This is the Pre-Published Version.

The following publication Y. Li, X. Wen, T. -M. Choi and S. -H. Chung, "Optimal Establishments of Massive Testing Programs to Combat COVID-19: A Perspective of Parallel-Machine Scheduling-Location (ScheLoc) Problem," in IEEE Transactions on Engineering Management, vol. 71, pp. 13380-13395, 2024 is available at https://doi.org/10.1109/TEM.2022.3199039.

Optimal establishments of massive testing programs to combat COVID-19: A perspective of parallel-machine Scheduling-location (ScheLoc) problem

Yantong Li, Xin Wen, Tsan-Ming Choi, and Sai-Ho Chung, *Member, IEEE*

***Abstract*—Massive testing to identify COVID-19-infected people plays a crucial role in combating COVID-19. However, from the perspective of facility location problems, many current massive testing programs are not properly set, leading to unreasonable travelling distances, long makespan, unbalanced workload, and long queues. This study proposes a decision framework for developing massive testing programs. Specifically, a bi-objective parallel-testing-site Scheduling-location (ScheLoc) model is formulated, which simultaneously minimizes the makespan and total travelling distance. The former can help reduce the time length of potential virus spread, and the latter can help alleviate the risk of virus spread and traveler inconvenience. To solve the proposed bi-objective ScheLoc problem, in addition to the standard ϵ-constraint method, we further develop two novel methods. The first one iteratively solves simpler approximate MIP models (IMIP). The second innovatively extends the classical logic-based Benders decomposition approach to solve bi-objective problems (B-LBBD). A Hong Kong-based case study shows that the proposed decision framework can significantly reduce the makespan and travelling distance (with a mean of 13% and 5.1%, respectively) and enhance workload-balancing. Besides, the developed solution methods, especially the B-LBBD, outperform the adapted ϵ-constraint method in various aspects.**

***Managerial relevance statement*—The decision framework of massive testing programs establishment developed in this study provides scientific and systematic suggestions for the testing site location decisions, community assignment decisions, and community scheduling decisions for governmental decision-makers. Using the developed framework, the shortened makespan and travelling distances of the tested residents bring significant benefits to the society. By scientifically assigning communities to testing sites, our proposed decision framework can reduce the risk of virus-spread-on-the-way, which simultaneously enhances the performance of the massive testing programs and protects the public. With the reduction in travelling distances and waiting time, the overall time required to finish all the test jobs (i.e., makespan) can be significantly reduced. Responsive actions taken by governments can alleviate public panic and help maintain the stability of the region, which in turn guarantees the success of the subsequent anti-virus policies launched by authorities. Moreover, the proposed model can better balance the workload assigned to the staff at different testing sites, improve staff satisfaction and help protect their health.**

***Index Terms*—COVID-19; Infection testing; Bi-objective optimization; Scheduling; Facility location.**

This study is supported by the China Postdoctoral Science Fund under Grant 2022M710018. We sincerely thank the department editor, Prof. Nadja Damij, and three anonymous reviewers for their great help and constructive comments which improve the study significantly. (*Corresponding author: Xin Wen.*)

Dr. Li is with School of Maritime Economics and Management, Dalian Maritime University, 116026 Dalian, China. His email is:yantong.li@dlmu.edu.cn.

Dr. Wen is with Department of Industrial and Systems Engineering, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong. Her email is windy.wen@polyu.edu.hk.

Prof. Choi is with Centre for Supply Chain Research, University of Liverpool Management School, Chatham Building, Liverpool L69 7ZH, the United Kingdom. His email is: tmjchoi@gmail.com.

Dr. Chung is with Department of Industrial and Systems Engineering, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong. His email is: nick.sh.chung@polyu.edu.hk.

I. Introduction

A. Background and motivation

Since late 2019, Corona Virus Disease 2019 (COVID-19 or SARS-Cov-2) has spread rapidly worldwide [1]–[3]. Strengthening public health systems has thus become critical [4]–[7]. Efficient testing of COVID-19 infections is essential to provide proper healthcare for patients and protect the uninfected population. Several protocols have been developed for quick testing of massive populations, like the SARS-Cov-2 nucleic acid test, the antigen test, and the antibody test. According to the Centers for Disease Control and Prevention, the COVID-19 tests can diagnose infections of both symptomatic and asymptomatic cases, guiding contact tracing, isolation requirements, and treatment arrangements [8]. Therefore, massive testing programs play a pivotal role in assessing the risk of releasing lockdown measures, avoiding new outbreak waves, and facilitating the resumption of normal societal and economic activities for the region, which have been widely applied in many countries.

Although the importance of massive testing programs has been well recognized, due to the lack of optimization tools and the fact that the virus usually appears suddenly, the government usually launches the massive testing programs in a hurry, without scientific and systematic analyses and evaluations. Thus, the massive testing programs are usually criticized by having low efficiency and effectiveness in resisting the COVID-19 outbreak. For example, the testing locations are generally selected based on decision-makers' experiences and previous decisions, or just randomly. However, the tested population is distributed unevenly in the communities located in the region, while the distances between communities with the randomly selected testing site locations vary significantly, which may lead to unreasonable travelling distances for the tested persons, increasing the risks of being infected or infecting others during the way to/back from the testing sites.

Besides, the communities that require tests are also randomly (or according to decision-makers' experiences) assigned to the established testing sites, causing an unbalanced flow of people at different testing sites. This inevitably leads to congestion at some sites and uneven utilization of facilities and human resources. It is reported that, in the massive testing program of Shenyang, China, it took 15 minutes or even less to finish the testing at some sites, while in some places, people had to wait around 50 minutes or longer, making them angry. Besides, in the massive testing of Chengdu in Dec 2020, medical workers had to work for 24 hours a day at some crowded sites, while the pressure at some other sites was much lighter[1]. Heavy workload may cause dissatisfaction and fatigue among testing staff. Fatigue of medical personnel is dangerous as it will impair the body's resistance to the virus, increasing the risks of being infected. Moreover, there also does not exist a scientific guideline for authorities to arrange the schedule of the tested communities. Long queues and long waiting times during peak times are thus commonly seen at some testing sites, making it difficult to control social distances, increasing the possibility of cross-infection, and causing dissatisfaction and complaints among residents. Therefore, it is seen that the current massive testing programs usually face poor decisions regarding testing locations, community assignments, and community scheduling, causing low efficiency. However, speed is essential for mass testing, and a scientific decision framework is urgently needed.

In this study, to address the challenges mentioned above, we aim to build a decision framework for optimally establishing massive testing programs, determining the location of testing sites, and the assignment and scheduling of the tested communities. Specifically, through analyzing the problem characteristics, it is found that the logic of the related decision-making behind is quite similar to the integrated parallel-machine ScheLoc problem in the optimization domain. The "ScheLoc" problem refers to the integrated scheduling and location problem, in which the facility location decisions, job-machine assignment decisions, and job scheduling decisions are made simultaneously. In the parallel-machine ScheLoc problem, machine locations must be selected from a set of candidate options, jobs that are dispersed on a network must be assigned to the located machines, and the jobs assigned to the same machine are scheduled [9]. Generally, the optimization objective of the parallel-machine ScheLoc problem is to minimize the makespan (i.e., the time to complete the last job). Applying a similar logic, the decisions to be made in the massive testing program establishment problem include:

i) The testing site (can be regarded as a "machine") locations shall be determined from a list of candidates in the region;
ii) The communities to be tested shall be assigned to the established testing sites (the testing task for a community can be regarded as a "job");
iii) The communities assigned to the same testing site should be scheduled so that the residents can be informed of the precise testing time to arrange the trip better, thus avoiding long queues.

[1]https://www.globaltimes.cn/content/1209827.shtml. Retrieved on 24 Nov 2021.

Based on the discussions above, we can mathematically formulate the massive testing program establishment problem by following the parallel-machine ScheLoc mechanism [10]. Specifically, in our problem setting, authorities decide to conduct testing for a list of communities (e.g., in response to a newly detected positive case in the region). Accordingly, several testing sites are to be settled at a set of potential locations, while the residents to be tested are located in different communities. Being assigned to a specific testing site, a resident must travel from his/her community to the related site with a travel time. The time point that the resident arrives at the testing site is then named as the release date. As the travelling time depends on the distance between the community and the testing site, the release dates of the residents from different communities may vary. Moreover, to avoid long queues, it is necessary to appropriately schedule the communities assigned to the same testing site (i.e., when to arrive at the site for a specific community). The makespan of the overall system thus equals the time when the last community finishes the test. Generally, authorities launch massive testing programs because positive cases are confirmed in the region. Therefore, minimizing the makespan of the testing program can help reduce the time length of potential virus spread. Besides, · public, which can help alleviate the risk of virus spread and traveller inconvenience. Accordingly, we develop a bi-objective parallel-testing-site ScheLoc problem to simultaneously minimize the makespan and travelling distance during the COVID-19 outbreak. We formally describe the problem and formulate it as a bi-objective mixed-integer linear program (MILP).

Methodological advancements− Regarding the solution methodology, we first follow the existing bi-objective optimization literature to apply the widely used ϵ-constraint method combined with the proposed MILP. However, computational experiments show its incapability to solve large instances. Therefore, we further develop two customized solution methodologies to deal with practical-sized instances in order to enhance the practical applicability of the proposed decision framework. The first is to iteratively solve approximate mixed-integer programming (MIP) models of the original problem with smaller scales (i.e., IMIP). For the second, we innova-tively extend the classical logic-based Benders decomposition approach, originally designed for single-objective optimization problems, to efficiently deal with bi-objective problems (i.e., B-LBBD). In the B-LBBD method, the original problem is decomposed into a master problem that takes care of the total distance objective and a subproblem that computes the makespan objective. The master problem is solved by a branch-and-cut (BC) algorithm to minimize the total distance. The subproblem is solved to generate cuts dynamically added to the master problem to help find solutions with a shorter makespan. One distinctive feature of the B-LBBD method is that it can obtain an approximate Pareto front by solving the model only once with cuts dynamically added to the search tree nodes. It is thus superior to the ϵ-constraint method that iteratively solves MILPs.

B. Contributions

This study contributes to the literature and advanced practices, which we elaborate on as follows.

Anti-COVID-19 Practice– To the best of our knowledge, this study is the first research analytically and scientifically exploring the location, assignment, and scheduling problems for massive testing programs during the COVID-19 pandemic, by transforming the problem into a bi-objective parallel-machine ScheLoc problem. The findings derived are of great practical significance for the governmental policy-making to fight against the outbreak of COVID-19, which can provide insightful guidelines for public health departments to make optimal decisions when the testing demands arise urgently. Second, the bi-objective optimization framework proposed in this study is beneficial to improve the test efficiency (i.e., reduce the overall time needed) and valuable for reducing the risks of virus spread (i.e., reduce the total travelling distance). Third, through scientific and systematic arrangements, the long queues of tested persons can be avoided, while the workload for medical staff can be balanced.

