rey PA, Rousseau LM (2019) A column generation approach for location-routing problems with pickup and delivery. *European Journal of Operational Research* 272(1): 121–131.

Dabia S, Ropke S, van Woensel T (2019) Cover inequalities for a vehicle routing problem with time windows and shifts. *Transportation Science* 53(5): 1354–1371.

Dang QV, Singh N, Adan I, Martagan T, van de Sande D (2021) Scheduling heterogeneous multi-load AGVs with battery constraints. *Computers & Operations Research* 136: 105517.

Desrochers M, Soumis F (1988) A generalized permanent labeling algorithm for the shortest path problem with time windows. *INFOR Information Systems and Operational Research* 26(3): 191–212.

Esfandeh T, Batta R, Kwon C (2018) Time-dependent hazardous-materials network design problem. *Transportation Science* 52(2): 454–473.

Fransoo JC, Lee CY (2013) The critical role of ocean container transport in global supply chain performance. *Production and Operations Management* 22(2): 253–268.

Geismar HN, Laporte G, Lei L, Sriskandarajah C (2008) The integrated production and transportation scheduling problem for a product with a short lifespan. *Informs Journal on Computing* 20(1): 21–33.

Geismar HN, Huang YW, Pillai SD, Sriskandarajah C, Youn S (2020) Location-Routing with Conflicting Objectives: Coordinating eBeam Phytosanitary Treatment and Distribution of Mexican Import Commodities. *Production and Operations Management* 29(6): 1506–1531.

Huang N, Li J, Zhu W, Qin H (2021) The multi-trip vehicle routing problem with time windows and unloading queue at depot. *Transportation Research Part E-Logistics and Transportation Review* 152: 102370.

Jepsen M, Petersen B, Spoorendonk S, Pisinger D (2008) Subset-row inequalities applied to the vehicle-routing problem with time windows. *Operations Research* 56(2): 497–511.

Kabir Q S, Suzuki Y (2018) Increasing manufacturing flexibility through battery management of automated guided vehicles. *Computers & Industrial Engineering* 117: 225–236.

Kabir Q S, Suzuki Y (2019) Comparative analysis of different routing heuristics for the battery management of automated guided vehicles. *International Journal of Production Research* 57(2): 624–641.

Kim K H, Bae J W (2004) A look-ahead dispatching method for automated guided vehicles in automated port container terminals. *Transportation Science* 38(2): 224–234.

Koyuncu I, Yavuz M (2019) Duplicating nodes or arcs in green vehicle routing: A computational comparison of two formulations. *Transportation Research Part E-Logistics and Transportation Review* 122: 605–623.

Krishnamurthy NN, Batta R, Karwan MH (1993) Developing conflict-free routes for automated guided vehicles. *Operations Research* 41(6): 1077–1090.

Kumawat GL, Roy D, de Koster R, Adan I (2021) Stochastic modeling of parallel process flows in intra-logistics systems: Applications in container terminals and compact storage systems. *European Journal of Operational Research* 290(1): 159–176.

Le-Anh T, de Koster M B M (2006) A review of design and control of automated guided vehicle systems. *European Journal of Operational Research* 171(1): 1–23.

Lee CY, Song DP (2017) Ocean container transport in global supply chains: overview and research opportunities. *Transportation Research Part B-Methodological* 95: 442–474.

Li L, Li Y, Liu R, Zhou Y, Pan E (2023a) A Two-stage Stochastic Programming for AGV scheduling with random tasks and battery swapping in automated container terminals. *Transportation Research Part E: Logistics and Transportation Review* 174: 103110.

Li XD, Peng Y, Tian Q, Feng T, Wang WY, Cao Z, Song XQ (2023b) A decomposition-based optimization method for integrated vehicle charging and operation scheduling in automated container terminals under fast charging technology. *Transportation Research Part E-Logistics and Transportation Review* 180: 103338.

Liu Y, Ji S, Su Z, Guo D (2019) Multi-objective AGV scheduling in an automatic sorting system of an unmanned (intelligent) warehouse by using two adaptive genetic algorithms and a multi-adaptive genetic algorithm. *Plos One* 14(12): e0226161.

Loui RP (1983) Optimal paths in graphs with stochastic or multidimensional weights. *Communications of the ACM* 26(9): 670–676.

Luo J, Wu Y (2015) Modelling of dual-cycle strategy for container storage and vehicle scheduling problems at automated container terminals. *Transportation Research Part E-Logistics and Transportation Review* 79: 49–64.

Luo Z, Liu M, Lim A (2019) A two-phase branch-and-price-and-cut for a dial-a-ride problem in patient transportation. *Transportation Science* 53(1): 113–130.

Luo Z, Qin H, Zhu W, Lim A (2017) Branch and price and cut for the split-delivery vehicle routing problem with time windows and linear weight-related cost. *Transportation Science* 51(2): 668–687.

