Accurate Tracking, Collision Detection, and Optimal Scheduling of Airport Ground Support Equipment

Shixiong Wang, *Graduate Student Member, IEEE*, Yuxin Che, Huangjie Zhao, and Andrew Lim

***Abstract*—In order to lower the ramp risk and improve the aircraft ground handling efficiency, we aim to: 1) track ground support equipment (GSE) in a real-time and high-accuracy manner so that we can not only conveniently obtain the positions and velocities of them but also reliably report latent collisions among aircraft and GSE. As a result, corresponding ramp risks could be detected and handled in advance and 2) schedule the GSE in an optimal manner based on the real-time data gathered in advance to make efficient use of GSE so that we can smoothly serve the annually increasing air traffic while controlling the ramp area congestion and GSE overheads. In detail, first, we develop a real-time and high-accuracy tracking device consisting of one real-time kinematic (RTK) unit and heading unit(s), for GSE including not only those which have only one carriage, such as tractors, shutters, and so forth but also baggage transit trains that contain one tug plus multiple dollies. The tracking accuracy for GSE could be limited within centimeters so that the monitor, avoidance, and fixation of unaware ramp risks become possible. Second, for optimal scheduling of GSE, a mixed-integer linear programming model and an efficient heuristic algorithm are proposed to minimize the total cost of equipment's rental and travel consumption while respecting the constraints, such as flights timetables, GSE moving speeds limit, the total number of GSE available in stock, the maximum number of dollies allowed to attach to each baggage transit train, and so on.**

***Index Terms*—Accurate tracking, aircraft ground handling, collision detection, ground support equipment (GSE), optimal scheduling, ramp safety.**

I. Introduction

A. Subject Matter

THE 21st century has seen the robust and exponential growth of worldwide air traffic [1] (see Fig. 1 therein). As an example, refer to the business report of the Hartsfield–Jackson Atlanta International Airport in the United States [2]. Since thoroughly changing and/or substantially expanding the current infrastructures of airports distributed around the world seems costly or even impossible [3], requiring the seamless [4]–[7], timely [3], safe [8], and efficient [3]–[8] ground services (also known as aircraft ground handling, ramp/apron/airside service, and ramp operation [8]) during the turnaround time periods of aircraft become preferable. Considering that the natures of seamlessness and timeliness can be actually generalized by efficiency, we, in this article, only study the safety and efficiency problems of ground service. Attached to the scope of this article, we exclude the studies of the nonground-service problems, such as flight scheduling [9], runway/taxiway scheduling [10], [11], gate assignment [12], security check, passenger services, and the like. In order not to confuse readers from different communities, we refer the reader to [8, pp. 36–39] for a glossary of terms regarding ground service, and [5] and [7] for the entire procedure of aircraft ground handling. For more information on airport ground service and the subject matter of this article, see the video in the supplementary material or our project website: https://alim.algorithmexchange.com/caas/.

B. Literature Review

Ground service for an arrival or departure aircraft, which involves various types of ground support equipment (GSE, such as shutters, tractors, tugs, dollies, and so forth) and numbers of ground crews within a crowded ramp, is usually complex, concurrent, and time-pressed [5], [8]. Due to weather conditions and human factors (fatigue, situational unawareness, insufficiency in crew training, fear of fine/punishment so that they did not report the GSE collisions/misoperations, etc.), the ramp safety (namely, reliability of ground service) is always an issue arousing the problems of flight risk, loss of reputations of airlines, and increase of cost of maintenance and management [8]. The situation gets worse because the current ground service system lacks sufficient data collection mechanisms of accidents and incidents, which makes the automated risk report and *post hoc* analysis impossible. Among all of the ramp risks imposed on aircraft, the GSE misoperations and the collisions among aircraft and ground support equipment are prominent [8]. Considering that the GSE misoperations are equipment related, they are out of scope to investigate case by case. In this article, we only study the automatic collision detection, which is unaware of and/or unreported by the ground crews. It is obvious that reliable collision detection requires accurate position determination of GSE. Unfortunately, existing tracking technologies for ground support equipment only include the traditional GPS [13] and the RFID method [14]–[17], providing tracking accuracy no better than meters which denies the utilization for collision

Manuscript received March 15, 2020; revised April 30, 2020; accepted June 22, 2020. Date of publication June 25, 2020; date of current version December 21, 2020. This work was supported by the National Research Foundation of Singapore under Grant NRF-RSS2016-004. *(Shixiong Wang, Yuxin Che, and Huangjie Zhao contributed equally to this work.) (Corresponding author: Shixiong Wang.)*

The authors are with the Department of Industrial Systems Engineering and Management, National University of Singapore, Singapore (e-mail: s.wang@u.nus.edu; yuxinche@u.nus.edu; huangjie@u.nus.edu; isealim@nus.edu.sg).

This article has supplementary downloadable material available at https://ieeexplore.ieee.org, provided by the authors.

Digital Object Identifier 10.1109/JIOT.2020.3004874

2327-4662 © 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: Imperial College London. Downloaded on September 19,2023 at 15:00:29 UTC from IEEE Xplore. Restrictions apply.

detection. Thus, a real-time and high-accuracy GSE tracking device supporting collision detection is urgent to develop. As an example, see an open tender issued by the Civil Aviation Authority of Singapore in 2018 [18].

Concerning the efficiency issue of the ground services, the research is also scarce, mainly due to insufficient emphasis from the traditional air transportation system [5]. It is the continuous growth of air traffic and the willing of improving ground service performance (to avoid the delay of a departure flight resulted from ineffective ground service and the like) while controlling the costs that arouse the research in this field since about 2012. Up to date, the ground support equipment and manpower at nearly all of the airports around the world are still manually scheduled, resulting in inefficient utilization of sources, waste of assets, congestion at the ramp, incapability of coping with emergencies, and uncontrollable delays of flights [4], [7], [18]–[21]. Although studies about optimal scheduling of deicing vehicles [22], ferry vehicles (also known as buses/shutters) [19], [21], baggage transport vehicles [20], and tractors (also known as a trailer, note that the vehicle towing the baggage transit train is called Tug) [23], [24] have been reported, it is easy to predict that we still have a long way to go until seeing the mature and ubiquitous implementation of the integrated GSE scheduling system. In addition, in the mentioned literature, although the scheduling of single-carriage GSE vehicles as tractors and shutters have been studied, the research on scheduling for multicarriage baggage transit trains is scarce. The latter problem is more challenging than the former because:

1) the number of dollies attached to the tug is changeable so that the loading capacity of each train is not fixed;
2) each train can serve more than one flight and each flight can be served by more than one train, depending on the real amount of baggage/cargoes contained in each flight;
3) each aircraft can convey more than one type of cargoes so that we need more than one type of dollies (such as dolly for loose baggage and dolly for unit load device/cargo pallet) for each flight;
4) the time window (timetable) of each flight parking at a gate (or a remote stand) can vary in the real world due to emergencies, early arrival, and/or delays so that the assumption of invariance is not practical.

All the listed four concerns challenge the study in [20]. Due to the instantiated reasons provided above, the cargo/baggage transit train scheduling problem is admitted to be hard to handle so that it is left as an open problem in [25].

Among the state-of-the-art studies, worthy of mentioning are another two papers presenting the holistic GSE scheduling. Instead of only paying attention to the single type of GSE, they try to take into account the integrated scheduling of all the GSE necessary for ground service [3], [25], resulting in the possible shortening of turnaround time. However, as pointed out by the authors, these two papers are at a tactical level ignoring the specific route planning of each type of GSE vehicle. That means, through such holistic optimizing, only the turnaround time of each flight could be controlled, while the issue of whether the GSE resources are optimally managed or not in an efficient and cost-saving manner is not considered.

In summary, the problems pertaining to not only accurate tracking and collision detection but also optimal scheduling of GSE still remain open.

C. Our Contributions

In this article, in order to bridge the research gap in this field, we aim to achieve the following.

1) We aim to develop, from hardware to software level, a real time, high-accuracy positioning, and heading device, which is able to determine the GSE positions within centimeter level so that it facilitates the find and management of GSE. This also makes reliable collision detection possible.
2) We aim to develop an optimal scheduling solution to GSE, especially the one to baggage transit train, which takes into consideration the three dominant aspects (the first two and the last one) among the new four practical challenges aforementioned. The detailed objectives, constraints, and related practical concerns of the problem will be explained in Section III.

D. Paper Structure

The methods of accurate tracking and collision detection, and optimal scheduling of GSE are presented in Sections II and III, respectively. In Section IV, we show the simulation performance of the proposed scheduling scheme. Finally, we conclude the work of this article in Section V.

II. Accurate Tracking and Collision Detection

At an airport, especially ramp areas, there exist hundreds or thousands of dollies and tugs serving the baggage transition. Over the years, they were manually placed, found, and scheduled by ground crews on apron areas. It caused uncontrolled service delays, unnecessary overheads, misplacement, congestion, etc. The issues have been lightened by the use of traditional GPS and RFID solutions [13], [14]. In this section, we develop a tracking device with centimeter-level accuracy that allows collision detection. Such a high-accuracy positioning solution also facilitates the autonomous GSE in the future, as envisaged in [5].

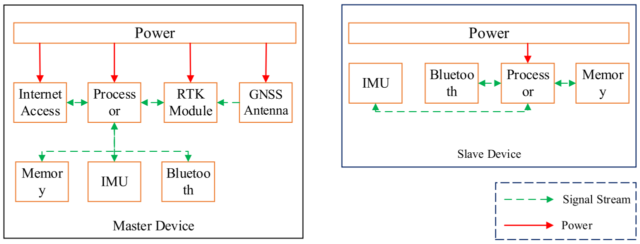
A. Hardware Design

The sensors that we use are real-time kinematic (RTK) and integrated inertial measurement unit (IMU). RTK uses carrier-phase enhancement to improve the precision of positions obtained from global navigation satellite systems (GNSSs) (BEIDOU, GPS, etc.) [26]. It can provide centimeter-level positioning accuracy. On the other hand, the advanced technologies of information fusion in instruments and measurements make possible the high accuracy and precision of integrated IMU for heading [27], [28]. Since the movement speed of vehicles at an airport is slow (maximum 25 km/h [5]) and the periodic calibration mechanism of IMU is designed and performed in this article, commercial IMU with satisfactory performances is reliable for heading the GSE vehicles. The recommended commercial models of RTK board and IMU module are listed in Table I.

Authorized licensed use limited to: Imperial College London. Downloaded on September 19,2023 at 15:00:29 UTC from IEEE Xplore. Restrictions apply.