Methodological advancements and contribution– Regarding the solution methodology, to deal with the proposed bi-objective parallel-machine ScheLoc problem and to obtain Pareto solutions efficiently, we (i) propose an iterative MIP (IMIP) method and (ii) innovatively develop a bi-objective logic-based Benders decomposition (B-LBBD) approach. The IMIP method obtains a set of Pareto solutions by iteratively solving a set of approximate MIP models with smaller scales (e.g., fewer variables and constraints). The B-LBBD method, on the other hand, decomposes the original bi-objective ScheLoc problem into a master problem (MP) that optimizes one objective (i.e., total travelling distance) and a subproblem (SP) to handle the other objective (i.e., makespan). To the best of our knowledge, this is the first study that devises the classical logic-based Benders decomposition method (originally designed for single-objective optimization problems) to handle bi-objective optimization problems. The proposed B-LBBD solution approach is significantly superior in producing a set of Pareto solutions by solving the MP only once using the Branch-and-cut technique that operates on a single search tree. Note that the application of our proposed novel B-LBBD method is not limited to the problem investigated in this study. Instead, it can potentially enhance the solution efficiency for other bi-objective optimization problems. This study thus makes significant methodological advancements for the logic-based Benders decomposition methodology and theoretically contributes to the bi-objective optimization literature. *Remarks–* The parallel-machine ScheLoc problem has recently received increasing attention due to its high practical applicability. Our study formulates the testing problem as a typical parallel-machine ScheLoc problem. Different from the existing works in the area, we first introduce a bi-objective optimization scheme into the decision framework so that we can explore the impacts of different optimization objectives on the final solution. This enhances the robustness of our findings concerning optimization objectives.

C. Paper structure

The remainder of this paper is structured as below. First, Section 2 reviews the related literature. The problem studied, and the proposed mathematical formulations are presented in Section 3. Next, Section 4 develops solution approaches for the problem. A case study based on Hong Kong Universal Community Testing Programme is presented in Section 5 to demonstrate the merits of the proposed decision framework. Section 6 then reports computational experiments, based on which Section 7 discusses the managerial implications of the study. Finally, Section 8 concludes this study.

II. Literature review

The massive testing program establishment studied in the paper is formulated as a bi-objective parallel-testing-site ScheLoc problem. This section reviews the related literature on the ScheLoc problem and multi-objective optimization.

A. Single- and parallel-machine ScheLoc problem

The ScheLoc problem is a new and growing research area, first introduced by Hamacher and Hennes in 2002. The ScheLoc problem is an integrated decision framework that simultaneously considers the job scheduling (assignment) and machine location problems that are traditionally two separated fields [9]. Such integration can overcome the traditional sequential solution process's shortcomings (e.g., sub optimality). The typical scheduling and location problems are known as NP-hard, and so is the ScheLoc problem. In a ScheLoc problem, the optimization objective is to minimize specific scheduling criteria (like makespan) by identifying the optimal locations to place machines, deciding optimal assignments of jobs to the machines, and making an optimal job schedule for each machine. The ScheLoc problem is further divided into the single-machine ScheLoc problem and the parallel-machine ScheLoc problem regarding the number of machines considered, as discussed in the following. The single-machine ScheLoc problem is pioneered by Hennes *et al.* [11], which optimizes the location of a machine in a given network to minimize makespan. Later, Elvik *et al.* [12] develop polynomial-time algorithms for the single-machine ScheLoc problem where a single machine can be located anywhere on a given planar. Besides, Kalsch *et al.* [13] propose two objectives for the single-machine ScheLoc problem, while a branch-and-bound methodology is developed to solve the problem efficiently.

In recent years, the parallel-machine ScheLoc problem has received increasing attention due to its high practical applicability. Hessler *et al.* [9] investigate a discrete parallel-machine ScheLoc problem that aims to minimize the makespan. In the problem setting of Hessler *et al.* [9], a subset of locations shall be selected from several discrete locations to place machines, while a set of jobs are to be assigned to and processed on the located machines. Several clustering heuristics in which jobs are grouped into clusters to be allocated to machines are developed as the solution methodology. A similar study can be found in Wang *et al.* [14]. However, different from [9], Wang *et al.* [14] propose a new modelling approach for the discrete parallel-machine ScheLoc problem using the network flow idea. Through computational experiments on 1450 instances, Wang *et al.* [14] show that their new model performs better than Hessler *et al.* [9] by solving more instances to optimality within the same time limit. Compared to Wang *et al.* [14] and Hessler *et al.* [9], we study a bi-objective parallel machine ScheLoc problem to simultaneously minimize the testing completion time and the total traveling distance of residents. In addition, we propose a new bi-objective MILP model and develop a novel method to solve the problem efficiently. Recently, Kramer and Kramer *et al.* [15] propose a novel arc-flow formulation for the discrete parallel-machine ScheLoc problem. Both exact (column generation) and heuristic solution methodologies are developed by Kramer and Kramer *et al.* [15]. The authors report identifying optimal solutions for all benchmark instances extracted from the existing literature and obtaining small optimality gaps for new challenging problems. On the other hand, Liu *et al.* [16] point out that most existing ScheLoc studies only consider deterministic situations. However, manufacturing systems are constantly challenged by job processing time uncertainties. Accordingly, Liu *et al.* [16] utilize the two-stage stochastic programming formulation to explore a stochastic parallel-machine ScheLoc problem in which the job processing times are uncertain.

In this study, we transform the massive testing program establishment problem into a parallel-machine ScheLoc problem which aims to minimize the makespan and the overall travelling distance of the tested persons.

B. Multi-objective optimization methods

Various methodologies have been proposed to deal with optimization problems with multiple objectives [17]–[20]. Generally, these methods are classified into preference-based approaches and generating approaches. The former type considers the decision maker's preference during the decision process (like goal programming, global criterion methods, and goal-attainment), thus providing only one solution. On the other hand, the latter type, such as weighted sum, ϵ-constraint, and multi-objective evolutionary methods, can derive a set of Pareto-optimal solutions for the decision-maker. Among these generating approaches, the weighted sum method may suffer as many different combinations of weightings can result in the same solution, while the multi-objective evolutionary method can only provide approximate Pareto solutions. Thus, the ϵ-constraint method is a useful approach, especially for bi-objective optimization problems [17]. The primary mechanism of the ϵ-constraint method is as follows. The original bi-objective optimization problem is transformed into a series of mono-objective problems with one principal objective, while the other objective is formulated as a ϵ-constraint. The set of mono-objective problems can be solved to obtain Pareto solutions by changing the value of ϵ. Based on the ϵ-constraint method, the Pareto front is obtained to provide guidelines for the policymaker [16]. Although the ϵ-constraint method has shown advantages, it still suffers from the difficulty of repeatedly solving the transformed mono-objective problems. When the original problem is complex, the transformed mono-objective problem may be further complicated by the insertion of the ϵ-constraint. Therefore, the repetition in solving the mono-objective optimization problems is computationally expensive and time-consuming and cannot handle large-scale problems. As a result, efficient solution algorithms are needed. The Logic-based Benders decomposition method is a promising one. Next, we briefly review the advances of the Logic-based Benders decomposition method.

C. Logic-based Benders decomposition method

The Benders decomposition (BD) algorithm is developed by Benders *et al.* [21] to handle complicated MIPs. The basic idea of the BD method is to transform the original problem into simpler subproblems by fixing the so-called "complicated variables." The classic BD decomposes the original problem into a master problem (MP) and a subproblem (SP), which are iteratively solved. The MP is augmented by Benders cuts generated from the solution of the SP. The SP should be a linear program in a classic BD method, and Benders cuts are generated from its dual information. This restricts the applicability of the classic BD algorithm for solving problems where the SPs are not linear programs. Hooker *et al.* [22] generalize the classic BD into the logic-based Benders decomposition (LBBD) approach by allowing the SP to take any form, while Benders cuts are derived through its logic information. An advanced implementation of the LBBD method is to generate cuts through iteratively solving the SP upon finding a feasible solution so that the MP is solved only once [23]. This advanced implementation is called as Branch and Check in the literature, implying that cuts are added when feasible solutions are identified in the Branch-and-cut search tree, which has been proven to be effective in solving a wide range of single-objective combinatorial optimization problems, e.g., the parallel machine scheduling problem [23] and the operating room planning problem [24]. In short, the merits of the LBBD method lie in (i) decomposing the original complex problem into simpler subproblems, (ii) the subproblems can take any form, and (iii) the MP can be operated on a single search tree and cuts can be added during the solving process. This paper innovatively extends the LBBD method to solve bi-objective optimization problems, which is novel in the literature.

D. Decision making during the COVID-19 pandemic

With the rapid spread of COVID-19, academia has paid great attention to improving the decision-making under the global pandemic [25]–[28]. The readers are referred to Kaplan *et al.* [29] for comprehensive discussions about the problems faced, models constructed, and suggestions offered in response to COVID-19. We briefly discuss some recent works as follows.

One research stream focuses on improving the forecasting accuracy of the virus spread trend. For instance, Guo *et al.* [30] extend the classical susceptible-infectious-recovered compartmental model to characterize the transmission procedure of

COVID-19. In a similar study, Chen *et al.* [31] develop a time-dependent susceptible-infected-recovered model to forecast the COVID-19 spread trend, which tracks the transmission and recovery rates at a given time point. Differently, Nikolopoulos*et al.* [32] apply the models in statistics, epidemiology, machine learning, and deep learning, and develop a hybrid approach based on nearest neighbors and clustering to predict the growth rate of COVID-19.

Past experiences have demonstrated that global epidemic (like influenza) generally causes significant disruptions to supply chains [33], which also applies to the COVID-19 crisis. For example, Nikolopoulos *et al.* [32] witness the excess demand for products and services from the COVID-19 epidemic. To address the associated supply chain disruptions, Nikolopoulos *et al.* [32] propose a new forecasting method to estimate the impact of the excess demand. Additionally, Singh *et al.* [34] propose a simulation model for the public food distribution system to deal with the disruptions in food supply chains brought by COVID-19. Specifically, Singh *et al.* [34] investigate different scenarios to highlight the difficulties in matching supply and demand and the growth in infected cases. In addition, Nagurney *et al.* [35] captures the nature of COVID-19 that brings people illness or death and studies the supply chain disruptions caused by labor shortages using game theory.

Besides, some studies explore the innovations in supply chain operations after the outbreak. For example, Choi *et al.* [36] analyzes how the innovations in logistics and technologies can bring the traditional static service operations to the "bring-service-near-your-home" type of operations. It is revealed that the subsidies provided by the government are crucial for the success of this new type of operation. With the outbreak of COVID-19, physicians' knowledge has become a critical and useful indicator to improve demand management in the healthcare industry. Accordingly, Govindan *et al.* [37] establish a novel decision support system for healthcare supply chains based on fuzzy inference systems, which firstly groups community residents according to the risk levels of immune systems, age, and pre-existing diseases.

