Stochastic Last-mile Delivery with Crowd-shipping and Mobile Depots

Kianoush Mousavi
Department of Civil and Mineral Engineering, University of Toronto, Toronto, Ontario M5S 1A4, Canada,
kianoush.mousavichashmi@mail.utoronto.ca

Merve Bodur
Department of Mechanical and Industrial Engineering, University of Toronto, Toronto, Ontario M5S 3G8, Canada,
bodur@mie.utoronto.ca

Matthew J. Roorda
Department of Civil and Mineral Engineering, University of Toronto, Toronto, Ontario M5S 1A4, Canada,
matt.roorda@utoronto.ca

This paper proposes a two-tier last-mile delivery model that optimally selects mobile depot locations in advance of full information about the availability of crowd-shippers, and then transfers packages to crowd-shippers for the final shipment to the customers. Uncertainty in crowd-shipper availability is incorporated by modeling the problem as a two-stage stochastic integer program. Enhanced decomposition solution algorithms including branch-and-cut and cut-and-project frameworks are developed. A risk-averse approach is compared against a risk-neutral approach by assessing conditional-value-at-risk. A detailed computational study based on the City of Toronto is conducted. The deterministic version of the model outperforms a capacitated vehicle routing problem on average by 20%. For the stochastic model, decomposition algorithms usually discover near-optimal solutions within two hours for instances up to a size of 30 mobile depot locations, 40 customers and 120 crowd-shippers. The cut-and-project approach outperforms the branch-and-cut approach by up to 85% in the risk-averse setting in certain instances. The stochastic model provides solutions that are 3.35% to 6.08% better than the deterministic model, and the improvements are magnified with increased uncertainty in crowd-shipper availability. A risk-averse approach leads the operator to send more mobile depots or postpone customer deliveries, to reduce the risk of high penalties for non-delivery.

Key words: Last-mile delivery, crowd-shipping, mobile depot, two-stage stochastic integer programming, decomposition, value of stochastic solution, conditional-value-at-risk

1. Introduction

Innovative and cost-effective last-mile delivery solutions are needed in response to the boom in e-commerce and the growing demand for fast home deliveries. The global last-mile delivery market value is expected to grow from 1.99 Billion USD to 7.69 Billion USD by 2026 driven by an upsurge of e-commerce and expectation for faster delivery times by customers (Insight Partners 2019). In Canada, 48 percent of e-shoppers reside in urban areas resulting in high density of demand for last mile delivery (Lee, Kim, and Wiginton 2019). This has led to an increase of business-to-consumer related commercial vehicle movements in already congested urban road networks. Truck movements in urban areas are notorious for slowing down traffic, parking illegally and causing air pollution and noise. For instance, in 2012, commercial vehicles received 66 percent of all parking tickets in downtown Toronto, Canada (Rosenfield et al. 2016). Reducing urban truck traffic would diminish some of the negative externalities in dense urban areas.

Mobile Depots. In response to these issues, urban consolidation centres have been established in some cities to increase consolidation of goods and decrease truck travel on urban road networks (Van Rooijen and Quak 2010). However, large public subsidy and high costs of establishing and operating urban consolidation centres make them unsustainable in some cases (Lin, Chen, and Kawamura 2016, Van Duin et al. 2016). Mobile depots have been proposed as a more cost-effective and flexible alternative. Verlinde et al. (2014) defined a mobile depot as a trailer that acts as a loading dock, warehousing facility, and a small office. They considered mobile depots, loaded with customer packages, being sent on a daily basis to urban areas and parked in central parking locations; and then riders of electric tricycles picking up packages from the mobile depots for final delivery. More recently, Marujo et al. (2018) conducted a case study in Rio de Janeiro, Brazil, and showed that the use of mobile depots with cargo tricycles can significantly reduce greenhouse gas emissions, at the same time yielding a small cost saving compared to traditional truck delivery.

Crowd-shipping. The sharing economy has provided a new solution for last-mile delivery, known as crowd-shipping. Crowd-shippers are commuters who are willing to make a delivery by deviating from their original route in exchange for a small compensation. Crowd-shipping can provide several economic, environmental, and social benefits. Since crowd-shippers are cheaper than regular drivers, courier companies can reduce their operational cost and fleet size and improve their service time by integrating crowd-shipping into their delivery operations. Courier companies can offer delivery tasks to available crowd-shippers through an online platform or a mobile app. In response, crowd-shippers offer time and cargo carrying capacity. Use of crowd-shipping can replace some delivery trucks on the roads, yielding reductions in congestion, illegal parking, air pollution, and noise. Several companies, such as Walmart, DHL, and Amazon Flex, have exploited crowd-shipping for last-mile delivery. Walmart, for example, recruits in-store customers to deliver online orders (Barr and Wohl 2013). Shipper Bee, a Canadian crowd-shipping company, uses crowd-shipping for middle-mile deliveries, by exploiting inter-city commuters to transfer packages between intermediate pick-up and drop-off points.