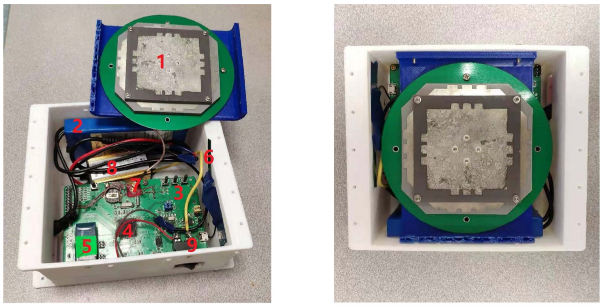
TABLE I
RECOMMENDED COMMERCIAL MODELS OF RTK AND IMU

Module	Suggested Model	Produced by	Performances (real field tested)		Business Website
			Accuracy	Precision	
RTK Board	P327	Hemisphere	0	3 cm^2 (variance)	www.hemispheregnss.com
IMU	WT101	Wit-motion	0 (calibrated)	0.1 deg^2 (variance)	www.wit-motion.com/english.php



(a) (b)

Fig. 1. (a) Schematic of the (master) device to track a single-carriage GSE vehicle. (b) Schematic of the (slave) device for tracking dollies.



(a) (b)

Fig. 2. Final product (prototype) of the designed Master tracking device. In (a), 1: GNSS Antenna, 2: Battery, 3: Motherboard, 4: Microprocessor (STM32F407), 5: Bluetooth, 6: GSM Antenna, 7: Port to connect the GNSS antenna with RTK board, 8: Voltage Convertor, and 9: Switch. The RTK board and the IMU module are at the other (bottom) side of the motherboard. (a) Inside view. (b) Top view.

It is straightforward to design the device for single-carriage GSE vehicles, such as tractors, buses, container loader, and so forth. The schematic of such a solution is demonstrated in Fig. 1(a). We refer to the device for tracking a single-carriage GSE vehicle as the master device because it is also used for tracking the tug of a multicarriage baggage transit train.

The Internet access module is used for two reasons: 1) sending the data to the remote server (namely, the integrated GSE scheduling system) and 2) obtaining the correction signals for RTK board from the remote RTK reference base(s). Bluetooth is used for communications among the master and the slave devices on a train. Memory is used to record the unique asset identification number and other necessary information.

Compared to the master device, a slave device only includes an IMU module so that the slave is significantly cheaper than the master because an RTK board and a GNSS antenna are notably more expensive than an IMU module.

The prototype of the final product is demonstrated in Fig. 2 (the top lid is removed). Visually, the place where to deploy the master device is illustrated in Fig. 3(a).

B. Coordinate Transformation

The master device returns the longitude and latitude ($\lambda - \phi$) of the position where it is placed. Using the transformation methods introduced in [29], we can obtain the Cartesian information ($x - y - z$) in the local geodetic coordinate (east-north-up) and/or in the universal transverse Mercator coordinate.

Remark 1: Note that the height data of GSE vehicles are fixed. It is enough to only investigate the information in x and y axes.

C. Tracking Algorithm

The tracking algorithm for single-carriage GSE vehicle is a special case of that for a baggage transit train. In the following, we only study the tracking problem for a train. A baggage transit train is shown in Fig. 3(a) and (b).

For the demonstration purpose and without loss of generality, in the figure, we have assumed the train consists of one tug and two dollies, although in practice the number of dollies attached to this train could range from zero to many. Besides, suppose all the master and slave tracking devices are placed along the symmetry axes of vehicles. The key geometries of the train are shown in Fig. 3(c).

As shown in Fig. 3(d), suppose we have a global geodetic coordinate $\boldsymbol{x}_A - \boldsymbol{P}_A - \boldsymbol{y}_A$, where $\boldsymbol{P}_A$ is the prechosen origin on the airport ground surface; and we have the local geodetic coordinates $\boldsymbol{x}_0 - \boldsymbol{P}_0 - \boldsymbol{y}_0$, $\boldsymbol{x}_1 - \boldsymbol{P}_1 - \boldsymbol{y}_1$, and $\boldsymbol{x}_2 - \boldsymbol{P}_2 - \boldsymbol{y}_2$, which are fixed on the vehicles. For simplicity, let $\boldsymbol{P}_i$, $i = 0, 1, 2$, represent the positions of the corresponding vehicles.

In practice, let I denote the real number of dollies attached to the tug (i.e., $I = 2$ in Fig. 3), and $\boldsymbol{P}_1, \boldsymbol{P}_2, \ldots, \boldsymbol{P}_I$ denote their positions in the global geodetic coordinate $\boldsymbol{x}_A - \boldsymbol{P}_A - \boldsymbol{y}_A$. Using the master device, we can directly obtain the real-time position $\boldsymbol{P}_0$ of the tug and its heading θ_0. Likewise, by using the slave devices, we can directly obtain the real-time headings $\theta_1, \theta_2, \ldots, \theta_I$ of the dollies. By the geometric relations and parameters defined in Fig. 3, we can see that the point $\boldsymbol{P}_i$ in the local system $\boldsymbol{x}_{i-1} - \boldsymbol{P}_{i-1} - \boldsymbol{y}_{i-1}$ is

$$^{i-1}\boldsymbol{P}_i = \boldsymbol{R}_{i-1} + \boldsymbol{\Theta}^{\mathrm{T}}\left(\Delta\theta_{i-1}^{i}\right) \cdot \boldsymbol{F}_i \tag{1}$$

where $\Delta\theta_{i-1}^{i} := \theta_i - \theta_{i-1}$; T here denotes the transpose of a matrix; and

$$\boldsymbol{\Theta}(\theta) = \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{bmatrix}$$

$$\boldsymbol{R}_{i-1} = \begin{pmatrix} 0 \\ -R_{i-1} \end{pmatrix} \quad \boldsymbol{F}_i = \begin{pmatrix} 0 \\ -F_i \end{pmatrix}.$$

Besides, in the global system $\boldsymbol{x}_A - \boldsymbol{P}_A - \boldsymbol{y}_A$, this point is

$$\boldsymbol{P}_i = \boldsymbol{P}_{i-1} + \boldsymbol{\Theta}^{\mathrm{T}}(\theta_{i-1}) \cdot {}^{i-1}\boldsymbol{P}_i. \tag{2}$$

Authorized licensed use limited to: Imperial College London. Downloaded on September 19,2023 at 15:00:29 UTC from IEEE Xplore. Restrictions apply.

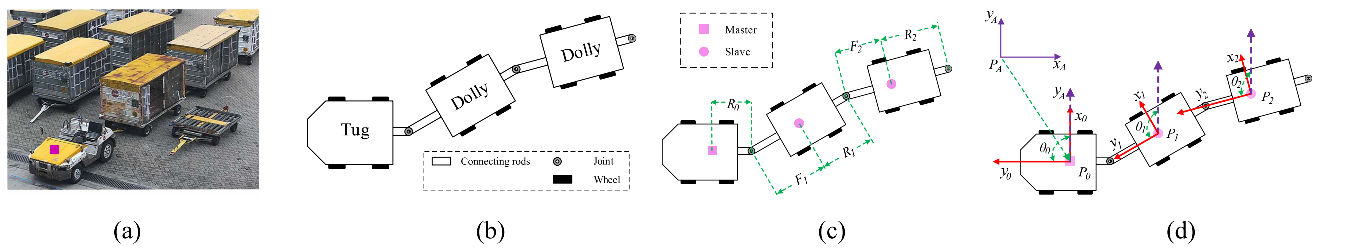


Fig. 3. Baggage transit train containing a tug and two dollies and its geometries. In (a), the pink-filled square shows the position where to deploy the master device (see also our attached demonstration video online or at: https://alim.algorithmexchange.com/caas/data/video3.mp4). The filled pink circles and squares in (c) and (d) denote the places where we should put the slave and the master devices, respectively. (a) Real train (source credit: [30]). (b) Illustrative train. (c) Key dimensions. (d) Global and local coordinate systems.

Since $\boldsymbol{P}_i = \boldsymbol{P}_0 + {}^0\boldsymbol{P}_i$, we have the positions of the dollies as

$$\begin{aligned}\boldsymbol{P}_i &= \boldsymbol{P}_{i-1} + \boldsymbol{\Theta}^{\mathrm{T}}(\theta_{i-1}) \cdot \left[\boldsymbol{R}_{i-1} + \boldsymbol{\Theta}^{\mathrm{T}}\left(\Delta\theta_{i-1}^{i}\right) \cdot \boldsymbol{F}_i\right] \\ &= \boldsymbol{P}_0 + \sum_{k=0}^{i-1}\left\{\boldsymbol{\Theta}^{\mathrm{T}}(\theta_k) \cdot \left[\boldsymbol{R}_k + \boldsymbol{\Theta}^{\mathrm{T}}\left(\Delta\theta_k^{k+1}\right) \cdot \boldsymbol{F}_{k+1}\right]\right\} \quad (3)\end{aligned}$$

where ${}^0\boldsymbol{P}_i$ denote the coordinates of the point $\boldsymbol{P}_i$ in the local system $\boldsymbol{x}_0 - \boldsymbol{P}_0 - \boldsymbol{y}_0$ and

$$\boldsymbol{R}_k = \begin{pmatrix} 0 \\ -R_k \end{pmatrix} \quad \boldsymbol{F}_{k+1} = \begin{pmatrix} 0 \\ -F_{k+1} \end{pmatrix}.$$

In order to efficiently schedule the trains and forecast latent upcoming collisions, we are interested in obtaining the moving velocities of GSE. This requires us to estimate the velocity from the collected position data. Since the time-difference method (i.e., $[x(n) - x(n-1)]/T$, T is the time slot) to estimate the velocity from the noisy position measurements $x(n)$ will amplify the noise, we should use tracking algorithms. Plus, by using the tracking algorithm, the real-time positioning accuracy could be further improved, compared to the direct measurements from the master and slave devices.