Similar to the literature discussed above, this study also explores the decision-making during the COVID-19 pandemic. However, different from the existing studies, we firstly investigate the optimal location decisions for testing sites and the assignment and scheduling decisions for tested persons mathematically, which is novel and valuable for both the literature and the practice.

III. Problem description and formulation

As discussed, this study aims to build a scientific massive testing program establishment decision framework regarding the location of testing sites and the assignment and scheduling of tested people during the outbreak of COVID-19. The problem considered is briefly described in Fig 1. There is a list of communities where the residents need to take the test. In the testing practice in China, residents are organized into groups of communities. There is an administrative team for each community. The authority plans to establish several testing sites at some candidate locations. Therefore, the authority needs to decide where to place the testing sites, which community to be assigned to which site, and the scheduling of the communities assigned to the same site. Based on the distinctive characteristics, the problem can be transformed into a parallel-machine ScheLoc problem. Accordingly, a customized bi-objective parallel-testing-site ScheLoc model is proposed, which is described in this section. Besides, it should be pointed out that our model setting is motivated by the compulsory testing programs established in China. The massive testing program plays an essential role in China's anti-pandemic policy, and all residents are regulated to perform tests in a massive testing program. Thus, it is reasonable for us to consider that all residents would perform the test at the scheduled time required by the government.

From Fig 1, we see that eight communities need to be tested. The authority must select two locations from 4 candidate locations to launch testing sites. Then the authority needs to make a scheduling plan to assign communities to open testing sites and arrange the testing sequences of communities. Given some input information, the authority uses the decision platform to establish a rapid testing program. The Gant chart in the right bottom indicates communities' assignment and scheduling results. In the example, we have 8 communities. Communities 1, 2, 3, 4 are assigned to location 2 and their testing sequence is 1, 4, 3, 2. Similarly, the other four communities are assigned to location 4, and processed in sequence as 7, 5, 6, 8. The idle time r_{12} indicates that location 2 starts testing when community 1 is released. We consider a region with a

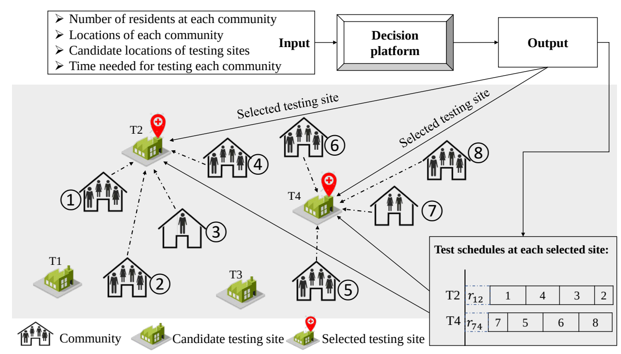

Fig. 1: Problem description for the massive testing program establishment.

list of communities where residents to be tested are located. A community consists of a set of residents living in the adjacent areas. The set of test jobs for the communities is denoted by J,indexed by i, j. The testing task for a community is regarded as a job. Besides, the authority decides that there will be at most m identical testing sites to be established for the massive testing program. Then the authority must select from a set K (indexed by k) of discrete candidate locations in the region to build the testing sites, like hospitals, community health service centers, and sports halls. Note that each candidate location can hold at most one testing site, while a testing site can process at most one test job each time. In other words, at a testing site, only one community can be tested each time, and preemption is not allowed. Moreover, each community j is featured with a deterministic processing time p_j. Note that each community may involve a different size of the population. Therefore, the processing time p_j for community j is decided by the number of residents in it. As discussed, the release dates of the test job for different communities (i.e., the earliest starting time for the residents of the community to be tested) may vary according to the community's locations and that of the assigned testing site. Accordingly, each test job j has a specific release date r_{jk} which is dependent on testing site k.

In this study, we aim to minimize the makespan, which is the maximum completion time of the testing for all communities, i.e., C_{max}, and the total traveling distance of the tested communities. We consider the situation where rapid testing for a large population should be performed. In this case, the principle objectives should be timeliness and travel distances. Since the optimization results of the two objectives significantly affect the spread of the virus. Therefore, the cost component seems not that important for decision-makers. The decisions to be made include the location of the testing sites w_k, the assignment of each test job to the testing sites v_{jk}(i.e., to assign which community to which testing site), the scheduling of the test jobs at each testing site x_{ij}(i.e., the testing sequences of the communities assigned to a testing site), the timing to start tests of each community C_j, and the maximum completion time of all tests C_{max}. The following assumptions are made: 1) the testing of each community cannot be preempted; 2) each community is assigned to exactly one testing site; 3) the release date of each community is dependent on the distance between its location and that of its assigned testing site; and 4) the testing time of each community is positively correlated with the number of residents in it, which can be estimated easily using the number of residents and the testing speed. The sets, parameters, and decision variables used in this study are summarized in TABLE I.

Sets	
J	The set of test jobs, indexed by i, j;
K	The set of candidate testing sites, indexed by k;
Parameters	
r_{jk}	The traveling distance between job j and site k;
p_j	The processing time of job j;
m	The number of testing sites to be established;
B	A large number;
Variables	
x_{ij}	Equal to 1 if job i is processed (not necessarily immediately) before job j; Otherwise, 0;
w_k	Equal to 1 if a testing site is placed at candidate location k; Otherwise, 0;
v_{jk}	Equal to 1 if test job j is processed at a testing site located at candidate location k; Otherwise, 0.

TABLE I: Summary of the notation

With the above defined notations, the bi-objective parallel-testing-site ScheLoc problem can be formulated into an MILP by using the linear ordering modelling approach as follows (model LO):

$$\min\ f_1 = C_{max} \tag{1}$$

$$\min\ f_2 = \sum_{j\in J}\sum_{k\in K} r_{jk}v_{jk} \tag{2}$$

s.t.

$$C_{max} \ \geq\ C_j \qquad \forall\, j \in J \tag{3}$$

$$\sum_{k\in K} v_{jk} = 1 \qquad \forall\, j \in J \tag{4}$$

$$\sum_{k\in K} w_k \leq m \tag{5}$$

$$v_{jk} \ \leq\ w_k \qquad \forall\, j \in J, k \in K \tag{6}$$

$$C_j \ \geq\ p_j + \sum_{k\in K} r_{jk}v_{jk} \qquad \forall\, j \in J \tag{7}$$

$$C_j \ \geq\ C_i + p_j - B(3 - x_{ij} - v_{jk} - v_{ik}) \qquad \forall\, i, j \in J, i < j, k \in K \tag{8}$$

$$C_i \ \geq\ C_j + p_i - B(2 + x_{ij} - v_{jk} - v_{ik}) \qquad \forall\, i, j \in J, i < j, k \in K \tag{9}$$

$$C_{max} \ \geq\ \min_{j\in J} r_{jk} + \sum_{j\in J} p_j v_{jk} \qquad \forall\, k \in K \tag{10}$$

$$C_j \ \geq\ 0 \qquad \forall\, j \in J \tag{11}$$

$$x_{ij} \ \in\ \{0, 1\} \qquad \forall\, i, j \in J, i < j \tag{12}$$

$$w_k \ \in\ \{0, 1\} \qquad \forall\, k \in K \tag{13}$$

$$v_{jk} \ \in\ \{0, 1\} \qquad \forall\, j \in J, k \in K \tag{14}$$

Objective (1) minimizes the makespan, while Objective (2) minimizes the total travelling distance. Note that as the travelling time is positively related to the travelling distance, we thus use the release date to measure distance. Constraint (3) represents that the makespan is at least the time when the last test job is completed (i.e., the last community finishes the test), where C_j is the completion time of test job j. Constraint (4) ensures that each test job is processed at exactly one testing site. A community shall be tested at exactly one testing site located at a candidate location. The total number of testing sites to be settled is regulated by Constraint (5), while Constraint (6) guarantees that a community can take the test at a location only if a testing site is allocated to that location. The lower bound of the completion time for test job j is given by Constraint (7), which is determined by the processing time p_j together with the release time of job j. Constraints (8) and (9) are the job sequence constraints, indicating that if test job j is processed after job j at the same testing site, the completion time of test job j must be greater than or equal to the completion time of test job j plus the processing time of test job j. To obtain a tight model, we define $B = \sum_{j\in J} p_j$+$\min_{j\in Jk\in K} r_{jk}$.Constraint (10) regulates that the makespan is greater than or equal to the minimum release date of all test job j to testing site k plus the total processing time of all jobs assigned to testing site k. Note that constraints (10) are valid inequalities that strengthen the formulation and do not affect the correctness of the model. Constraint (11) ensures that the completion time of job j and the makespan are non-negative, while Constraints

(12) to (14) define the binary decision variables.

IV. Solution method

The proposed decision framework aims to complete the tests of all residents quickly and reduce the risk of cross-infection. This is done by optimally locating testing sites, allocating residents to these sites, and sequencing communities at each site. To the best of our knowledge, we are the first to introduce such an integrated decision framework under the context of a pandemic. The massive testing task faced is unprecedented, and decision-makers lack proper methods to handle such a large-scale planning problem. Our model can help in obtaining high-quality solutions if being optimally solved. However, the above model is generally challenging to be solved by using an off-the-shelf solver, e.g., CPLEX. First, we must solve a bi-objective optimization problem. Second, the studied problem is known to be NP-hard. Therefore, we must develop new methods to handle practical-sized instances.

As a benchmark, in Section IV-A, we first apply the commonly used ϵ-constraint method for bi-objective optimization problems in the literature. However, this method cannot handle large problems. Therefore, we further develop two novel solution methods to deal with practical-sized instances (i.e., IMIP, and B-LBBD), as presented in Sections IV-B and IV-C, respectively.

The ϵ-constraint method is one of the most widely used and standard methods for solving bi-objective optimization problems. Given the above-developed model LO, we can adapt the well-known ϵ-constraint method to solve the studied bi-objective parallel-testing-site ScheLoc problem, denoted as the ϵ-LO method. Its basic idea is to transform the original problem into a mono-objective problem minimizing one principal objective while formulating the other objective as a ϵ-constraint. Then the mono-objective problem is iteratively solved to obtain the Pareto front. The main drawback of the ϵ-LO method is that the transformed mono-objective problem must be solved repeatedly, and each iteration produces at most one Pareto solution. At the same time, the insertion of the ϵ-constraint into the original model may further complicate the problem. The considered bi-objective parallel-testing-site ScheLoc problem contains two hard combinatorial optimization problems, i.e., the facility location problem and parallel machine scheduling problem with makespan minimization, which are both NP-hard. Thus, the ϵ-LO method encounters difficulties in solving large-sized instances. Preliminary results show that the ϵ-LO method cannot obtain any Pareto solution within a reasonable time when the number of jobs exceeds 100.