Luo Z, Qin H, Cheng T, Wu Q, Lim A (2021) A branch-and-price-and-cut algorithm for the cable-routing problem in solar power plants. *INFORMS Journal on Computing* 33(2): 452–476.

Ma N, Zhou C, Stephen A (2021) Simulation model and performance evaluation of battery-powered AGV systems in automated container terminals. *Simulation Modelling Practice and Theory* 106: 102146.

Mak HY, Rong Y, Shen Z (2013) Infrastructure planning for electric vehicles with battery swapping. *Management Science* 59(7): 1557–1575.

Mousavi M, Yap HJ, Musa SN, Tahriri F, Dawal SZM (2017) Multi-objective AGV scheduling in an FMS using a hybrid of genetic algorithm and particle swarm optimization. *Plos One* 12(3): e0169817.

Meng Q, Wang S, Andersson H, Thun K (2014) Containership routing and scheduling in liner shipping: Overview and future research directions. *Transportation Science* 48(2): 265–280.

Murthy I, Sarkar S (1996) A relaxation-based pruning technique for a class of stochastic shortest path problems. *Transportation Science* 30(3): 220–236.

Prescott-Gagnon E, Desaulniers G, Drexl M, Rousseau LM (2010) European driver rules in vehicle routing with time windows. *Transportation Science* 44(4): 455–473.

Ravindra KA, Thomas LM, and James BO. Network flows: theory, algorithms and applications. Prentice hall, 1995.

Ropke S, Pisinger D (2006) An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. *Transportation Science* 40(4): 455–472.

Roy D, de Koster R, Bekker R (2020) Modeling and design of container terminal operations. *Operations Research* 68(3): 686–715.

Schneider F, Thonemann UW, Klabjan D (2018) Optimization of battery charging and purchasing at electric vehicle battery swap stations. *Transportation Science* 52(5): 1211–1234.

Singh N, Dang QV, Akcay A, Adan I, Martagan T (2022) A matheuristic for AGV scheduling with battery constraints. *European Journal of Operational Research* 298(3): 855–873.

Sioshansi R (2012) Modeling the Impacts of Electricity Tariffs on Plug-In Hybrid Electric Vehicle Charging, Costs, and Emissions. *Operations Research* 60(2): 506–516.

Vis I F A, de Koster R (2003) Transshipment of containers at a container terminal: An overview. *European Journal of Operational Research* 147(1): 1–16.

Vis I F A (2006) Survey of research in the design and control of automated guided vehicle systems. *European Journal of Operational Research* 170(3): 677–709.

Wang K, Zhen L, Wang S, Laporte G (2018) Column Generation for the Integrated Berth Allocation, Quay Crane Assignment, and Yard Assignment Problem. *Transportation Science* 52(4): 812–834.

Wang T, Wang X, Meng, Q (2018) Joint berth allocation and quay crane assignment under different carbon taxation policies. *Transportation Research Part B-Methodological* 117: 18–36.

Wang T, Du Y, Fang D, Li Z (2020) Berth allocation and quay crane assignment for trade-off between service efficiency and operating cost considering carbon emission taxation. *Transportation Science* 54(5): 1307–1331.

Widrick RS, Nurre SG, Robbins MJ (2018) Optimal policies for the management of an electric vehicle battery swap station. *Transportation Science* 52(1): 59–79.

Wu Y, Wang SA, Zhen L, Laporte G (2023) Integrating operations research into green logistics: A review. *Frontiers of Engineering Management* 10: 517–533.

Xiang X, Liu C (2021) Modeling and analysis for an automated container terminal considering battery management. *Computers & Industrial Engineering* 156: 107258.

Xue L, Luo Z, Lim A (2016) Exact approaches for the pickup and delivery problem with loading cost. *Omega* 59: 131–145.

Yang Y, Zhong M, Dessouky Y, Postolache O (2018) An integrated scheduling method for AGV routing in automated container terminals. *Computers & Industrial Engineering* 126: 482–493.

Zhong LS, Zeng ZL, Huang ZK, Shi XW, Bie YM (2024) Joint optimization of electric bus charging and energy storage system scheduling. *Frontiers of Engineering Management* 11: 676–696.

Zhong M, Yang Y, Dessouky Y, Postolache O (2020) Multi-AGV scheduling for conflict-free path planning in automated container terminals. *Computers & Industrial Engineering* 142: 106371.

Zhu W, Ao Z, Baldacci R, Qin H, Zhang Z (2023) Enhanced solution representations for vehicle routing problems with split deliveries. *Frontiers of Engineering Management* 10: 483–498.

Zhu X, Wilhelm WE (2012) A three-stage approach for the resource-constrained shortest path as a sub-problem in column generation. *Computers & Operations Research* 39(2): 164–178.

Zou B, Xu X, Gong Y, de Koster R (2018) Evaluating battery charging and swapping strategies in a robotic mobile fulfillment system. *European Journal of Operational Research* 267(2): 733–753.