Suppose we are interested in tracking the tug of a train, we have the Markov linear tracking system [31], [32] as

$$\begin{cases} \boldsymbol{X}(n+1) = \boldsymbol{\Phi}\boldsymbol{X}(n) + \boldsymbol{G}\boldsymbol{W}(n) \\ \boldsymbol{Y}(n) = \boldsymbol{H}\boldsymbol{X}(n) + \boldsymbol{V}(n) \end{cases} \quad (4)$$

where n denotes the discrete time index; $\boldsymbol{W}$ and $\boldsymbol{V}$ are the process noise vector and measurement noise vector with proper dimensions, respectively; and $\boldsymbol{Y}(n) := \boldsymbol{P}_0(n)$ denotes the measurement vector. If we use the constant acceleration model [33] to approximate the running dynamics of the tug (in general, a GSE vehicle), the state vector $\boldsymbol{X}$, the system matrix $\boldsymbol{\Phi}$, and the measurement matrix $\boldsymbol{H}$ are defined as

$$\boldsymbol{X} = [P_{0,x}, V_{0,x}, A_{0,x}, P_{0,y}, V_{0,y}, A_{0,y}]^{\mathrm{T}}$$

$$\boldsymbol{\Phi} = \begin{bmatrix} \boldsymbol{I}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{I}_{3\times3} \end{bmatrix} \otimes \begin{bmatrix} 1 & T & T^2/2 \\ 0 & 1 & T \\ 0 & 0 & 1 \end{bmatrix}$$

$$\boldsymbol{G} = \begin{bmatrix} \boldsymbol{I}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{I}_{3\times3} \end{bmatrix} \otimes \begin{bmatrix} T^2/2 \\ T \\ 1 \end{bmatrix}$$

$$\boldsymbol{H} = \begin{bmatrix} 1\,0\,0\,0\,0\,0 \\ 0\,0\,0\,1\,0\,0 \end{bmatrix}$$

in which $P_{0,x}$ and $P_{0,y}$ are true positions of tug in x and y axes, respectively; $V_{0,x}$, $V_{0,y}$, $A_{0,x}$, and $A_{0,y}$ are true velocities and true accelerations; $\boldsymbol{I}_{3\times3}$ and $\boldsymbol{0}_{3\times3}$ are identity matrix and zero matrix with a dimension of 3; T here is the time slot between n and $n+1$; and $\otimes$ stands for the Kronecker product. Suppose that we have two matrices $\boldsymbol{A}$ and $\boldsymbol{B}$. The Kronecker product defines a new block matrix $\boldsymbol{C}$ as

$$\boldsymbol{C} = \boldsymbol{A} \otimes \boldsymbol{B} = \begin{bmatrix} \boldsymbol{A}_{11} & \boldsymbol{A}_{12} \\ \boldsymbol{A}_{21} & \boldsymbol{A}_{22} \end{bmatrix} \otimes \boldsymbol{B} = \begin{bmatrix} \boldsymbol{A}_{11}\boldsymbol{B} & \boldsymbol{A}_{12}\boldsymbol{B} \\ \boldsymbol{A}_{21}\boldsymbol{B} & \boldsymbol{A}_{22}\boldsymbol{B} \end{bmatrix}$$

where we assume that $\boldsymbol{A}$ is a block matrix constructed by four smaller matrices $\boldsymbol{A}_{11}$, $\boldsymbol{A}_{12}$, $\boldsymbol{A}_{21}$, and $\boldsymbol{A}_{22}$.

After executing the canonical Kalman filter [32, Ch. 5.1], we can obtain the accuracy-improved positions and estimated velocities, respectively, in x and y axes, of the tug.

The tracking algorithm for dollies keeps the same since measurements of positions of dollies could be obtained by (3). However, in order to support the optimal scheduling of GSE, it is enough to only track a tug, through which the moving velocities are estimated (the estimates of $V_{0,x}$ and $V_{0,y}$ are $\hat{V}_{0,x}$ and $\hat{V}_{0,y}$, respectively).

D. Periodic Calibration of Heading Module

Heading module (IMU) may suffer from the drift problem in a long run. When the estimates of accelerations ($\hat{A}_{0,x}$ and $\hat{A}_{0,y}$) are approximately zero, meaning the train is now running in a straight line with constant velocity, it is safe to reset the IMU value (current heading) as

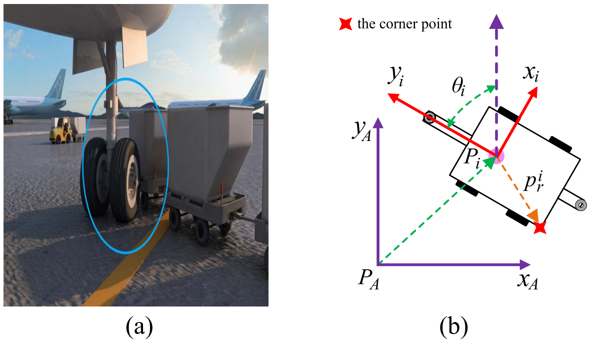
$$\theta_0 := \arctan\left(\hat{V}_{0,x}/\hat{V}_{0,y}\right). \quad (5)$$

Note that this calibration equation can only be performed for the tug of a train, excluding the dollies. However, when a train runs with the constant velocity for a relatively long time, the train should be straight, meaning $\theta_0 \equiv \theta_i$, $i = 1, 2, \ldots, I$. Thus, the values of heading modules in the slave devices could also be reset to the value used for calibrating the master device.

E. Collision Detection

Equation (3) gives all the positions of both tugs and dollies. However, for collision detection, we are concerned with the corner points on a carriage (tug or dollies), especially the corner points on the last dolly of a train (see Fig. 4). For the ith dolly, we want to investigate the real-time position of the

Authorized licensed use limited to: Imperial College London. Downloaded on September 19,2023 at 15:00:29 UTC from IEEE Xplore. Restrictions apply.



(a) (b)

Fig. 4. (a) Collision illustration between aircraft and a dolly. (b) Relative geometry of a corner point on a dolly.

red-filled tetragon corner point. Suppose its relative position in the local coordinate $\boldsymbol{x}_i - \boldsymbol{P}_i - \boldsymbol{y}_i$ is $\boldsymbol{p}_r^i$. We can obtain its absolute position in the global coordinate $\boldsymbol{x}_A - \boldsymbol{P}_A - \boldsymbol{y}_A$ as

$$\boldsymbol{P}_r^i = \hat{\boldsymbol{P}}_i + \boldsymbol{\Theta}^{\mathrm{T}}(\theta_i) \cdot \boldsymbol{p}_r^i \tag{6}$$

where $\hat{\boldsymbol{P}}_i$ is the accuracy-improved position of the ith dolly obtained from the tracking algorithm with the raw measurement $\boldsymbol{P}_i$; typically, $i = I$ (the tail dolly of a train) for the collision detection problem.

The detailed three-dimensional collision detection methods are mature in computer graphics as long as the required data are available [34]. We ignore the specific implementation here.

III. Optimal Scheduling

In this section, we study the optimal scheduling problem for GSE. Since the ground service is performed in parallel at different gates and/or remote stands [5], [25], we only consider the scheduling problem in individuals for each type of GSE vehicle instead of in holistic. Besides, as the scheduling problem for baggage transit trains is the general case of the scheduling problems for other GSE, we only formulate and solve the scheduling problem for baggage transit trains.

A. General Assumptions, Objectives, and Constraints

This section gives some general concerns on assumptions, objectives, and constraints, which will be taken into account in the following modeling section. Assumptions listed below are the premises of modeling.

1) We have full information access to the real-time flight schedules (timetables) which are updated along with the weather, emergencies, flights early arrivals and delays, and so forth, at an airport that we are working on.
2) All the airlines and ground service providers operating at the airport collaborate with each other so that the global and unified scheduling for every flight and GSE are possible.

The objective of our scheduling model is to minimize the total cost, including: 1) the cost of renting tugs and dollies and 2) total fuel (or electricity power) consumption of tugs. In this article, we suppose rental for each tug/dolly during a fixed time period (for example, a day) is a constant, while the tug fuel cost is proportional to the total travel distance. We also assume that fuel consumption per unit kilometer is constant and it will cause no fuel consumption during the waiting period.

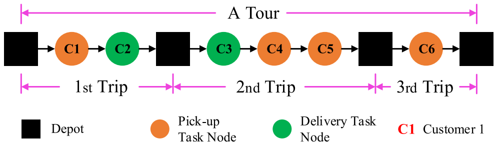


Fig. 5. Tour of a vehicle with three trips. In VRP settings, a *tour* of a vehicle includes all the pickup and delivery events during one duty period, for example, one day. In this tour, this vehicle can start off from and end at the depot many times (multitrip). In this diagram, we can see that in the first trip, the vehicle starts off from the depot to pickup the goods from customer 1 ($C1$). Then, it delivers goods (loaded from the depot) to customer 2 ($C2$). Finally, it returns to the depot (to release the goods picked up from $C1$).

Besides, we have the following constraints.

1) Every train must strictly respect the timetables (time schedules excluding the necessary preceding procedures, such as in-chock and so on) of the aircraft parked not only at gates but also at remote stands.
2) The baggage waiting time is limited (in this article within 60 min immediately starting from flights' in-chock) in order not to outrage the passengers.
3) Each tug cannot exceed the maximum speed of 25 km/h [5].
4) Each train can include no more than six dollies.
5) The time to load/unload each baggage container to/from a dolly is about 3 min (estimated from [5] and [20]).
6) Each train can serve more than one flight and each flight can be served by more than one train.
7) Each flight can be served by at most four trains in order to avoid the ramp congestion.

B. Motivations of Modeling

We observe that this problem can be categorized as a variant of the vehicle routing problem (VRP). The VRP, first proposed in 1959 [35], is one of the classical combinatorial optimization problems that is studied extensively in the past few decades. The basic VRP considers that each vehicle starts and ends at the depot to visit a set of customers to pickup (or deliver) goods required by customers, while each customer is served exactly once by one of the vehicles, and the total demand of customer served by each vehicle is limited by the vehicle's loading capacity. The objective is to minimize the total travel distance. In the recent few decades, different practical extensions, relaxations, and constraints based on real-world applications are applied to tackle different requirements from both customer and commercial side, for example, see [36]–[39]. In addition, various methods are developed to resolve the computational difficulty induced by these problems, which are proved to be NP-hard. In our study, the problem has the following additional properties compared with basic VRP: 1) multitrip [38]: each baggage transit train may visit the baggage management center (namely, the depot in the VRP problem setting) multiple times within one tour, for intuitive understanding (see Fig. 5); 2) time window [40]: each flight has an arriving time (in-chock) and departure time (preplanned by the flight

Authorized licensed use limited to: Imperial College London. Downloaded on September 19,2023 at 15:00:29 UTC from IEEE Xplore. Restrictions apply.

scheduling problem [9]), which makes the service for this flight restricted within this time window; 3) split pickup and delivery [36]: each flight would result in two events: pickup event and delivery event, which represent the events of unloading/loading luggage from/to the aircraft, respectively, and each pickup/delivery event of one aircraft may be served by more than one physical train, or only one physical train but with multiple trips; 4) simultaneous pickup and delivery [37], [41]: pickup and delivery events can both exist in the same trip; 5) replaceable attached dollies: the number of dollies attached to a tug is not fixed. Dollies should be attached/detached in accordance with pickup/delivery tasks subject to the maximum number of attached dollies allowed (set as 6 in this article); 6) limited baggage waiting time [42]: the baggage transit train should deliver the collected baggage (at a gate or a remote stand) to the baggage management center within limited time slot so that passengers will not wait for their baggage for a very long time; and 7) precedence constraint of pickup event: all pickup service should be in precedence to the delivery service for each flight.