We further develop two heuristic methods to obtain approximate Pareto fronts to tackle practical-sized problems. The first one is called the iterative MIP-based (IMIP) method, which iteratively solves an approximate MIP model (with smaller scales). New constraints are periodically added to avoid generating the same solution. The values of the two objectives are recorded in each iteration, and the process terminates when a given maximum number of iterations or time limit is reached. Then we can obtain an approximate Pareto front. The second method is called the bi-objective logic-based Benders decomposition (B-LBBD) method, which exploits the advantages of the LBBD method. The B-LBBD method decomposes the original bi-objective ScheLoc problem into a master problem (MP) and a subproblem (SP). The MP optimizes the second objective (total travel distances) and determines the machine (testing sites) location and job assignment decisions. The first objective (makespan) is obtained by solving the SP. Once the SP is solved, combinatorial Benders cuts are generated and added to the MP when the obtained makespan is greater than or equal to the current best makespan. This procedure iterates till no cuts can be added, a maximum number of iterations is performed, or the time limit is reached. The obtained values of the two objectives are recorded during the iteration process, and a set of approximate Pareto solutions is obtained. Compared with the ϵ-LO method, the B-LBBD is advantageous as the solution process begins with a relatively simpler problem (i.e., the MP). At the same time, it can obtain an approximate Pareto front by solving the model only once with cuts dynamically added to the nodes of the search tree upon finding an integer solution, instead of solving many mono-objective optimization problems as in the ϵ-LO method. Next, we present in detail the developed ϵ-LO, IMIP, and B-LBBD methods.

A. ϵ-constraint method based on model LO

Let model LO take the following form:

$$\{\min\ f_1 = \varphi(\boldsymbol{x}), f_2 = \omega(\boldsymbol{x}) | \boldsymbol{x} \in \mathcal{X}\} \tag{15}$$

where $\varphi(\boldsymbol{x})$ represents the makespan and $\omega(\boldsymbol{x})$ denotes the total travel distances. Vector $\boldsymbol{x}$ denotes the vector of all variables, and $\mathcal{X}$ is the solution space of $\boldsymbol{x}$. The ϵ-LO method first determines a principal objective and transforms the other objective into a constraint bounded by ϵ. For our problem, we set the second objective f_2 as the principal objective and transform the first objective into a constraint.

Note that we also tried to set the first objective as the principal objective and transform the second objective into constraints. However, preliminary results show that it is more efficient to optimize the second objective and treat the first objective as a constraint. The transformed mono-objective problem Q(ϵ) is presented as follows.

$$\{\min\ f_2 = \omega(\boldsymbol{x}) | \varphi(\boldsymbol{x}) \leq \epsilon, \boldsymbol{x} \in \mathcal{X}\} \tag{16}$$

For this problem, a series of mono-objective problems can be generated and solved by varying the value of ϵ within a fixed interval $[f_1^I, f_1^N]$, which is determined by defining an ideal point (f_1^I, f_2^I) and a nadir point (f_1^N, f_2^N). They are obtained by exactly solving the following mono-objective problems:

$$f_1^I = \min\{\varphi(\boldsymbol{x}) | \boldsymbol{x} \in \mathcal{X}\} \tag{17}$$

$$f_2^I = \min\{\omega(\boldsymbol{x}) | \boldsymbol{x} \in \mathcal{X}\} \tag{18}$$

$$f_1^N = \{\min\ \varphi(\boldsymbol{x}) | \omega(\boldsymbol{x}) = f_2^I, \boldsymbol{x} \in \mathcal{X}\} \tag{19}$$

$$f_2^N = \{\min\ \omega(\boldsymbol{x}) | \varphi(\boldsymbol{x}) = f_1^I, \boldsymbol{x} \in \mathcal{X}\} \tag{20}$$

Then, the mono-objective problem $Q(\epsilon)$ is solved iteratively by defining a step size Δ . Initially, ϵ takes the value of f_1^N-

Δ. Then, it is reduced by the step size Δ in each iteration. In the s^{th} iteration of $Q(\epsilon_s)$, where $s = \{1, 2, \ldots, h\}$ and $h \leq \lceil \frac{(f_1^N - f_1^I)}{\Delta} \rceil$, let the obtained value of the first objective be represented by f_1^s, then ϵ is set to be $f_1^s - \Delta$ in the $(s+1)^{th}$ iteration. Let $P_{\epsilon-LO}$ be the set of Pareto solutions and s be the iteration counter. The outline of the ϵ-LO method is shown in Algorithm 1.

Algorithm 1 ϵ-constraint (ϵ-LO) method

```
Initialize P_{ε-LO} = ∅ and Δ
Solve (17)–(20) to obtain f_1^I, f_2^I, f_1^N, and f_2^N
Set P_{ε-LO} = P_{ε-LO} ∪ {(f_1^I, f_2^N), (f_1^N, f_2^I)}
Set s = 1 and ε_s = f_1^N − Δ
while ε > f_1^I, do
    Solve Q(ε_s) to get f_2^s, and compute f_1^s
    P_{ε-LO} = P_{ε-LO} ∪ {(f_1^s, f_2^s)}
    Set s = s + 1, ε_s = f_1^{s−1} − Δ
end while
Remove the dominated points from P_{ε-LO} and return P_{ε-LO}
```

B. Iterative MIP method (IMIP)

The IMIP method obtains an approximate Pareto front by iteratively solving an approximate MIP model (model AP), which is shown as follows:

$$\min C_{max} + \sum_{j \in J} \sum_{k \in K} r_{jk} v_{jk} \tag{21}$$

$$\text{s.t.} \quad (3) - (7), (10), (13), \text{and} \, (14).$$

Comparing the above model AP with model LO, the binary sequencing variables are removed, and the objective is to minimize the sum of the two objectives. This model corresponds to the uncapacitated facility location problem, which is NP-hard. However, it can be quickly solved by existing commercial solvers, e.g., CPLEX. Note that the resulting C_{max} of the model is not the real makespan since it only partially considers the release date of each job in (7). However, it can provide a good approximation of the real makespan. Model AP is iteratively solved to obtain the total travel distances of all communities. Given the obtained v_{jk} value, the corresponding makespan value can be computed by solving a series of single machine scheduling problems with release date to minimize the makespan, denoted as $1|r_j|C_{max}$. Let s be the iteration counter and $\hat{v}_{jk}^s$ be the solution value of the assignment variable obtained by model AP in the s^{th} iteration. In the $(s+1)^{th}$ iteration, we add the following inequality into the model AP:

$$\sum_{v_{jk}|\hat{v}_{jk}^s = 0} v_{jk} + \sum_{v_{jk}|\hat{v}_{jk}^s = 1} (1 - v_{jk}) \geq 1 \tag{22}$$

The above inequality is added to force the model AP to generate a different solution in subsequent iterations. It ensures that at least one variable takes a different value from the solution obtained in the s^{th} iteration. Then, model AP is iteratively solved with the added inequality (22) at the end of each iteration. The added inequalities are stored in a pool $cons_v$. This solving process may be trapped into local optima by obtaining the same value of the location variable w_k. Thus, if the iteration counter reaches a given threshold $iter_{div}$, we force the location scheme to change by adding the following inequality to the model:

$$\sum_{w_k|\hat{w}_k^s = 0} w_k + \sum_{w_k|\hat{w}_k^s = 1} (1 - w_k) \geq 1 \tag{23}$$

where $\hat{w}_k^s$ is the value of the machine location variable obtained in the s^{th} iteration. At the same time, we remove all previously added inequalities in the pool $cons_v$ when inequality (23) is added, and set $cons_v \leftarrow \emptyset$. The procedure repeats either until the maximum number of iterations is performed or until a time limit is reached. Let P_{IMIP} be the set of Pareto solutions. The IMIP is presented in Algorithm 2.

Algorithm 2 Iterative MIP (IMIP) method

```
Initialize P_{IMIP} ← ∅, iteration counter s ← 0, s' ← 0, cons_v ← ∅,
    iter_{div}, and iter_{max}. Formulate model AP
while s ≤ iter_{max}, do
    Solve model AP, get the value of f_2^s , v̂_{jk}^s and ŵ_k^s
    Solve a series of 1|r_j|C_{max} problem to get the real makespan f_1^s
    Set P_{IMIP} = P_{IMIP} ∪ (f_1^s, f_2^s)
    Add inequality (22) to AP
    Set s ← s + 1, s' ← s' + 1, and cons_v ← cons_v ∪ (22)
    if s' ≥ iter_{div} then
        Add inequality (23) to AP and remove all inequalities in cons_v
        from AP
        Set cons_v ← ∅, s' ← 0
    end if
end while
Remove the dominated points from P_{IMIP} and return P_{IMIP}
```

C. The bi-objective Logic-based benders decomposition method

As discussed in Section 2, the LBBD method is designed initially to solve single-objective optimization problems. In this study, we innovatively develop a B-LBBD method to deal with bi-objective optimization problems. The basic idea is to decompose the original problem into an MP and an SP. The MP optimizes the main objective, while cuts represent the secondary objective. The MP is solved by the Branch-and-cut technique that operates on a single search tree, while cuts are iteratively added upon finding a feasible solution. A Pareto solution can be generated by solving the SP upon finding a feasible solution for the MP. To solve the proposed bi-objective parallel-testing-site ScheLoc problem, we decompose it into an MP and an SP. The MP optimizes the main objective (f_2, total travelling distance), while the secondary objective (f_1, makespan) can be calculated through solving the SP, which corresponds to a series of $1|r_j|C_{max}$ problems, upon finding a feasible solution for the MP. In the MP, job sequencing variables and their corresponding constraints are relaxed. The MP thus becomes much easier to deal with. Given the feasible solution to the MP (in the nodes of the Branch-and-cut search tree), the corresponding value of f_2 can be obtained, and the resulted SP is then solved with variables being fixed as the values designated by the MP, to obtain f_1 and to generate cuts. Note that the $1|r_j|C_{max}$ problem can be optimally solved using the earliest release date (ERD) rule in polynomial time.

Algorithm 3 Bi-objective logic-based Benders decomposition (B-LBBD) method

```
Initialize P_IMIP ← ∅, iteration counter s ← 0, s' ← 0, cons_v ← ∅,
  iter_div, and iter_max. Formulate model AP
Decompose the original problem into an MP and an SP
while stopping criteria not met, do
    Solve the MP, get the value of f_2, w_k, v_jk, and R_k
    Solve the SP, get the value of f_1
    Update C_max: if f_1<U, set U←f_1
    Set P_LBBD = P_LBBD ∪ (f_1, f_2)
    for every location k with w_k=1, do
        if C_max (R_k)≥U, then
            Add cuts (25) to the MP
            Set cons_v←∅, s'←0
        end if
    end for
end while
Remove the dominated points from P_LBBD and return P_LBBD
```

A Pareto solution can thus be formed with the obtained values of f_1 and f_2. Simultaneously, the generated cuts are dynamically added to the MP, which helps the algorithm to find better values of f_1 (i.e., the makespan), while, at the same time, may worsen the main objective (i.e., total travelling distance). Note that the MP is solved only once with cuts added to the nodes of the search tree upon finding a feasible solution. In the feasible solution of the MP, let R_k be the set of jobs assigned to the testing site at location k, U be the current best-known upper bound on the makespan, and $C_{max}(R_k)$ be the resulted makespan for optimally scheduling the set R_k of jobs on testing site at location k. P_{LBBD} represents the set of Pareto solutions obtained by B-LBBD. The framework of the proposed B-LBBD is shown in Algorithm 3 and depicted in Figure 2. The three key components, i.e., the MP, SP, and Benders cuts, of the B-LBBD are detailed in the following three sections.