Given the information and analysis above, we are inspired to formulate this variant of the VRP problem as a mixed-integer linear programming (MILP) problem.

C. Notations

Let $I := \{1, \ldots, n\}$ define a set of n flights to serve; $i \in I$ denote the flight i to serve during the scheduling process. In addition, we let $k \in K$ denote the available tugs; $h \in H$ denote the gate (or remote stand) where flights park; $r \in R$ denote the rth trip of a tug since each tug is allowed to take multiple trips from the baggage management center to flights within time window. Let $A \geq \{B, C\}$, $A = \{B, C\}$ denote that A is no less than, equals to both B and C, respectively. Let $\{B, C\} \in A$ denote both the elements B and C belong to the set A.

D. Problem Formulation

For ease of description, we jointly refer to gates and remote stands as gates. We are concerned with the baggage service for a set of flights I within a specific time period. The demands of each flight are how many tugs and dollies needed for pickup and delivery events, respectively. Ideally, we expect a joint scheduling problem of tugs and dollies (for serving several flights at gates) that each tug not only pulls the laden dollies for delivery tasks but also pulls enough preplanned vacant dollies for pickup tasks. However, such joint scheduling is a dynamic problem. The dynamical nature is from the fact that when a flight is served by more than one train, the arrival sequence of trains matters. As an illustrative example, suppose that: 1) one flight is accepting service at one of the gates; 2) six vacant dollies for pickup are required and two tugs for baggage service are assigned; and 3) at the GSE parking area of this gate, there already exists one vacant dolly, which means the two tugs need to take other five vacant dollies so that this flight could be timely served. If tug no. 1 is planned to pickup five baggage-laden dollies, but only takes three new vacant dollies itself for this flight, it must wait for tug no. 2 which will provide a new vacant dolly to this gate because there are in total four vacant dollies available for tug no. 1. The issue could be avoided if tug no. 2 comes no later than tug no. 1. Obviously, the joint scheduling problems are indeed train-arrival sequence-related and dynamic, which makes the problem intricate, even impossible to model in the static programming frame. Note that recording the arrival sequence of trains and taking actions accordingly afterward is a dynamic programming setting. To make the scheduling problem a static and easier one, we assume the following.

1) Trains starting off from the depot only pull laden dollies for delivery tasks, pulling *no* vacant dollies for pickup. They use the vacant dollies at the parking area (for information of parking area, see [7]) of each gate for pickup.
2) Trains leave the laden dollies at the gates' parking area after delivery tasks so that those laden dollies will become newly available vacant dollies at the corresponding gates.

Under this new setting, we can have a prior scheduling mechanism that ensures there are always sufficient vacant dollies in the parking area. Following the above example, the new strategy is that we guarantee, by a prior scheduling process, the parking area has five (instead of one, meaning four new vacant dollies are replenished in advance) vacant dollies so that even if the tug no. 1 comes first there is no need to wait for tug no. 2. By taking this strategy, the arrival sequence (namely, the dynamic nature) of tugs will no longer matter.

Therefore, a two-stage tugs and dollies dispatching model is proposed. The first stage is to schedule the tugs and dollies for real-time baggage service, during which the minimum number of vacant dollies required at each gate at the beginning of each decision window (see later) could be calculated. In order to prepare enough vacant dollies for each gate calculated in the first stage optimization, another independent dolly dispatching plan (namely, the prior scheduling mentioned above) is made in the second stage. In summary: 1) the optimal scheduling model of tugs for baggage service is as (7)–(48), referred to as the master ($\mathcal{M}$) problem and 2) the optimal scheduling model of dollies' prior dispatch is as (49)–(67), referred to as the slave ($\mathcal{S}$) problem.

The intuitive explanations to the objectives and constraints of these two models are detailed in Sections III-E and III-F, respectively.

Detailed parameters and decision variables (shorted as variables in the following) settings are defined in Tables II and III, for the master problem and slave problem, respectively. The values of some parameters therein are specified for simulation in this article.

In addition, in order to adapt our scheduling algorithm to the changeable flights' timetables due to early arrival, delay, emergency, and so on, we do not make a very-long-time plan of tugs and dollies scheduling in advance, for example, a 24-h plan at 0 o'clock according to the flights' timetables of this day. Instead, we execute the scheduling algorithm every, for example, 4 h in line with newly updated flights' timetables. This point will be further explained in Section III-G (model solving). This short time period for renewing the

Authorized licensed use limited to: Imperial College London. Downloaded on September 19,2023 at 15:00:29 UTC from IEEE Xplore. Restrictions apply.

TABLE II
PARAMETERS AND VARIABLES FOR THE MASTER PROBLEM

Parameter	Description	Value
τ_{ij}	travel time between parked flight i and j	
c	average time for loading/unloading baggage to/from each dolly	3min
d	maximal number of dollies a tug can carry	6
m	maximal number of trips allowed to serve a flight for pick up or delivery	4
n	maximal number of vacant dollies allowed at the parking area of a gate	12
$[e_i, l_i]$	service time window for flight i	
g	maximum duration for passengers to wait for baggages starting immediately from in-chock	60min
s	average speed for a tug	18km/h
r_1	cost for renting one tug	20
r_2	cost for unit travel distance of a tug	0.05/km
p_i	number of dollies flight i needs for pick-up	
q_i	number of dollies flight i needs for delivery	
M	a large real-valued number	10^5
Variable	**Description**	**Domain**
α_i^{kr}	whether tug k serves flight i in its r_{th} trip	$\{0,\ 1\}$
x_i^{kr}	whether tug k picks up baggage from flight i in its r_{th} trip	$\{0,\ 1\}$
y_i^{kr}	whether tug k delivers baggage to flight i in its r_{th} trip	$\{0,\ 1\}$
Z_{ij}^{kr}	whether tug k serves flight j after serving flight i in its r_{th} trip	$\{0,\ 1\}$
σ^k	whether tug k is used in this scheduling process	$\{0,\ 1\}$
$N_{x,i}^{kr}$	number of dollies of tug k in its r_{th} trip to pick up from flight i	Z^+
$N_{y,i}^{kr}$	number of dollies of tug k in its r_{th} trip to deliver to flight i	Z^+
$P_{x,i}^{kr}$	number of dollies of tug k for pick-up after tug k serves flight i in its r_{th} trip	Z^+
$P_{y,i}^{kr}$	number of dollies of tug k for delivery after tug k serves flight i in its r_{th} trip	Z^+
$T_{x,i}^{kr}$	service start time when tug k serves flight i in its r_{th} trip to pick up baggage	R^+
$T_{y,i}^{kr}$	service start time when tug k serves flight i in its r_{th} trip to deliver baggage	R^+
T_0^{kr}	service start time of tug k's r_{th} trip	R^+
T_{n+1}^{kr}	time when tug k returns to destination depot in its r_{th} trip	R^+
H_h^0	number of vacant dollies needed at gate h before the first flight is served	Z^+

scheduling decision is termed as *decision window* in this article.

Remark 2: Note that in this simplified frame, the congestion due to redundant dollies in the parking area of each gate is not an issue because the replenishment and the use of vacant dollies, which were resulted from delivery and pickup, respectively, are dynamic with the arrival and departure of aircraft at this gate. Besides, within a not-very-long decision window, the number of flights at each gate is limited so that the number of vacant dollies at each gate will not gather up to many. This point will be validated in the simulation study later.

$$(\mathcal{M}) : \ \min \ r_1 \sum_k \sigma_k + r_2 \sum_i \sum_j \sum_k \sum_r Z_{ij}^{kr} \cdot \tau_{ij}. \tag{7}$$

Subject to

$$\beta^{kr} \leq \sigma^k \quad \forall k, r \tag{8}$$

TABLE III
PARAMETERS AND VARIABLES FOR THE SLAVE PROBLEM

Parameter	Description	Value
τ_{hl}	travel time between gate h and l	
c	average service time for each dolly	1min
d	maximal number of dollies a tug can carry	6
T_h	deadline for prior dispatch for gate h	
H_h^0	the number of dollies gate h needs	
M	a large real-valued number	10^5
Variable	**Description**	**Domain**
α_h^{kr}	whether tug k serves gate h in its r_{th} trip	$\{0,\ 1\}$
Z_{hl}^{kr}	whether tug k serves gate l after serving gate h in its r_{th} trip	$\{0,\ 1\}$
σ^k	whether tug k is used in this scheduling process	$\{0,\ 1\}$
N_h^{kr}	number of dollies of tug k in its r_{th} trip to serve gate h	Z^+
T_h^{kr}	service start time when tug k serves gate h in its r_{th} trip	R^+

$$\sigma^k \leq \sum_r \beta^{kr} \quad \forall k \tag{9}$$

$$\alpha_i^{kr} \leq \beta^{kr} \quad \forall i \in I, k, r \tag{10}$$

$$\alpha_i^{kr} \geq \{x_i^{kr},\ y_i^{kr}\} \quad \forall i \in I, k, r \tag{11}$$

$$\alpha_i^{kr} \leq x_i^{kr} + y_i^{kr} \quad \forall i \in I, k, r \tag{12}$$

$$\sum_{j \in I \cup \{0\}} Z_{ji}^{kr} = \alpha_i^{kr} \quad \forall i \in I, k, r \tag{13}$$

$$\sum_{j \in I \cup \{n+1\}} Z_{ij}^{kr} = \alpha_i^{kr} \quad \forall i \in I, k, r \tag{14}$$

$$\sum_{j \in I} Z_{0,j}^{kr} = \beta^{kr} \quad \forall k, r \tag{15}$$

$$\sum_{j \in I} Z_{j,n+1}^{kr} = \beta^{kr} \quad \forall k, r \tag{16}$$

$$\sigma^{k-1} \geq \sigma^k \quad \forall k \in K/\{1\} \tag{17}$$

$$\beta^{k,r-1} \geq \beta^{kr} \quad \forall k, r \in R/\{1\} \tag{18}$$

$$\{P_{x,0}^{kr},\ N_{x,0}^{kr}\} = 0 \quad \forall k, r \tag{19}$$

$$\{P_{y,0}^{kr},\ N_{y,0}^{kr}\} = \sum_{i \in I} N_{y,i}^{kr} \quad \forall k, r \tag{20}$$

$$y_i^{k,r} \leq N_{y,i}^{k,r} \leq q_i y_i^{k,r} \quad \forall i \in I, k, r \tag{21}$$

$$x_i^{k,r} \leq N_{x,i}^{k,r} \leq p_i x_i^{k,r} \quad \forall i \in I, k, r \tag{22}$$

$$\sum_k \sum_r N_{x,i}^{kr} = p_i \quad \forall i \in I \tag{23}$$

$$\sum_k \sum_r N_{y,i}^{kr} = q_i \quad \forall i \in I \tag{24}$$

$$P_{x,i}^{kr} + P_{y,i}^{kr} \leq d \quad \forall i \in I', k, r \tag{25}$$

$$\sum_k \sum_r x_i^{kr} \leq m \quad \forall i \in I \tag{26}$$

$$\sum_k \sum_r y_i^{kr} \leq m \quad \forall i \in I \tag{27}$$

$$\{N_{x,i}^{kr},\ N_{y,i}^{kr},\ P_{x,i}^{kr}\} \geq 0 \quad \forall i \in I, k, r \tag{28}$$

$$\{P_{y,i}^{kr},\ T_{x,i}^{kr},\ T_{y,i}^{kr}\} \geq 0 \quad \forall i \in I, k, r \tag{29}$$

$$\{T_0^{kr},\ T_{n+1}^{kr}\} \geq 0 \quad \forall k, r \tag{30}$$

Authorized licensed use limited to: Imperial College London. Downloaded on September 19,2023 at 15:00:29 UTC from IEEE Xplore. Restrictions apply.

$$\{\alpha_i^{kr},\ x_i^{kr}, y_i^{kr}\} \in \{0,1\} \quad \forall i \in I, k, r \tag{31}$$

$$\{\beta^{k,r}, \sigma^k\} \in \{0,1\} \quad \forall k, r. \tag{32}$$

See (33)–(48), shown at the bottom of the page.

E. Explanation to the Master Problem

We add $i = 0$ and $i = |I| + 1 = n + 1$ as the dummy gates (or remote stands) used to represent the starting and ending depot (physically the same depot). We denote $I' := I \cup \{0\}$, $I'' := I \cup \{n+1\}$. Objective (7) is to minimize the total cost including the cost of trains (tugs) involved in ramp operation and travel cost. Constraints (8) and (9) mean the tug k can serve flights only if it is chosen to use during the scheduling process. Constraint (10) means flight i can be served by tug k in its rth trip only when tug k starts its rth trip. Constraints (11) and (12) mean the tug k can be used for pickup or delivery task serving the flight i in its rth trip. Constraints (13) and (14) mean if flight i is served by tug k in its rth trip, there must be two edges connected to flight i. Constraints (15) and (16) are the degree constraint for depots. Constraint (17) restricts the sequential use of tug $(k-1)$th and tug kth. Constraint (18) means the prerequisite to start its rth trip for a tug k is to work in its $(r-1)$th trip. Constraints (19) and (20) give the dollies' settings when a tug departs from the baggage management center. Constraints (21) and (22) ensure that the tug must conduct pickup or delivery tasks once it travels to a flight. Constraints (23) and (24) ensure that the pickup and delivery demand for each flight are satisfied. Constraint (25) limits the maximal number of dollies a tug can carry. Constraints (26) and (27) limit the service times of each flight for pickup and delivery service, respectively. Constraints (28)–(32) are variable domain constraints. Constraints (33) and (34) are time window limits for all pickup and delivery tasks serving flight i. Constraint (35) ensures that all the pickup service is prior to the delivery service for each flight. Constraints (36)–(39) ensure the time transition between two consecutive flights in the same trip of the same tug. Constraints (40)–(43) ensure the time constraint between flight and depot in the same trip of the same tug. Constraint (44) ensures the time transition between the last service in $(r-1)$th trip and the start of the rth trip of the same tug. Constraint (45) ensures the baggage waiting time for each flight. Constraints (46) and (47) restrict the number of dollies tug k carries after serving a flight. Constraint (48) is the domain constraints for variable Z_{ij}^{kr}.

$$(\mathcal{S}):\ \min\ r_1 \sum_k \sigma_k + r_2 \sum_h \sum_l \sum_k \sum_r Z_{hl}^{kr} \cdot \tau_{hl}. \tag{49}$$

Subject to

$$\beta^{k,r} \le \sigma^k \quad \forall k, r \tag{50}$$

$$\alpha_h^{kr} \le \beta^{kr} \quad \forall k, r, h \in H \tag{51}$$

$$\sum_{l \in H \cup \{|H|+1\}} Z_{hl}^{kr} = \alpha_h^{kr} \quad \forall h \in H, k, r \tag{52}$$

$$\sum_{l \in H \cup \{0\}} Z_{lh}^{kr} = \alpha_h^{kr} \quad \forall h \in H, k, r \tag{53}$$

$$\sum_{l \in H} Z_{0l}^{kr} = \beta^{k,r} \quad \forall h \in H, k, r \tag{54}$$

$$\sum_{l \in H} Z_{l,|H|+1}^{kr} = \beta^{k,r} \quad \forall h \in H, k, r \tag{55}$$

$$\beta^{k,r-1} \ge \beta^{kr} \quad \forall k, r \in R/\{1\} \tag{56}$$

$$\{\alpha_{|H|+1}^{kr}, \alpha_0^{kr}\} \ge \alpha_h^{kr} \quad \forall h \in H, k, r \tag{57}$$

$$x_i^{kr} \cdot e_i - 2M(1 - x_i^{kr}) \le T_{x,i}^{kr} \le x_i^{kr} \cdot l_i + 2M(1 - x_i^{kr}) \quad \forall i \in I, k, r \tag{33}$$

$$y_i^{kr} \cdot e_i - 2M(1 - y_i^{kr}) \le T_{y,i}^{kr} \le y_i^{kr} \cdot l_i + 2M(1 - y_i^{kr}) \quad \forall i \in I, k, r \tag{34}$$

$$T_{x,i}^{k_1 r_1} + c \cdot N_{x,i}^{k_1 r_1} \le T_{y,i}^{k_2 r_2} \quad \forall i \in I, k_1, k_2, r_1, r_2 \tag{35}$$

$$T_{x,j}^{kr} \ge T_{x,i}^{kr} + c \cdot N_{x,i}^{kr} + \tau_{ij} - 2M(2 - x_i^{kr} - Z_{ij}^{kr}) \quad \forall i \ne j \in I \ \ \forall k, r \tag{36}$$

$$T_{x,j}^{kr} \ge T_{y,i}^{kr} + c \cdot N_{y,i}^{kr} + \tau_{ij} - 2M(2 - y_i^{kr} - Z_{ij}^{kr}) \quad \forall i \ne j \in I \ \ \forall k, r \tag{37}$$

$$T_{y,j}^{kr} \ge T_{x,i}^{kr} + c \cdot N_{x,i}^{kr} + \tau_{ij} - 2M(2 - x_i^{kr} - Z_{ij}^{kr}) \quad \forall i \ne j \in I \ \ \forall k, r \tag{38}$$

$$T_{y,j}^{kr} \ge T_{y,i}^{kr} + c \cdot N_{y,i}^{kr} + \tau_{ij} - 2M(2 - y_i^{kr} - Z_{ij}^{kr}) \quad \forall i \ne j \in I \ \ \forall k, r \tag{39}$$

$$T_{x,i}^{kr} \ge T_0^{kr} + c \cdot N_{y,0}^{kr} + \tau_{0,i} - 2M(2 - x_i^{kr} - Z_{0,i}^{kr}) \quad \forall i \in I \ \ \forall k, r \tag{40}$$

$$T_{y,i}^{kr} \ge T_0^{kr} + c \cdot N_{y,0}^{kr} + \tau_{0,i} - 2M(2 - y_i^{kr} - Z_{0,i}^{kr}) \quad \forall i \in I \ \ \forall k, r \tag{41}$$

$$T_{n+1}^{kr} \ge T_{x,i}^{kr} + c \cdot N_{x,i}^{kr} + \tau_{i,n+1} - 2M(2 - x_i^{kr} - Z_{i,n+1}^{kr}) \quad \forall i \in I \ \ \forall k, r \tag{42}$$

$$T_{n+1}^{kr} \ge T_{y,i}^{kr} + c \cdot N_{y,i}^{kr} + \tau_{i,n+1} - 2M(2 - y_i^{kr} - Z_{i,n+1}^{kr}) \quad \forall i \in I \ \ \forall k, r \tag{43}$$

$$T_{n+1}^{k,r-1} + \sum_{i \in I} c \cdot N_{x,i}^{k,r-1} \le T_0^{k,r} + 2M(1 - \beta^{k,r}) \quad \forall k, r \in R/\{1\} \tag{44}$$

$$T_{n+1}^{kr} - e_i \le g + 2M(1 - x_i^{k,r}) \quad \forall i \in I, k, r \tag{45}$$

$$P_{x,j}^{kr} + N_{x,i}^{kr} + M(1 - Z_{ji}^{kr}) \ge P_{x,i}^{kr} \ge P_{x,j}^{kr} + N_{x,i}^{kr} - M(1 - Z_{ji}^{kr}) \quad \forall i \in I, j \in I', i \ne j, k, r \tag{46}$$

$$P_{y,j}^{kr} - N_{y,i}^{kr} + M(1 - Z_{ji}^{kr}) \ge P_{y,i}^{kr} \ge P_{y,j}^{kr} - N_{y,i}^{kr} - M(1 - Z_{ji}^{kr}) \quad \forall i \in I, j \in I', i \ne j, k, r \tag{47}$$

$$Z_{ij}^{kr} \in \{0,1\} \quad \forall i \in I', j \in I'', i \ne j, k, r \tag{48}$$

Authorized licensed use limited to: Imperial College London. Downloaded on September 19,2023 at 15:00:29 UTC from IEEE Xplore. Restrictions apply.

$$\sum_k \sum_r N_h^{kr} \geq H_h^0 \quad \forall h \in H/\{0\} \tag{58}$$

$$\sum_h N_h^{kr} \leq d \quad \forall k, r \tag{59}$$

$$N_h^{kr} \geq 0 \quad \forall h \in H, k, r \tag{60}$$

$$\{\alpha_h^{kr}, \beta^{k,r}, \sigma^k\} \in \{0, 1\} \quad \forall h \in H, k, r \tag{61}$$

$$Z_{hl}^{kr} \in \{0, 1\} \quad \forall h \in H', l \in H'', k, r \tag{62}$$

$$0 \leq T_{|H|+1}^{kr} \leq T_h \quad \forall k, r. \tag{63}$$

See (64)–(67), shown at the bottom of the page.