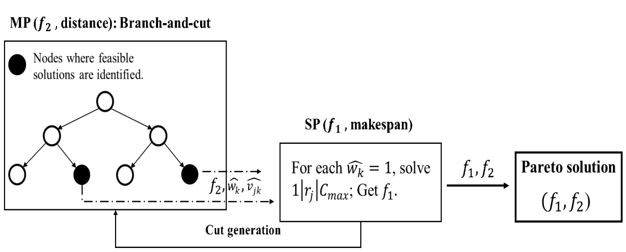

Fig. 2: The outline of the proposed novel B-LBBD method.

1) The master problem (MP): In the proposed B-LBBD method, to solve the bi-objective parallel-testing-site ScheLoc problem, we define the following MP:

$$\textbf{MP} \min\ : f_2 = \sum_{j \in J} \sum_{k \in K} r_{jk} v_{jk} \tag{24}$$

$$\text{s.t.}\ (4) - (6), \text{and to}$$

$$Cuts. \tag{25}$$

The MP corresponds to a relaxed version of LO, aiming to minimize the total travel distance. Once the MP is solved, we get the value of f_2, the machine location variable w_k, job assignment variable v_{jk} and the set R_k of jobs assigned to the testing site at location k. The MP corresponds to a p-median problem, which is known to be NP-hard [38]. Note that the MP does not involve the objective of makespan (f_1) and the sequencing variables, together with their corresponding constraints. Next, we elaborate on the resulting SP given a feasible solution to the MP.

2) The subproblem (SP): Given a feasible solution to the MP, we obtain the values of the location variable w_k, job assignment variable v_{jk}, and the set R_k of jobs assigned to machine k. The SP determines the processing sequences of jobs on each located testing site, which corresponds to a series of independent $1|r_j|C_{max}(R_k)$ problem. The problem $1|r_j|C_{max}$ can be optimally solved in polynomial time using the ERD rule. Once the SP is solved, we obtain the value of the secondary objective f_1. Here, a Pareto solution would be obtained.

3) Cut generation: The key to the B-LBBD method is to generate Benders cuts based on the information obtained from the solution of the SP. The cuts aim to reduce the makespan by restricting the set of jobs that are simultaneously assigned to a machine. This is done by performing the following steps. First, the best-known upper bound of the makespan is updated if the current $f_1 \leq U$, where U is initialized by setting $U \leftarrow +\infty$, then we set U_1. Second, we check for every location k with $R_k \neq \emptyset$. If $C_{max}(R_k) \geq U$, we add the next cut:

$$\sum_{j \in R_k} v_{jk} \leq |R_k| - 1 \tag{26}$$

The cut will exclude at least one job from the set R_k in subsequent iterations. In other words, it forbids the set R_k of jobs to be simultaneously assigned to the same machine k in subsequent iterations. The aim of adding the cut is to reduce the makespan through re-assignments of jobs. The above cut can be strengthened by (i) finding the minimal infeasible subset of R_k and (ii) the lifting technique that adds jobs to its left side. We apply the same techniques to perform such improvements as introduced in [10].

D. Comparison of the three solution methods

We have presented three solution methods for the studied bi-objective rapid testing problem, i.e., ϵ-LO, IMIP, and B-LBBD. We next summarize the cons and pros of the three methods in Table II. We next explain the novelty of the B-LBBD method. In a bi-objective optimization problem, if one of the objectives can be represented by cuts, this problem can thus be solved efficiently by our proposed B-LBBD method. This cut-type objective is regarded as the secondary objective, while the other is the main objective. The B-LBBD decomposes the original complicated bi-objective optimization problem into an MP considering only the main objective, and an SP to optimize the secondary objective which involves complex variables and constraints. The MP thus becomes easier to handle, which is solved only once by the Branch-and-cut technique operating on a single search tree. During the solving process of the MP, upon finding a feasible solution, the B-LBBD computes the corresponding value of the main objective, while solving

Methods	Advantage	disadvantage
ϵ-LO	Easy to implement, control the number of expected Pareto solutions	Dependent on solving the MILP time-consuming
IMIP	Easy to implement, the transferred model is easy to solve	solution quality can not be guaranteed
B-LBBD	Operate on a single search tree to solve a single MILP and generate Pareto solutions as the solution process evolves	May exclude some Pareto optimal solutions

TABLE II: Solution methods comparison.

a resulted SP to (i) determine the value of the secondary objective and (ii) to generate cuts that are then added to the MP. Since the MP initially excludes the secondary objective and only optimizes the main objective, the B-LBBD may identify good values for the main objective but bad ones for the secondary objective at the beginning. To improve the quality of the secondary objective, Benders cuts are generated and added to the MP, which motivates the algorithm to find a better secondary objective by sacrificing the main objective. The above process is iterated till a stopping criterion is reached. In the end, a set of Pareto solutions can be obtained. The B-LBBD approach is advantageous over the standard ϵ-constraint method. Its solution process begins with a relatively simpler problem (i.e., the MP) which is handled only once (through the Branch-and-cut technique), while it can obtain a set of Pareto solutions at the end.

V. A REAL CASE STUDY

In this section, we demonstrate the benefits of the proposed decision framework compared with the real-world practice applied in the massive COVID-19 testing program establishment through a case study based on the Universal Community Testing Programme (referred to as "the programme" hereafter) launched by the Hong Kong Special Administrative Region (HKSAR) Government in Sept 2020[2] . The official government leaflet states that the programme "aims to gauge better the COVID-19 infection situation in Hong Kong and fi¬nd asymptomatic patients as early as possible to achieve early identification, early isolation, and early treatment, and to cut the virus transmission chain in the community". Through the 14-day programme in which 1,783,000 specimens were collected, at least 42 patients were identified, which allowed the government to trace close contacts of these confirmed cases[3] . Without the programme, these patients would continue to carry out various activities in the community, leading to further community outbreaks and clusters.

6000 healthcare personnel, 4000 serving and retired civil servants, and 2000 supporting personnel participate in the programme. The government established 141 testing sites in 18 districts, including medical centres, community centres, community halls, sports halls, public schools, etc. We select Sham Shui Po District as our study target. Specifically, eight testing sites were located in this district, like the Kowloon Technical School and Cheung Sha Wan Sports Centre. To explore the location decisions, we add eight more potential locations (like municipal services buildings and sports centers). A total of 16 potential locations are available to be selected as the testing site. We consider the 17 Public Rental Housing (PRH) estates managed by the government in this district as the communities to be tested. For ease of presentation, the PRHs are denoted by 1 to 16, while the potential locations are denoted by A to P (A-H are the testing sites selected by the HKSAR government in the real world). The detailed information on the testing sites, potential locations, and communities is given in Table A-1 (please see Appendix II). The communities (labeled with a square), potential locations (labeled with a flag), and the determined testing centres (labeled with a star) are depicted in Figure FigA-1 (appendix). Besides, we collect the numbers of residents in the PRHs from Hong Kong Housing Authority[4] and the distances between the PRHs with the potential locations from Google Map, which are used to determine the processing time and release time for each community, respectively. We consider that there are five pieces of testing equipment at a testing site, while each piece of equipment could handle two testing tasks in one minute. Therefore, the processing time of each community equals the no. of residents divided by 10 (in minutes). Besides, the travelling time from a community to the testing location equals the distance multiplying 10 (in minutes).

A. Practice

As introduced, the HKSAR has selected eight testing sites in the district. According to the real-world practice, we consider that the PRHs is assigned to the determined testing site according to proximity. Besides, the communities assigned to the same testing site are scheduled based on the Earliest release date (ERD) rule. In the current practice, residents would start their journeys to the testing site at the beginning of the programme, which may cause a long waiting time at the site. The waiting time for a community equals the testing starting time minus its release date. The solution details are given in Table A-2 (Appendix II). Briefly, the makespan is 3545min, the total travelling distance is 94426km, while the total waiting time is 5.56209e+06min.

[2]https://www.communitytest.gov.hk/_doc/doc/Community_Testing_Programme_Leaflet_EN.pdf. Retrieved on 8 Sept 2021.

[3]https://www.info.gov.hk/gia/general/202009/15/P2020091500931.htm. Retrieved on 8 Sept 2021.

[4]https://www.housingauthority.gov.hk/en/common/pdf/aboutus/publications and statistics/PopulationReport.pdf. Retrieved on 9 Oct 2021.

B. Solution derived from our model

Applying our proposed bi-objective parallel-testing-site ScheLoc model and the standard ϵ-LO solution method, eight Pareto solutions can be obtained, as summarized in Table A-3 (Appendix II). The makespan required by the Pareto solutions is 3082min, while the average travelling distance is 89607km. Besides, waiting is avoided by our model as the residents will arrive at the testing sites according to the schedule.

C. Solution comparisons

Based on the discussions above, Figure 3 compares the solutions obtained by the practice and the proposed model (the Pareto front). The horizontal axis represents the makespan in the figure, while the vertical axis stands for the total travelling distance. The circle denotes the Pareto solutions obtained by the proposed model, and the square represents the solution by the practice. From Figure 3, it is obvious that the decisions derived by the practice are generally much inferior to those generated by our proposed model, showing the poor decision quality of the practice. Comparing the decisions obtained from the real-world practice and the Pareto solutions derived by our proposed model, it is found that the proposed model can achieve an average reduction of 13% in makespan, which is of great importance for the massive testing programs as quick detection is always essential for avoiding new outbreak waves. The enhanced testing efficiency facilitated by our proposed model can significantly reduce the time required to identify asymptomatic cases in the region to prevent further virus propagation, which is a key factor for the success of the public anti-pandemic measures. As estimated by Hong Kong Food and Health Bureau, the positivity rate of universal community testing is 0.0024%[5] . Assuming that all the population in the city takes the test, around 180 positive cases are thus possible to be identified. Besides, the effective reproduction number (Rt) for a local case in Hong Kong is around 3 in the fourth outbreak wave[6] . Therefore, a 13% reduction in the testing programme makespan then may help avoid 180*3*13%=70 further infections at the time of testing, which would lead to an exponential growth of infected people if not identified[7] . Besides, by applying our model, the residents to be tested can travel less to the assigned testing sites (a mean of 5.1%), which is helpful in reducing the risk of virus spread during the trip and traveller inconvenience. Moreover, as the proposed model can provide accurate information like the starting time for the test job of each community, residents can better plan their trip to avoid unnecessary waiting at the testing sites. Accordingly, the developed decision framework can achieve zero waiting, compared with the 5.56209e+06min of waiting derived by the practice. The dilemma of long waiting in queues, the difficulty in social distance control, and the associated complaints can thus be solved by applying our paradigm.