F. Explanation to the Slave Problem

We add $h = 0$ and $h = |H| + 1$ to represent the starting and ending depot (physically the same depot). Note that H_h^0, which is a decision variable in the master problem, acts as a known input parameter in the slave problem. Objective (49) is to minimize the total cost including the cost of trains (tugs) involved in ramp operation and travel cost. Constraint (50) means the tug k can serve flights only if it is chosen to use during the scheduling process. Constraint (51) means the tug k can serve flights in its rth trip only if it starts its rth trip. Constraints (52) and (53) mean if gate h is served by tug k in its rth trip, there must be two edges connected to gate h. Constraints (54) and (55) are the degree constraint for depots. Constraint (56) means the prerequisite to start its rth trip for a tug k is to work in its $(r-1)$th trip. Constraint (57) ensures that every tug must start and end at the baggage management center on each trip. Constraint (58) ensures that the demand for each gate is satisfied. Constraint (59) limits the maximal number of dollies a tug can carry. Constraints (60)–(63) are variable domain constraints. Constraints (64)–(66), ensure the time transition between two consecutive gates in the same trip of the same tug. Constraint (67) restricts the time transition between the last service on $(r-1)$th trip and the start of the rth trip of the same tug.

G. Model Solving

1) Rolling Decision Window: As mentioned in Section III-D, the rolling decision window mechanism is applied to adaptively respect the changeable flights' timetables. Suppose the length of our decision window is 6 h, we will, at each day and before 0 o'clock, apply the master model to plan the GSE running schedules for tugs and dollies for the first 6 h (00:00 A.M–06:00 A.M.). After that, we will know how many vacant dollies are needed at each gate so that we can invoke the slave model to schedule the other tugs to replenish vacant dollies for each gate. This replenishment process is expected to finish *before* 0 o'clock. Therefore, from 0 o'clock on, the baggage service should go smoothly in the next 6 h. Similarly, we will, before 6 o'clock, use the scheduling models for the second 6 h (06:00 A.M–12:00 P.M.) with the updated flights' timetable. Note that in the second scheduling process, some (integer) decision variables should remain the same values to guarantee some tugs could continue their *unfinished delivery* tasks assigned in the first scheduling process. However, different from delivery tasks, tugs are allowed to renew their pickup destinations and execute their newly assigned pickup tasks. By repeating such a procedure, we can adapt our models into the changeable flights' timetables. However, this strategy will raise a new issue: what if there exists at least one flight scheduled across the boundary of two adjacent decision windows, for example, in the above setting, from 5:30 A.M. to 6:30 A.M? Fortunately, we can detour this dilemma by letting two adjacent decision windows crossed. Specifically, we make scheduling decisions with decision windows 00:00 A.M–06:00 A.M., 03:00 A.M–09:00 A.M., 06:00 A.M–12:00 P.M., and so on. It means we make a 6-h scheduling decision while we only execute this decision for the first 3 h (00:00 A.M–03:00 A.M., 03:00 A.M–06:00 A.M., 06:00 A.M–09:00 A.M., and so on). The first 3-h time gap separated from its decision window is termed as the halved execution window. Note that the possible turnaround is no longer than 2 h (see Table IV). Therefore, as long as the overlap of two adjacent decision windows is no shorter than 2 h, the issue that there may exist at least one flight scheduled across the boundary of two adjacent decision windows could be tackled. As an illustration of the crossed decision window and halved execution window (see Fig. 6).

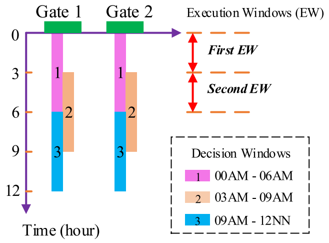


Fig. 6. Crossed decision window and halved execution window.

2) Heuristic Tricks: Due to the large scale and high complexity of the proposed model, it is impractical to solve with exact algorithms, such as branch-and-bound/cut/price. Therefore, we present an improved heuristic algorithm named adaptive large neighborhood search (ALNS), which was first

$$T_l^{kr} \geq T_h^{kr} + c \cdot N_h^{kr} + \tau_{hl} - M(1 - Z_{hl}^{kr}) \quad \forall h, l \in H, h \neq l \quad \forall k, r \tag{64}$$

$$T_l^{kr} \geq T_0^{kr} + \tau_{0l} - M(1 - Z_{0l}^{kr}) \quad \forall l \in H \quad \forall k, r \tag{65}$$

$$T_{|H|+1}^{kr} \geq T_l^{kr} + c \cdot N_l^{kr} + \tau_{l,|H|+1} - M(1 - Z_{l,|H|+1}^{kr}) \quad \forall l \in H \quad \forall k, r \tag{66}$$

$$T_{|H|+1}^{k,r-1} \leq T_0^{k,r} + M(1 - \beta^{k,r}) \quad \forall k, r \in R/\{1\} \tag{67}$$

Authorized licensed use limited to: Imperial College London. Downloaded on September 19,2023 at 15:00:29 UTC from IEEE Xplore. Restrictions apply.

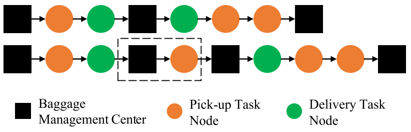


Fig. 7. Illustration of insertion operation via scheme 2). The lower is generated from the upper by inserting one depot and one pickup task node.

proposed by Ropke and Pisinger [43], to solve this problem. Our method is motivated by [44].

In this framework, each tug tour is represented as a sequence of nodes that starts at the depot, followed by discrete task nodes and intermediate depots, and end at the same depot. Each trip is a continuous subsequence between two depots. For concepts of tour and trip, recall Fig. 5. Note that those (logical) depots are physically the same depot.

Subject to the train capacity constraint (six dollies at most) and the amount limit of tug using for each flight (at most four tugs for each flight), the pickup/delivery tasks of each flight are randomly split further into smaller pieces. For example, if one flight requires seven dollies to serve, we can require three dollies first as one task and another four dollies subsequently as the other task. Note that these two tasks can be served by one tug with two different trips or by two different tugs with individual trips. We call each task (after splitting) as a task node.

The removal and insertion operations in ALNS are redefined compared to the classic ALNS. Different from the single trip VRP, removal and insertion operations in multitrip VRP involve not only the task nodes but also the depot. Thus, five insertion schemes are proposed: 1) insert a task node; 2) insert an intermediate depot and a task node; 3) insert a task node and an intermediate depot; 4) insert a single task node trip that starts and ends with an intermediate depot; and 5) insert a tour with only one task node. Fig. 7 shows a tour with two trips conducts an insertion with scheme 2). In addition, three removal heuristics are proposed to determine how task nodes to be removed are selected: 1) random removal; 2) worst removal; and 3) trip removal. The worst removal indicates selecting nodes with largest cost savings [45], and trip removal is to remove trips one by one in increasing order according to the number of task nodes this trip includes. Note that in the removal procedure, if two adjacent depots meet each other after removing a task node located between them, we should remove one of the depots and connect the two adjacent trips together. In each optimization iteration, three removal heuristics are picked by the roulette wheel. Once a node to be removed is selected, this node is always inserted at the position with the least cost increase. The same greedy-based heuristic insertion strategy is also used to generate an initial solution for the successive solution improving. If the solution after initialization is infeasible, a repair procedure is conducted by removing the task nodes that make the solution infeasible and reinserting into the best position with five insertion schemes. Furthermore, to speed up the tour feasibility verification, a TRIE structure [46] is implemented to track the feasibility of partial tours. The ALNS framework is summarized in Algorithm 1, where T is the maximum number of solution-unimproved iterations when got stuck in local optimum, c is the number of task nodes to operate (remove and/or insert) for one tour, and $c_{\max}$ is the upper limit of c. The *machine time limit* is the maximum running time allowed for executing the optimization algorithm. The acceptance criterion is controlled by the classical simulating annealing algorithm (see [47]).

Algorithm 1 ALNS

```
Prior Operations: Split pickup and delivery tasks; Initialize the solution $S$ using insertion operators; Initialize the roulette wheel
Initialization: $S_{best} \leftarrow S$, $t \leftarrow 0$, $c \leftarrow 1$, $T$, $c_{max}$
Input: Flight timetables and gate configuration information
while machine time limit is not reached and $t < T$ do
    Select a removal heuristic $H_r$ using roulette wheel;
    Remove $c$ nodes from $S$ using $H_r$;
    Reinsert $c$ nodes using saving algorithm to get a new solution $S'$;
    if acceptance criterion is met then
        $S \leftarrow S'$, $t \leftarrow 0$, $c \leftarrow 1$
    else
        $t \leftarrow t + 1$, $c \leftarrow c + 1$
        if $c > c_{max}$ then $c \leftarrow 1$
        end if
    end if
    if $S'$ is a new best solution then $S_{best} \leftarrow S'$
    end if
    Update the roulette wheel;
end while
Output: $S_{best}$
```

IV. Simulation Study

All the source data, executable application (which realizes the ALNS), and experimental results with detailed usage instructions are available online at GitHub: https://github.com/Jesmine0902/GSE-Management. In order to protect intellectual property, the source codes would not be disclosed. Nevertheless, the discussion with inspired readers is always welcomed.

A. Simulation Settings

We consider Singapore Changi Airport as a reference. See [48] for its infrastructure (four terminals and about 30 gates per terminal) and [49] for its business statistics (served about 380 000 flights a year via four terminals). That means each terminal/gate serves about 250/8 flights each day. In our simulation, we consider one of the terminals having at most 30 gates, and every two adjacent gates are separated by 150 m. Each gate serves about eight flights on average each day.

During the ground service process, the flights' timetables (i.e., turnaround time period) have already been planned by the flight scheduling problem [9], and the amount of baggage of each flight has already been counted from the ticket-booking system. Note that during ticket booking, the customers are required to provide the information on their registered/checked baggage. Therefore, the turnaround time period and the amount of baggage are known information for our GSE scheduling in reality when the real data are

Authorized licensed use limited to: Imperial College London. Downloaded on September 19,2023 at 15:00:29 UTC from IEEE Xplore. Restrictions apply.

TABLE IV
TURNAROUND TIME PERIOD AND BAGGAGE AMOUNT OF AIRCRAFT

	Large	Medium	Small
Turnaround	90-120	60-90	40-60
Baggage Amount	8-10	5-7	2-4

available. However, for the simulation purpose, we should generate the data with some reasonable assumptions. According to [5] and [7], aircraft with different sizes has different capacity so that the amount of laden baggage and the number of accommodated customers are changeable. In detail, more turnaround time is needed for a larger aircraft and a larger aircraft can convey more baggage. Thus, it is reasonable to consider the aircraft size as an input parameter to determine the values of the turnaround time period and baggage amount. Specifically, for each aircraft size, we give possible value intervals for the two parameters based on [5] and [7]. Besides, in consideration of other uncertain factors influencing the turnaround time period and the baggage amount, the values of the two parameters are not fixed and uniformly sampled from the given intervals.