[5]https://www.legco.gov.hk/research-publications/english/2021rb01-challenges-and-economic-impacts-arising-from-coronavirus-disease-2019-20201214-e.pdf. Retrieved on 8 July 2021.

[6]https://covid19.sph.hku.hk/dashboard.Retrieved on 8 July 2021.

[7]Note that estimating the number of infections is not the focus of this study. We thus provide rough estimation of the infections reduced.

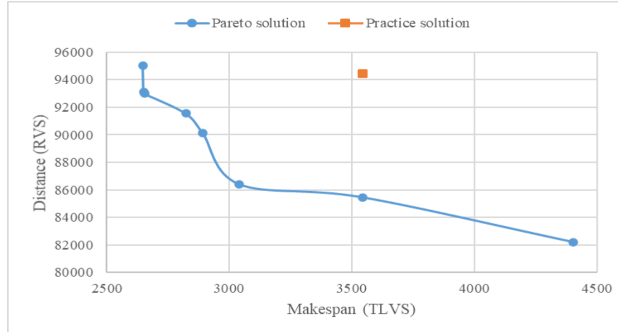

Fig. 3: Solution comparisons of the proposed model and the random practice.

1) The master problem (MP): Furthermore, it is interesting to identify that our proposed decision scheme can better balance the workload burdened by each testing site due to the systematic location, assignment, and scheduling decisions. Note that the workload is evaluated by the length of working time for the testing site, as determined by the communities assigned. From the perspective of statistics, as the overall workload for all communities is fixed (which is equal to the sum of the processing time for all communities to be tested) and the number of testing sites is also determined, the average workload for each site is the same for the two decision frameworks (i.e., the practice and our proposed model). However, the variance of workload is much higher for the practice. Specifically, the standard deviation of working time for the practice is 1230, while that derived from our model is only 835, achieving a 32% reduction.

In summary, from the case study, it is revealed that the proposed decision framework can significantly reduce makespan, shorten the overall travelling distance, eliminate waiting at testing sites, as well as balance the workload at different sites.

VI. Computational Experiments

This section illustrates the superior performances of the proposed bi-objective parallel-testing-site ScheLoc decision framework in establishing massive testing programs under the COVID-19 pandemic, and the advantages of the developed solution methods through computational experiments. Experiments were conducted on a personal computer with Windows 10 operating system and Intel (R) CORE i7-8700 CPU 3.2 GHz at 16 GB RAM. The proposed models and algorithms are coded in the C++ programming language by using Microsoft Visual Studio 2015 linked with CPLEX (12.10.0.). First, the characteristics of the tested instances are described. Then, the improvement in decision-making achieved by our proposed model over the current applied practice is demonstrated through a case study. Last, the three solution methods applied in this study are evaluated by applying five performance indicators. Note that the upper limit for the computation time is set as 3600s, while the maximum time allowed for the mono-objective problem in each iteration of the ϵ-LO method is 300s. In the implementation of IMIP, the $iter_{max}$and $iter_{div}$ are set to 2000 and 200, respectively. In ϵ-LO, the step size Δ is set to 1. All MILP models are solved by CPLEX with a relative gap tolerance of 0.1%.

A. Instance generation

To thoroughly evaluate the performances of the proposed bi-objective parallel-testing-site ScheLoc model and solution methods, we randomly generate 20 sets of instances with 5 for each, totaling 100 instances. Two dimensions characterize these instances. The first dimension is the number of communities to be tested (that is, the number of test jobs). The second dimension is for the number of candidate locations for testing sites. Specifically, we consider 20, 40, 60,…, 400 communities (a total of 20 sets of instances), while for each set of instances, there are 4, 8, 12, 16, and 20 candidate testing site locations. The number of testing sites to be established (the number of machines to be settled) is equal to half of the candidate testing site locations. For example, for the first set of instances, residents from 20 communities must take the test. For Instance 1 in this set, the authority plans to settle two testing sites at four candidate locations, while for Instance 2 in the same set, we consider eight candidate locations to place four testing sites. The processing time for each community to finish the test is randomly generated from a Uniform distribution U∼ [1,100][8] . The time unit is minute. The coordinates of the communities (x_i, y_i) and candidate locations (x_k, y_k), are generated from U∼[1,200]. The travelling time between communities and the candidate locations is equal to the Euclidean distance (km) between the two places × 10. The unit is set as minute, used to obtain the release dates as model inputs. The distance unit is set as 100 meters.

B. Performance evaluation of the proposed solution methods

Here, we evaluate the solution methods constructed in this work. First of all, the computational results obtained from small-size instances are compared to analyze the performances of the three methods. Then, the IMIP and B-LBBD methods are further investigated to illustrate the efficacy and merits of the B-LBBD method for practical application based on large-size instances where the ϵ-LO method becomes unsolvable.

We use A_F, B_F and C_F to denote the Pareto solution sets obtained by the ϵ-LO method, IMIP, and B-LBBD, respectively. To be specific, the approximate Pareto front A_F, B_F and C_F are compared with a reference set R_F which consists of all Pareto solutions obtained by the three solution methods to generate managerial insights, i.e., R_F=$A_F \cup B_F \cup C_F$ and all dominated solutions are removed. To facilitate performance evaluations, we employ five widely applied indicators [17], [40], namely the cardinality (i.e., $|A_F|$, $|B_F|$ and $|C_F|$), hypervolume ratio (H), average e-dominance (D), maximum spread (MS), and computation time (T). The evaluation mechanisms of the five indicators are briefly explained in appendix III.

[8]We use the uniformly distribution here is for simplicity, which also follows other mainstream literature in transportation and operations research [39]. Replacing it by other distributions (such as the normal distribution) does not affect the main qualitative conclusion.

1) Solution methods comparisons based on small-size instances: Based on small-size instances, this part demonstrates the comparisons of the three solution methods developed in this study (the adapted ϵ-LO method, the proposed IMIP, and the proposed B-LBBD method). The values of the five indicators introduced above obtained by the three methods for Instances 1 to 25 are given in Table A4 (Appendix). Note that the unit of computation time (T) is second. Besides, we summarize the evaluation outcomes of the three methods for all instances in Table A4 (Appendix), where the number “1” represents the best, “2” stands for the medium, and “3” means the worse. From the third row of Table III, it is seen that none of the three methods is definitely superior or inferior to the others. For instance, on average, the IMIP performs better than the ϵ-LO by generating (i) a higher cardinality (i.e., more non-dominated Pareto solutions), thus providing higher flexibility for decision-makers, (ii) a higher hypervolume ratio, and (iii) a better average e-dominance, while worse from the perspectives of maximum spread and running time. The B-LBBD is advantageous over the ϵ-LO in terms of all the five evaluation indicators.

However, when comparing the IMIP and the B-LBBD, it is interesting to identify that the IMIP can provide slightly more non-dominated Pareto solutions (i.e., 1.3%) even though the IMIP is disadvantageous for the remaining four indicators.

Although it is difficult to tell exactly which method is the best for the small-scale instances, it is obvious that the B-LBBD shows prominent merits over the other two methods from the aspects of the computation time, maximum spread, hypervolume ratio, and average e-dominance. Averagely, the B-LBBD only uses 1031 seconds to identify solutions, while the ϵ-LO consumes twice (i.e., 2303 seconds). Surprisingly, even though the ϵ-LO fails to tackle large-scale problems, it requires less computation time than the IMIP for these small-scale instances. The IMIP spends as long as 2639 seconds, ranking the highest among the three methods. Note that for some instances, ϵ-LO stops before the total time limit is reached because it fails to obtain any feasible solution when solving a mono-objective problem in 300 seconds. Accordingly, considering that reaction speed is a critical component in decision-making during the pandemic, the B-LBBD demonstrates excellent potential for practical application. The incapability to solve large problems dramatically limits the applicability of the standard ϵ-LO method. Regarding the maximum spread, the B-LBBD, on average, realizes a 59.5% and a remarkable 217.9% improvement over the ϵ-LO and the IMIP, respectively. In terms of the hypervolume ratio, the B-LBBD also performs better than the ϵ-LO and the IMIP (with an average increase of 12.43% and 5.98%, respectively). Besides, the B-LBBD lowers the average e-dominance by 2.13% and 0.47% compared with the ϵ-LO and the IMIP, respectively.

In summary, the experiments based on small-size instances demonstrate the impressive performances of the B-LBBD for most evaluation indicators. The computational time, which is crucial for the efficiency and effectiveness of massive testing programs in resisting the COVID-19 outbreak, is validated.

Instance	ϵ-LO					IMIP					B-LBBD				
	$\|A_F\|$	H	D	MS	T	$\|B_F\|$	H	D	MS	T	$\|C_F\|$	H	D	MS	T
1-25	3	3	3	2	2	1	2	2	3	3	2	1	1	1	1
26-50	unsolved					2	2	2	2	2	1	1	1	1	1
51-75						2	2	1	2	2	1	1	2	1	1
76-100						2	2	1	2	2	1	1	2	1	1
26-100						2	2	1	2	2	1	1	2	1	1

TABLE III: Summary of the solution methods evaluations.

Moreover, the IMIP is characterized by its success in providing the most non-dominated Pareto solutions.

2) Demonstration of the merits of B-LBBD based on large-size instances: When the problem scale increases (i.e., the number of communities to be tested exceeds 100, Instances 26 to 100), the adapted ϵ-LO method cannot obtain any feasible Pareto solution within the given time limit. Accordingly, in this part, we further explore the performances of the proposed B-LBBD and IMIP for large-size instances. For the following comparisons, the reference set $R_F = B_F \cup C_F$ and all dominated solutions are removed. For ease of presentation, the values of indicators obtained by the two methods based on Instances 26 to 50, 51 to 75, and 76 to 100 are listed in three separate tables (i.e., Table A-5 (Appendix), Table A-6 (Appendix), and Table A-7 (Appendix)), respectively. The evaluation outcomes of the two methods for these instances are summarized in the fourth to sixth rows of Table 4, while the last row presents the overall evaluation for all the large-size instances.

Obviously, the B-LBBD demonstrates overwhelming merits over the IMIP for these large instances. The only exception appears in the indicator of average e-dominance for Instances 51 to 75 and 76 to 100, where the IMIP realizes a mean of 0.15% and 0.5% lower e-dominance than the B-LBBD, respectively. From the perspective of all large instances (Instances 26 to 100), the IMIP lowers the average e-dominance by 0.07% compared with the B-LBBD, achieving a slight improvement for this indicator. When looking at the other indicators for these large instances, overall speaking, compared with the IMIP, the B-LBBD is shown to be prominently advantageous in (i) providing much higher decision flexibility for authorities (by generating an average of 73.38% more non-dominated Pareto solutions), (ii) a much larger maximum spread (on average 482.54%), and (iii) a higher hypervolume ratio (with a mean of 15.38%), while consuming much less computation time (on average 39.12%; the IMIP reaches the upper time limit for the majority of the large instances), which greatly enhances its potential and value for daily use in building fast-response public health systems under the outbreak of COVID-19.

To sum up, both the proposed B-LBBD and IMIP can deal with large-scale problems within the given time limit, showing the capability for practical applications. In particular, the B-LBBD is distinctly superior to the IMIP, mainly by generating higher decision flexibility and reducing computation times. At the same time, the IMIP shows a marginal advantage regarding the average e-dominance for these large instances.

VII. Managerial implications

In this section, we generalize the managerial implications derived from our study from two perspectives. First, we discuss the improvements achieved by our proposed novel bi-objective parallel-testing-site ScheLoc decision framework for the massive COVID-19 testing program establishment compared with the real-world practice in a random manner. Then, the applicability of our proposed model facilitated by the developed solution algorithms for practical utilization is discussed, further verifying the significance and great value of this study for fighting COVID-19.

A. Decision-making improvement (for establishing massive COVID-19 testing programs)

The massive testing program establishment decision framework developed in this study provides scientific and systematic suggestions for the testing site location decisions, community assignment decisions, and community scheduling decisions for governmental decision-makers. Using the developed framework, the shortened makespan (with an average of 13%) and travelling distances of the tested residents (by a mean of 5.1%) bring great benefits to society. Imagine that a large population has to travel to the testing sites using public transportation tools (e.g., buses, subways). The possibility of an infected resident meeting healthy people or a healthy resident meeting infected people during the trip thus increases significantly, which exacerbates the spread of the virus. By scientifically assigning communities to testing sites, our proposed decision framework can reduce the risk of virus-spread-on-the-way, which simultaneously enhances the performance of the massive testing programs and protects the public. Besides, the zero-waiting realized by our proposed model also brings enormous benefits. In the current real-world practice, residents always need to wait a long time in the queue before getting tested. Social distance control is difficult in a crowd, increasing the chance of cross-infection. Residents are also dissatisfied with the poor governmental arrangements, which is harmful to authorities in building their credibility. With the reduction in travelling distances and waiting time, the overall time required to finish all the test jobs (i.e., makespan) can be significantly reduced. Responsive actions taken by governments can alleviate public panic and help maintain the stability of the region, which in turn guarantees the success of the subsequent anti-virus policies launched by authorities. Moreover, as shown from the case study, the proposed model can better balance the workload assigned to the staff at different testing sites (the standard deviation of working time is reduced by 32%). In the current practice, the testing personnel at different sites are required to work under different pressure levels. Those allocated with longer working time can quickly encounter dissatisfaction and generate complaints about the authority, impairing working efficiency and accuracy. More importantly, a heavy workload will lead to the fatigue problem, weakening the immune system of the staff, and increasing their risks of being infected. Therefore, it is believed that the proposed scientific decision framework, i.e., the solution to the bi-objective parallel-machine ScheLoc problem, can greatly help improve governments' decision-making.

B. Practical applicability achieved by the developed solution methods

Although the ϵ-constraint method has been widely applied to solve bi-objective problems, it cannot efficiently deal with large-scale real-world problems, as shown from our experiments. In reality, the scale of COVID-19 testing is usually huge (e.g., hundreds of communities requiring tests). Therefore, the ϵ-constraint method is unsuitable for practical use. During a global pandemic, to effectively halt the wide and fast spread of the COVID-19 virus, it is crucial to develop efficient solution algorithms that can obtain satisfactory solutions within a reasonable computation time to provide useful suggestions for authority decision-makers. This is especially important for establishing massive COVID-19 testing programs, as the reaction speed is a primary determinant for the performance of the related governmental policies to avoid new outbreak waves when new positive cases are detected. Accordingly, the novel solution methods proposed in this study (i.e., the iterative MIP method and the bi-objective Logic-based benders decomposition method) are demonstrated to handle large-scaled instances. Moreover, between the two options, we would recommend the bi-objective Logic-based benders decomposition method more for authorities considering its superior performance in various solution evaluation indicators. It provides much higher decision flexibility and consumes much less computation time, which is very important for real-world large-scale COVID-19 fighting practice requiring quick responses. As shown from the experiments, the B-LBBD provides (i) much higher decision flexibility by generating an average of 73.38% more non-dominated Pareto solutions, (ii) much larger maximum spread (on average 482.54%), and (iii) a higher hypervolume ratio (with a mean of 15.38%), with much less computation time (on average 39.12%).

VIII. Concluding Remarks and Future Studies

The recent global pandemic has triggered an increasing demand for COVID-19 tests to diagnose symptomatic and asymptomatic infections. Massive testing programs (in which many residents from various communities are required/suggested to take the test) play a crucial role in assessing the risk of releasing lockdown measures, avoiding new outbreak waves, and facilitating the resumption of normal societal and economic activities.

However, the current practice in establishing massive testing programs suffers from the random decisions about testing site locations, community assignments, and community schedules, leading to unreasonable travelling distances for the tested persons, the unbalanced workload at different testing sites, and long queues, which inevitably causes a long completion time and increases the risk of virus spread and traveller inconvenience. To address these challenges, this study aims to utilize the tool of optimization [41]–[44], to construct a massive testing program establishment decision framework regarding the location of testing sites, as well as the assignment and scheduling of the tested communities. Through analyzing the problem characteristics, it is found that the logic of the related decision-making behind is similar to the parallel-machine ScheLoc problem, following which we mathematically formulate the massive testing program establishment problem as a bi-objective parallel-testing-site ScheLoc problem which simultaneously minimizes the makespan and the total travelling distance. To solve the problem, we first propose a bi-objective MILP. We then adapt the widely used ϵ-constraint method and develop two novel approaches. The first is to iteratively solve approximate MIP models of the original problem with smaller scales (i.e., the IMIP method), while the second innovatively extends the typical logic-based Benders decomposition approach, which is originally developed for single-objective optimization problems, to handle bi-objective problems (i.e., the B-LBBD method), thus theoretically contributing to the bi-objective optimization literature.

A case study based on Hong Kong and computational experiments can identify several significant improvements achieved by our proposed decision framework compared with the current random practice. First, the proposed model can significantly shorten the makespan (with an average of 13%), roughly translating to the avoidance of 70 further infections if all Hong Kong population take the test. Second, the traveling distance of the tested person is reduced by a mean of 5.1%. Third, unnecessary waiting and long queues at testing sites can be eliminated. Fourth, the proposed decision framework can better balance the workload burdened by each testing site (e.g., the standard deviation is reduced by 32%). Therefore, it is believed that the public health system could be greatly strengthened by applying our proposed massive testing program establishment decision framework, which helps alleviate the adverse impacts of the epidemic on public health and the economy.

Moreover, from computational experiments, the two newly developed solution methods, especially the B-LBBD, are shown to outperform the adapted ϵ-constraint method from various aspects including the ability to handle large-scale problems, providing higher decision flexibility through generating more Pareto solutions, and reducing computation time, which further enhances the applicability of the proposed model in daily use to fight against COVID-19, and to establish efficient and responsive public health systems during the pandemic. At last, it is worth noting that our proposed model and solution methods are not only applicable to COVID-19. It can be extended to other public emergency scenarios (e.g., other epidemic diseases and natural disasters like an earthquake) where the location of facilities and the assignment and scheduling of personnel or materials are required.

For future research, the current study focuses on the mass testing for COVID-19. We have made an assumption that the testing time of a community is deterministic and dependent on its population. However, since a community is composed of many residents, the testing time of a community may be uncertain. Therefore, further research may focus on a stochastic counterpart of the studied problem. In this case, we could formulate the problem using stochastic or robust models. In addition, the current study aims at makespan and travel distance minimization. However, the cost of locating testing sites, allocating personnel, and distributing testing toolkits are also worthy of study. Therefore, the current problem can be extended to consider different practical operational costs. From the engineering management perspective, the process management [45] for COVID-19 mass testing is also critical. More studies should be conducted to examine the related standard operating procedures and processes. Recently, more and more researchers propose that vaccination is a key to the success of the COVID-19 battle. However, vaccine supply chains are usually decentralized during a pandemic and suffer from supply and demand mismatches due to supply uncertainties [46], [47]. Therefore, coordinating the COVID-19 vaccine supply chain to satisfy better the mass vaccination demand based on the ScheLoc decisions of mass vaccination programs (i.e., locations of vaccination sites, assignments of communities, and schedules of communities) is valuable to investigate.

References

[1] S. Bag, S. Gupta, T.-M. Choi, and A. Kumar, "Roles of innovation leadership on using big data analytics to establish resilient healthcare supply chains to combat the COVID-19 pandemic: a multimethodological study," *IEEE Transactions on Engineering Management*, 2021, doi: https://doi.org/10.1109/TEM.2021.3101590.

[2] D. Ivanov, "Digital supply chain management and technology to enhance resilience by building and using end-to-end visibility during the COVID-19 pandemic," *IEEE Transactions on Engineering Management*, 2021, doi: https://doi.org/10.1109/TEM.2021.3095193.

[3] D. Ozdemir, M. Sharma, A. Dhir, and T. Daim, "Supply chain resilience during the COVID-19 pandemic," *Technology in Society*, vol. 68, p. 101847, 2022.

[4] K. S. Al-Omoush, D. Palacios-Marqué, S. Ribeiro-Navarrete, and K. Ulrich, "The innovative impact of trust in social government: An analysis during COVID-19 crisis," *IEEE Transactions on Engineering Management*, 2021, doi: https://doi.org/10.1109/TEM.2021.3130222.

[5] A. Arunachalam, V. Ravi, V. Acharya, and T. D. Pham, "Toward data-model-agnostic autonomous machine-generated data labeling and annotation platform: COVID-19 autoannotation use case," *IEEE Transactions on Engineering Management*, 2021, doi: https://doi.org/10.1109/TEM.2021.3076603.

[6] J. P. Devarajan, A. Manimuthu, and V. R. Sreedharan, "Healthcare operations and black swan event for COVID-19 pandemic: a predictive analytics," *IEEE Transactions on Engineering Management*, 2021, doi: https://doi.org/10.1109/TEM.2021.3076603.

[7] I. Vanany, M. H. Ali, K. H. Tan, A. Kumar, and N. Siswanto, "A supply chain resilience capability framework and process for mitigating the COVID-19 pandemic disruption," *IEEE Transactions on Engineering Management*, 2021, doi: https://doi.org/10.1109/TEM.2021.3116068.

[8] Centers for Disease Control and Prevention, "Overview of testing for sars-cov-2 (COVID-19)," 2020, https://www.cdc.gov/coronavirus/2019-ncov/hcp/testing-overview.html.