We assume among all flights that there are three types of aircraft: 1) large; 2) medium; and 3) small size [20], [24]. The value intervals of the turnaround time period (unit: minutes) and the baggage amount (unit: no. of dollies) are listed in Table IV, estimated from [5] and [7]. Note that in our simulation, the turnaround starts from the time instant of in-chock action (not time instant of reaching a gate) and ends at the time instant of off-chock.

In generating the simulation data, we uniformly sample turnaround time and baggage amount from the given intervals for each flight, during which we assume that among all flights each day, one third is large, another one third is medium, and the last one third is small. Besides, we assume that among all flights each day, one half is for turnaround (meaning, this airport is a transferring one), another one quarter is only for arrival (last airport of the journey), and the last one quarter is only for departure (first airport of the journey).

For different numbers of gates (from 1 to 30) at this terminal, we, respectively, generate 15 independent and identically distributed instances in line with the data generating rules above.

For a simple demonstration and without loss of generality, we suppose the flights' timetables of one day keeps unchanged. Therefore, we can schedule GSE for this day only one time with the length of the decision window as 24 h (1440 min).

For parameters in Algorithm 1, T is set to 500 and $c_{\max}$ is the number of tugs used in the initial solution. Plus, the machine time limit for each instance is set to 10 min in the experiments. Other values of parameters for simulation are already given in Tables II and III.

B. Results and Discussions

1) Results: Table VI gives the heuristic results of our GSE scheduling problem. The first column is the number of gates and the next three represent total cost, total travel time of used tugs, and the total number of used tugs, which are averaged over the results of 15 instances, respectively. Note that a heuristic solution is not guaranteed to be globally optimal. It also shows the minimum number of tugs used among 15 instances and $H^0 := (\sum_h H_h^0)/|H|$. Note that in order to save space, here we do not show the results of all gate size. They are available from online complementary materials.

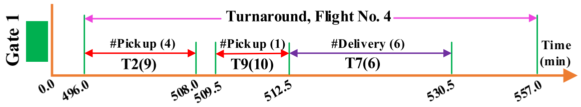


Fig. 8. Tug activities around flight 4. In the figure, *#Pickup(4)* denotes the number of dollies to pickup is 4; *#Delivery(6)* denotes the number of dollies to deliver is 6; and *T2(9)* denotes the tug 2 when in its trip 9.

We give an example of the result of an instance with ten gates for explanation (see at GitHub for data file: *AP10_5.txt*; and result file: *result_AP10_5.txt*). In this instance, the terminal uses 11 tugs to serve 82 flights with 10 gates. For the tug no. 1, its activities (including three trips in this tour) are shown in Table V. From this table, we can see during the first and third trips, this tug only executes the delivery task. However, on the second trip, this tug executes both the pickup task and the delivery task. Specifically, on the second trip, from the time 759.5th min on, it starts to load two dollies of baggage for delivery to flight 46 (at gate 6). At 765.5, it finishes loading and departs the baggage management center. At 768.0, it arrives at gate 5 to serve flight 38 for picking up four dollies of baggage. At 780.0, it finishes picking up and departs gate 5. At 802.5, it arrives at gate 6 to serve flight 46 for delivering two dollies of baggage. At 808.5, it finishes the delivery task and starts to return to the baggage management center. At 811.5, it arrives at the baggage management center and uses 12 min to unload four dollies of baggage picked up from flight 38 (at gate 5). Note that the average time for loading/unloading one dolly of baggage is 3 min. Therefore, it takes 6 min (759.5–765.5) to load two dollies; and 12 min (811.5–823.5) to unload four dollies.

With the same data instance and simulation results, we study the activities around, as an example, flight no. 4, see Fig. 8. It shows flight 4 is parked at gate 1 during the time window [496, 557] (turnaround from in-chock on: 61 min) with five dollies of baggage to pickup and six dollies of baggage to deliver. The pickup task is split into two pieces: 1) using tug 2 in its trip 9 to pickup four dollies and 2) using tug 9 in its trip 10 to pickup one dolly. The delivery task is executed only by tug 7 with six vacant dollies in its trip 6.

2) Discussions: As we can see from Table VI, along with the increase of gate size, the total number of tugs required for baggage service increases as well. Besides, Table V and Fig. 8 show the ground service, such as baggage loading/unloading, for aircraft could be correctly and timely finished within the given time window while respecting the train capacity constraint (six dollies at most), amount limit of tug using for each flight (at most four tugs for each flight), and so on. All the above facts support that our optimal scheduling could indeed guarantee the efficiency of aircraft ground handling. Since this article is the first place to consider such

Authorized licensed use limited to: Imperial College London. Downloaded on September 19,2023 at 15:00:29 UTC from IEEE Xplore. Restrictions apply.

TABLE V
ACTIVITIES OF TUG 1

	Depart Depot		Pick up Events					Delivery Events					Return Depot	
Trip	Loading Win	Dep at	G	F	Arr at	Serv Win	#D	G	F	Arr at	Serv Win	#D	Arr at	Unloading Win
1	[133.0, 151.0]	151.0						2	11	152.0	[152.0, 170.0]	6	171.0	[171.0, 171.0]
2	[759.5, 765.5]	765.5	5	38	768.0	[768.0, 780.0]	4	6	46	802.5	[802.5, 808.5]	2	811.5	[811.5, 823.5]
3	[823.5, 838.5]	838.5						3	21	840.0	[840.0, 855.0]	5	856.5	[856.5, 856.5]

Note: **Arr**: Arrive; **Dep**: Depart; **Serv**: Service; **Win**: (Time) Window; **G**: Gate; **F**: Flight; **#D**: Number of Dollies Used for Pickup/Delivery Event

TABLE VI
RESULTS OF THE ALNS ALGORITHM

Gate size	Avg Total cost	Avg Travel time(min)	Avg No of tugs	Min No of tugs	H^0
5	152.65	221.07	7	6	11
10	251.06	826.20	11	9	8
15	376.21	1880.80	17	15	10
20	467.69	3445.80	20	17	9
25	971.47	6186.93	43	23	9
30	1859.44	10984.73	84	78	9

complicated scheduling problem for multicarriage vehicles with new properties of multitrip, split pickup and delivery, changeable attached dollies, and the like, we do not conduct the comparison experiments here.

V. Conclusion and Future Work

In order to easily locate, manage, and schedule the GSE scattered at an airport, a real-time and high-accuracy GSE tracking device is developed, based on which we can also monitor, identify, and lower the ramp risk such as collisions among GSE and aircraft. Meanwhile, the GSE optimal scheduling problem is studied which allows a reliable and efficient management for resources at an airport. Simulation studies show the feasibility of our proposed schemes. Our solutions are independent of the airports' configurations so that the piratical implementation at every airport is possible. For a demonstration, see our project website: https://alim.algorithmexchange.com/caas/.

As one future work, we would like to incorporate the camera, infrared, and the like sensors into our system to enrich the data collection which will further facilitate the monitor and identification of the ramp risks such as collisions caused by accidents and/or incidents. As another future work, we aim to take into account the following considerations.

1) The total number of drivers involved will be minimized in order to lower the overhead.
2) Each driver cannot continuously work longer than a given time duration to lower the drivers' fatigue.
3) Multiple baggage management centers will be considered.

As the third future work, we aim to transplant the idea of holistic scheduling (for all types of GSE in a unified manner) [3], [25] into our work so that we might further improve the efficiency of ground service and lower the overhead. Ultimately, the flight scheduling, runway/taxiway scheduling, and GSE scheduling are also expected to coordinate together, which might be the most efficient way to lower the turnaround time periods and handle the annually increasing air traffic.

The ultimate purpose of this article is to motivate and facilitate the development of the integrated aviation management system for airports, which supports the joint scheduling and management of flights and ground services.

References

[1] D. Guimarans, P. Arias, M. Tomasella, and C.-L. Wu, "A review of sustainability in aviation: A multidimensional perspective," in *Sustainable Transportation and Smart Logistics*. Amsterdam, The Netherlands: Elsevier, 2019, pp. 91–121.

[2] H. Jackson. (2019). *Business Information—Reports*. Accessed: Nov. 6, 2019. [Online]. Available: http://www.atl.com/business-information/reports/

[3] S. Padrón and D. Guimarans, "An improved method for scheduling aircraft ground handling operations from a global perspective," in *Proc. Asia–Pac. J. Oper. Res.*, vol. 36, no. 4, 2019, Art. no. 1950020.

[4] M. A. A. Makhloof, M. E. Waheed, and U. A. E.-R. Badawi, "Real-time aircraft turnaround operations manager," *Prod. Plan. Control*, vol. 25, no. 1, pp. 2–25, 2014.

[5] D. A. Tabares and F. Mora-Camino, "Aircraft ground handling: Analysis for automation," in *Proc. 17th AIAA Aviation Technol. Integr. Oper. Conf.*, 2017, p. 3425.

[6] J. N. Sanchez and M. A. P. Eroles, "Causal analysis of aircraft turnaround time for process reliability evaluation and disruptions' identification," *Transportmetrica B Trans. Dyn.*, vol. 6, no. 2, pp. 115–128, 2018.

[7] Z. Y. Sng and R. J. Hansman, *A Petri Net Framework for the Representation and Analysis of Aircraft Turnaround Operations*, Massachusetts Inst. Technol., Cambridge, MA, USA, 2019.

[8] J. Landry and S. Ingolia, *Ramp Safety Practices*, vol. 29, Transp. Res. Board, Washington, DC, USA, 2011.

[9] J. P. Pita, C. Barnhart, and A. P. Antunes, "Integrated flight scheduling and fleet assignment under airport congestion," *Transp. Sci.*, vol. 47, no. 4, pp. 477–492, 2012.

[10] S. Chandrasekar and I. Hwang, "Algorithm for optimal arrival and departure sequencing and runway assignment," *J. Guid. Control Dyn.*, vol. 38, no. 4, pp. 601–613, 2014.

[11] A. E. Brownlee, M. Weiszer, J. Chen, S. Ravizza, J. R. Woodward, and E. K. Burke, "A fuzzy approach to addressing uncertainty in airport ground movement optimisation," *Transp. Res. C Emerg. Technol.*, vol. 92, pp. 150–175, Jul. 2018.

[12] L. Xu, C. Zhang, F. Xiao, and F. Wang, "A robust approach to airport gate assignment with a solution-dependent uncertainty budget," *Transp. Res. B Methodol.*, vol. 105, pp. 458–478, Nov. 2017.

[13] G. Pestana, T. R. da Silva, and P. Reis, "Handling airport ground operations using an A—SMGCS approach," in *Proc. IEEE Aerosp. Conf.*, 2011, pp. 1–15.

[14] P. G. Ansola, A. G. Higuera, F. J. Otamendi, and J. de las Morenas, "Agent-based distributed control for improving complex resource scheduling: Application to airport ground handling operations," *IEEE Syst. J.*, vol. 8, no. 4, pp. 1145–1157, Dec. 2014.

[15] M. Heutger and M. Kückelhaus, "Logistics trend radar," *Delivering Insight Today. Creating Value Tomorrow*, 2016. [Online]. Available: https://www.dhl.com/content/dam/dhl/global/core/documents/pdf/dhl-logistics-trend-radar-2016.pdf

[16] I. Kovynyov and R. Mikut, "Digital technologies in airport ground operations," *NETNOMICS Econ. Res. Electron. Netw.*, vol. 20, no. 1, pp. 1–30, 2019.

[17] A. Lim and K. Zhang, "A robust RFID-based method for precise indoor positioning," in *Proc. Int. Conf. Ind. Eng. Appl. Appl. Intell. Syst.*, 2006, pp. 1189–1199.

[18] CAAS. (2018). *Raising Airport Capabilities*. Accessed: Nov. 3, 2019. [Online]. Available: https://www.caas.gov.sg/who-we-are/areas-of-responsibility/developing-the-industry/raising-airport-capabilities

[19] Q. Li, J. Bi, and Z. Li, "Research on ferry vehicle scheduling problem within airport operations," in *Proc. IEEE 10th Int. Symp. Comput. Intell. Design (ISCID)*, vol. 2, 2017, pp. 248–251.

Authorized licensed use limited to: Imperial College London. Downloaded on September 19,2023 at 15:00:29 UTC from IEEE Xplore. Restrictions apply.

[20] W. Guo, P. Xu, Z. Zhao, L. Wang, L. Zhu, and Q. Wu, "Scheduling for airport baggage transport vehicles based on diversity enhancement genetic algorithm," *Nat. Comput.*, to be published.

[21] P. Zhao, W. Gao, X. Han, and W. Luo, "Bi-objective collaborative scheduling optimization of airport ferry vehicle and tractor," *Int. J. Simulat. Model.*, vol. 18, no. 2, pp. 161607–161620, 2019.

[22] Z. Xing and G. Lian, "Cooperative game theoretical research for aircraft deicing operation scheduling," in *Proc. IEEE 10th World Congr. Intell. Control Autom.*, 2012, pp. 2407–2411.

[23] J. Y. Du, J. O. Brunner, and R. Kolisch, "Planning towing processes at airports more efficiently," *Transp. Res. E Logist. Transp. Rev.*, vol. 70, pp. 293–304, Oct. 2014.

[24] Z. Zhou, S. Liu, and K. Huang, "Research on airport trailer emergency scheduling model based on genetic simulation annealing algorithm," in *Proc. IOP Conf. Mater. Sci. Eng.*, vol. 383, 2018, Art. no. 012044.

[25] S. Padrón, D. Guimarans, J. J. Ramos, and S. Fitouri-Trabelsi, "A bi-objective approach for scheduling ground-handling vehicles in airports," *Comput. Oper. Res.*, vol. 71, pp. 34–53, Jul. 2016.

[26] P. Henkel and A. Sperl, "Real-time kinematic positioning for unmanned air vehicles," in *Proc. IEEE Aerosp. Conf.*, 2016, pp. 1–7.

[27] W. Li and J. Wang, "Effective adaptive Kalman filter for MEMS-IMU/magnetometers integrated attitude and heading reference systems," *J. Navig.*, vol. 66, no. 1, pp. 99–113, 2013.

[28] R. Gonzalez and P. Dabove, "Performance assessment of an ultra low-cost inertial measurement unit for ground vehicle navigation," *Sensors*, vol. 19, no. 18, p. 3865, 2019.

[29] T. Meyer, "Grid, ground, and globe: Distances in the GPS era," *Surveying Land Inf. Sci.*, vol. 62, no. 3, pp. 179–202, 2002.

[30] Wikimedia. (2015). *Dolly for ULD and Dollies for Loose Luggages*. Accessed: Nov. 7, 2019. [Online]. Available: https://commons.wikimedia.org/wiki/File:Dolly_for_ULD_and_dollies_for_loose_luggages.JPG

[31] E. Mazor, A. Averbuch, Y. Bar-Shalom, and J. Dayan, "Interacting multiple model methods in target tracking: A survey," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 34, no. 1, pp. 103–123, Jan. 1998.

[32] D. Simon, *Optimal State Estimation: Kalman, H-Infinity, and Nonlinear Approaches*. New York, NY, USA: Wiley, 2006.

[33] X. R. Li and V. P. Jilkov, "Survey of maneuvering target tracking. Part I. Dynamic models," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 39, no. 4, pp. 1333–1364, Oct. 2003.

[34] P. Jiménez, F. Thomas, and C. Torras, "3D collision detection: A survey," *Comput. Graph.*, vol. 25, no. 2, pp. 269–285, 2001.

[35] G. B. Dantzig and J. H. Ramser, "The truck dispatching problem," *Manag. Sci.*, vol. 6, no. 1, pp. 80–91, Oct. 1959.

[36] Z. Luo, H. Qin, W. Zhu, and A. Lim, "Branch and price and cut for the split-delivery vehicle routing problem with time windows and linear weight-related cost," *Transp. Sci.*, vol. 51, no. 2, pp. 668–687, 2016.

[37] Q. Lu and M. Dessouky, "An exact algorithm for the multiple vehicle pickup and delivery problem," *Transp. Sci.*, vol. 38, no. 4, pp. 503–514, 2004.

[38] D. Cattaruzza, N. Absi, D. Feillet, and T. Vidal, "A memetic algorithm for the multi trip vehicle routing problem," *Eur. J. Oper. Res.*, vol. 236, no. 3, pp. 833–848, 2014.

[39] T. Vidal, G. Laporte, and P. Matl, "A concise guide to existing and emerging vehicle routing problem variants," *Eur. J. Oper. Res.*, vol. 289, no. 2, pp. 401–416, 2020.

[40] D. Pecin, C. Contardo, G. Desaulniers, and E. Uchoa, "New enhancements for the exact solution of the vehicle routing problem with time windows," *INFORMS J. Comput.*, vol. 29, no. 3, pp. 489–502, 2017.

[41] A. Subramanian, E. Uchoa, A. A. Pessoa, and L. S. Ochi, "Branch-and-cut with lazy separation for the vehicle routing problem with simultaneous pickup and delivery," *Oper. Res. Lett.*, vol. 39, no. 5, pp. 338–341, 2011.

[42] T. Gschwind and S. Irnich, "Effective handling of dynamic time windows and its application to solving the dial-a-ride problem," *Transp. Sci.*, vol. 49, no. 2, pp. 335–354, 2014.

[43] S. Ropke and D. Pisinger, "An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows," *Transp. Sci.*, vol. 40, no. 4, pp. 455–472, 2006.

[44] V. François, Y. Arda, and Y. Crama, "Adaptive large neighborhood search for multitrip vehicle routing with time windows," *Transp. Sci.*, vol. 53, no. 6, pp. 1706–1730, 2019.

[45] G. Clarke and J. W. Wright, "Scheduling of vehicles from a central depot to a number of delivery points," *Oper. Res.*, vol. 12, no. 4, pp. 568–581, 1964.

[46] P. Brass, *Advanced Data Structures*, vol. 193. Cambridge, U.K.: Cambridge Univ. Press, 2008.

[47] F. Y. Vincent, A. P. Redi, Y. A. Hidayat, and O. J. Wibowo, "A simulated annealing heuristic for the hybrid vehicle routing problem," *Appl. Soft Comput.*, vol. 53, pp. 119–132, 2017.

[48] Wikipedia. (2019). *Infrastructure of Singapore Changi Airport*. Accessed: Nov. 27, 2019. [Online]. Available: https://en.wikipedia.org/wiki/Infrastructure_of_Singapore_Changi_Airport

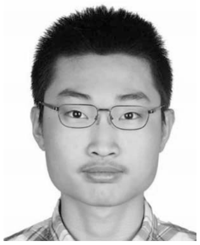
[49] Changi. (2020). *Traffic Statistics*. Accessed: Mar. 8, 2020. [Online]. Available: http://www.changiairport.com/corporate/our-expertise/air-hub/traffic-statistics.html

Shixiong Wang (Graduate Student Member, IEEE) received the B.Eng. degree in detection, guidance and control technology and the M.Eng. degree in systems and control engineering from the School of Electronics and Information, Northwestern Polytechnical University, Xi'an, China, in 2016 and 2018, respectively. He is currently pursuing the Ph.D. degree with the Department of Industrial Systems Engineering and Management, National University of Singapore, Singapore.

His research interest includes continuous and discrete optimization with applications in signal processing (especially optimal estimation theory and target tracking).

Yuxin Che received the B.Eng. and M.Eng. degrees from the Department of Computer Science, School of Information Science and Technology, Xiamen University, Xiamen, China, in 2015 and 2018, respectively. She is currently pursuing the Ph.D. degree with the Department of Industrial Systems Engineering and Management, National University of Singapore, Singapore.

Her research interests include vehicle routing problem, packing problem, machine scheduling, and other discrete optimization problems.



Huangjie Zhao received the B.Eng. degree in information engineering and media from the School of Electronical and Electronic Engineering, Nanyang Technological University, Singapore, in 2017. He is currently pursuing the Ph.D. degree with the Department of Industrial Systems Engineering and Management, National University of Singapore, Singapore.

His research interests include mixed-integer linear programming, vehicle routing problems, and robust optimization.

Andrew Lim received the Bachelor of Computer Science and Doctor of Philosophy degrees in computer and information sciences from the University of Minnesota, Minneapolis, MN, USA, in 1987 and 1992, respectively.

He is a Professor with the Department of Industrial Systems Engineering and Management, National University of Singapore, Singapore. He attaches great importance to the synergy between world-class scientific research and innovation. His research interest includes big data analytics, demand generation, and supply management problems in the domains of logistics, transportation, and healthcare. Many of his works have been published on leading journals, such as Operations Research, Management Science, and Transportation Science.

Prof. Lim is one of the five Returning Singaporean Scientists and a recipient of China 1000 Talents Scheme. For more information, see his personal website: https://www.limandrew.org/.

Authorized licensed use limited to: Imperial College London. Downloaded on September 19,2023 at 15:00:29 UTC from IEEE Xplore. Restrictions apply.