[9] C. Heßler and K. Deghdak, "Discrete parallel machine makespan scheloc problem," *Journal of Combinatorial Optimization*, vol. 34, no. 4, pp. 1159–1186, 2017.

[10] Y. Li, J.-F. Côté, L. Coelho, and P. Wu, "Novel formulations and logic-based benders decomposition for the integrated parallel machine scheduling and location problem," *INFORMS Journal on Computing*, vol. 34, pp. 1048–1069, 06 2021.

[11] H. Hennes, *Integration of scheduling and location models*. PhD Thesis, University of Kaiserslautern, Germany, 2005.

[12] D. Elvikis, H. W. Hamacher, and M. T. Kalsch, "Simultaneous scheduling and location (scheloc): the planar scheloc makespan problem," *Journal of Scheduling*, vol. 12, no. 4, pp. 361–374, 2009.

[13] M. T. Kalsch and Z. Drezner, "Solving scheduling and location problems in the plane simultaneously," *Computers & Operations Research*, vol. 37, no. 2, pp. 256–264, 2010.

[14] S. Wang, R. Wu, F. Chu, J. Yu, and X. Liu, "An improved formulation and efficient heuristics for the discrete parallel-machine makespan scheloc problem," *Computers & Industrial Engineering*, vol. 140, p. 106238, 2020.

[15] R. Kramer and A. Kramer, "An exact framework for the discrete parallel machine scheduling location problem," *Computers & Operations Research*, vol. 132, p. 105318, 2021.

[16] M. Liu, X. Liu, E. Zhang, F. Chu, and C. Chu, "Scenario-based heuristic to two-stage stochastic program for the parallel machine scheloc problem," *International Journal of Production Research*, vol. 57, no. 6, pp. 1706–1723, 2019.

[17] Y. Li, F. Chu, C. Feng, C. Chu, and M. Zhou, "Integrated production inventory routing planning for intelligent food logistics systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 3, pp. 867–878, 2018.

[18] X. Wen, H.-L. Ma, S.-H. Chung, and W. A. Khan, "Robust airline crew scheduling with flight flying time variability," *Transportation Research Part E: Logistics and Transportation Review*, vol. 144, p. 102132, 2020.

[19] X. Wen, X. Sun, Y. Sun, and X. Yue, "Airline crew scheduling: Models and algorithms," *Transportation Research Part E: Logistics and Transportation Review*, vol. 149, p. 102304, 2021.

[20] W. A. Khan, H.-L. Ma, S.-H. Chung, and X. Wen, "Hierarchical integrated machine learning model for predicting flight departure delays and duration in series," *Transportation Research Part C: Emerging Technologies*, vol. 129, p. 103225, 2021.

[21] J. Benders, "Partitioning procedure for solving mixed-integer programming problems," *Numerische Mathematik*, vol. 4, 1962.

[22] J. N. Hooker and G. Ottosson, "Logic-based benders decomposition," *Mathematical Programming*, vol. 96, no. 1, pp. 33–60, 2003.

[23] T. T. Tran, A. Araujo, and J. C. Beck, "Decomposition methods for the parallel machine scheduling problem with setups," *INFORMS Journal on Computing*, vol. 28, no. 1, pp. 83–95, 2016.

[24] V. Roshanaei and B. Naderi, "Solving integrated operating room planning and scheduling: Logic-based benders decomposition versus branch-price-and-cut," *European Journal of Operational Research*, vol. 293, no. 1, pp. 65–78, 2021.

[25] M. Karatas, L. Erişkin, and E. Bozkaya, "Transportation and location planning during epidemics/pandemics: Emerging problems and solution approaches," *IEEE Transactions on Intelligent Transportation Systems*, 2022.

[26] X. Qian and S. V. Ukkusuri, "Connecting urban transportation systems with the spread of infectious diseases: A trans-seir modeling approach," *Transportation Research Part B: Methodological*, vol. 145, pp. 185–211, 2021.

[27] D. Wolfinger, M. Gansterer, K. F. Doerner, and N. Popper, "A large neighbourhood search metaheuristic for the contagious disease testing problem," *European Journal of Operational Research*, 2021, doi: https://doi.org/10.1016/j.ejor.2021.10.028.

[28] C. Zhang, Y. Li, J. Cao, and X. Wen, "On the mass COVID-19 vaccination scheduling problem," *Computers & Operations Research*, vol. 141, p. 105704, 2022.

[29] E. H. Kaplan, "Om forum—COVID-19 scratch models to support local decisions," *Manufacturing & Service Operations Management*, vol. 22, no. 4, pp. 645–655, 2020.

[30] P. Guo, Q. Li, F. Liu, and Y. Wang, "Battling against COVID-19 via mass testing and mass vaccination," p. Working Paper, 02 2021.

[31] Y.-C. Chen, P.-E. Lu, C.-S. Chang, and T.-H. Liu, "A time-dependent sir model for COVID-19 with undetectable infected persons," *IEEE*

Transactions on Network Science and Engineering, vol. 7, no. 4, pp. 3279–3294, 2020.

[32] K. Nikolopoulos, S. Punia, A. Schäfers, C. Tsinopoulos, and C. Vasilakis, "Forecasting and planning during a pandemic: COVID-19 growth rates, supply chain disruptions, and governmental decisions," *European Journal of Operational Research*, vol. 290, no. 1, pp. 99–115, 2021.

[33] K. Arifoğlu and C. S. Tang, "A two-sided incentive program for coordinating the influenza vaccine supply chain," *Manufacturing & Service Operations Management*, vol. 24, no. 1, pp. 235–255, 2022.

[34] S. Singh, R. Kumar, R. Panchal, and M. K. Tiwari, "Impact of COVID-19 on logistics systems and disruptions in food supply chain," *International Journal of Production Research*, vol. 59, no. 7, pp. 1993–2008, 2021.

[35] A. Nagurney, "Supply chain game theory network modeling under labor constraints: Applications to the COVID-19 pandemic," *European Journal of Operational Research*, vol. 293, no. 3, pp. 880–891, 2021.

[36] T.-M. Choi, "Innovative "bring-service-near-your-home" operations under corona-virus (COVID-19/sars-cov-2) outbreak: can logistics become the messiah?" *Transportation Research Part E: Logistics and Transportation Review*, vol. 140, p. 101961, 2020.

[37] K. Govindan, H. Mina, and B. Alavi, "A decision support system for demand management in healthcare supply chains considering the epidemic outbreaks: A case study of coronavirus disease 2019 (COVID-19)," *Transportation Research Part E: Logistics and Transportation Review*, vol. 138, no. C, 2020.

[38] O. Kariv and S. L. Hakimi, "An algorithmic approach to network location problems. i: The p-centers," *SIAM Journal on Applied Mathematics*, vol. 37, no. 3, pp. 513–538, 1979.

[39] Y. Li, F. Chu, C. Chu, and Z. Zhu, "An efficient three-level heuristic for the large-scaled multi-product production routing problem with outsourcing," *European Journal of Operational Research*, vol. 272, no. 3, pp. 914–927, 2019.

[40] B. E. Oruc and B. Y. Kara, "Post-disaster assessment routing problem," *Transportation Research Part B: Methodological*, vol. 116, pp. 76–102, 2018.

[41] N. Damij and T. Damij, "An approach to optimizing kanban board workflow and shortening the project management plan," *IEEE Transactions on Engineering Management*, 2021.

[42] C. Dong, C. Chen, X. Shi, and C. T. Ng, "Operations strategy for supply chain finance with asset-backed securitization: Centralization and blockchain adoption," *International Journal of Production Economics*, vol. 241, p. 108261, 2021.

[43] N. Liu, J. Lin, S. Guo, and X. Shi, "Fashion platform operations in the sharing economy with digital technologies: recent development and real case studies," *Annals of Operations Research*, pp. 1–21, 2022.

[44] X. Shi, H.-L. Chan, and C. Dong, "Impacts of competition between buying firms on corporate social responsibility efforts: Does competition do more harm than good?" *Transportation Research Part E: Logistics and Transportation Review*, vol. 140, p. 101985, 2020.

[45] N. Damij, T. Damij, *et al.*, "Process management," *A Multi-disciplinary Guide to*, 2014.

[46] E. Adida, D. Dey, and H. Mamani, "Operational issues and network effects in vaccine markets," *European Journal of Operational Research*, vol. 231, no. 2, pp. 414–427, 2013.

[47] K. Arifoğlu, S. Deo, and S. M. Iravani, "Consumption externality and yield uncertainty in the influenza vaccine supply chain: Interventions in demand and supply sides," *Management Science*, vol. 58, no. 6, pp. 1072–1091, 2012.

Yantong Li received the B.S. degree in traffic and transportation from Beijing Jiaotong University, Beijing, China, in 2011, the M.S. degree in transportation planning and management from Military Transportation University, Tianjin, China, in 2013, and the Ph.D. degree in Automation from University of Paris Saclay, Evry, France, in 2019.

Dr. Li is currently an Associate Professor with Dalian Maritime University, Dalian, China. His research interests include planning and scheduling in production and logistics systems, integrated optimization in supply chain, smart logistics, and mathematical programming-based methods. His research papers are published or accepted by some international journals, including INFORMS Journal on Computing, European Journal of Operational Research, International Journal of Production Research, International Journal of Production Economics, IEEE Transactions on Automation Science and Engineering, IEEE Transactions on Intelligent Transportation Systems, and Computers & Operations Research.

Xin Wen is currently a Research Assistant Professor in Department of Industrial and Systems Engineering at The Hong Kong Polytechnic University. She has published in journals such as IEEE Transactions on Systems, Man, and Cybernetics - Systems, International Journal of Production Research, International Journal of Production Economics, and Transportation Research – Part E. Her current research interest is on transportation and logistics engineering.

Tsan-Ming Choi received the Ph.D. degree in operations and supply chain management from Department of Systems Engineering and Engineering Management, The Chinese University of Hong Kong in 2002. He has published extensively in leading journals in the fields of operations management, engineering management, logistics and supply chain management. He is currently serving the profession as the Co-Editor-in-Chief of Transportation Research Part E: Logistics and Transportation Review, a Department Editor of IEEE Transactions on Engineering Management, a Senior Editor of Production and Operations Management, and Decision Support Systems, and an Associate Editor of Decision Sciences, and IEEE Transactions on Systems, Man and Cybernetics - Systems. He is also a member of the Engineering Panel of Research Grants Council of Hong Kong. He has taught at The Chinese University of Hong Kong, The Hong Kong Polytechnic University, National Taiwan University, and University of Liverpool Management School.

Sai-Ho Chung is an Associate Professor and Associate Head in Department of Industrial and Systems Engineering, The Hong Kong Polytechnic University. His research interests include logistics and supply chain management, supply chain collaboration, production scheduling, distributed scheduling, distribution network, etc. He has been the principal investigator of about ten research projects and authored and coauthored more than 90 international journal papers. He has guest edited a number of special issues for various journals, e.g., Transportation Research Part E, Industrial Management & Data Systems.