Proposed Business Model. In this paper, inspired from the abovementioned solutions, we present a new variant of the last mile delivery business model integrating crowd-shipping and mobile depots. Our proposed model aims to leverage crowd-shippers' commute, willingness to support more sustainable shipping operation and their motivation to earn money. In the proposed model, instead of relying solely on trucks for delivering packages to the customer from the main depot (as in the traditional capacitated vehicle routing problem), we consider a two-tier delivery operation using trucks as mobile depots for the first tier and crowd-shippers for the second tier. Mobile depots (i.e., trucks) bring customer packages from the main depot to selected stopping locations (e.g., parking lots). While mobile depots spend a predetermined time (e.g., several hours) at the selected stopping locations, crowd-shippers pick up parcels from them to deliver to customers during their commute. As the first step of analyzing the potential benefits of this new business model, in this study, we allow a single stopping location/package assignment for each mobile depot/crowd-shipper per time frame (such as morning and evening). We leave the incorporation of routing of mobile depots/crowd-shippers among multiple stopping/customer locations for future research. Section 3 provides the detailed description of the proposed business model.

Mobile depots vs. fixed depots. In terms of mathematical modeling of the proposed business model, implementing mobile depots compared to fixed depots only impose slight differences. However, implementing mobile depots has noteworthy differences in contrast to fixed depots as a business model. Establishing fixed depots requires a significant initial investment compared to mobile depots, especially in urban areas. In some settings, using fixed depots as storage facility for customer packages might justify the high initial investment. In contrast, mobile depots are temporary depots, with no on-site storage in mobile depots stopping locations; therefore mobile depots must remain on stopping locations for passing customer packages to crowd-shippers. Furthermore, in the case of mobile depots, there exists the flexibility of selecting stopping locations (e.g., parking lots across a city) among many available options. This operational flexibility of mobile depots enables them to adapt more effectively to crowd-shippers' daily commuting pattern and customers' location in daily operation planning. In comparison to fixed depots, this adaptation may yield extra cost-savings for the company and may increase crowd-shippers' willingness to participate in last-mile delivery.

Importance of Uncertainty. We study the problem of determining mobile depot stopping locations, distributing customer packages to these depots, and assigning available crowd-shippers to packages for final delivery. However, the availability and willingness of commuters to participate in crowd-shipping are uncertain. Since crowd-shippers are not necessarily committed to undertake the delivery task, some may not show up. Accounting for this uncertainty in the decision-making process can have potential benefits for the success and cost-saving of the daily operations. By assuming that the company in a crowd-shipping operation has information of potential crowd-shippers' commuting patterns and show-up probability (e.g., through historical data collected via a mobile app), we incorporate uncertainty in crowd-shipper availability into our mathematical model. For this purpose, we develop a two-stage stochastic integer programming model which considers availability of crowd-shippers as the main source of uncertainty. The probability of crowd-shipper availability is considered as an exogenous uncertainty (i.e., not dependent on decision variables in the model). In reality, this probability may depend on other factors such as the crowd-shippers' compensation rate and the required detour by crowd-shippers to serving a customer. Some other exogenous uncertainties can be easily incorporated into the two-stage stochastic programming framework, at the expense of significantly increased computational effort to derive good-quality solutions. We note that demand uncertainty is not relevant to the proposed business model since we assume that the company only considers the orders received a day before or overnight. This allows the company to load the mobile depots to be sent to their stopping locations on the morning of the crowd-shipping operation.

Contributions. The main contributions of this paper are as follows. We introduce a new variant of the business model combining crowd-shipping and mobile depots, and define the Stochastic Mobile Depot and Crowd-shipping Problem (SMDCP). We formulate the problem as a two-stage stochastic integer program which decides on mobile depot stopping locations and assignment of customers to selected mobile depots in advance of full information about the set of available crowd-shippers. At this step, only their availability *probability* is known, i.e., their actual availability on the day of operation is revealed later in the operational stage. To our best knowledge, this is the first attempt in quantifying value of stochasticity in a crowd-sourced delivery operation. We develop enhanced decomposition solution algorithms, implemented in branch-and-cut and cut-and-project frameworks (Bodur et al. 2017). Also, we extend our model into a risk-averse setting, and apply similar decomposition algorithms for its solution. To the best of our knowledge, this constitutes the first application of the cut-and-project framework in a risk-averse setting. We conduct an extensive computational study on instances generated for the City of Toronto, Canada. This study illustrates (i) the benefits of the proposed business model over a traditional delivery model (i.e., a capacitated vehicle routing problem), (ii) the computational efficiency of the enhanced decomposition solution approach, (iii) the value of incorporating uncertainty in the availability of crowd-shippers as well as a risk measure into the model.

The remainder of this paper is organized as follows. In Section 2, we review the relevant literature and highlight the novelty of our problem. In Section 3 and Section 4, we present an elaborated description of the SMDCP and its mathematical formulation, respectively. In Section 5, we present our proposed methodology. In Section 6, we explain how our model and methodology can be extended to a risk-averse setting. In Section 7, we describe our case study, present our computational experiment results, and provide managerial insights. Finally, we provide conclusions and future research directions in Section 8.

2. Literature Review

Crowd-shipping has been recently introduced to the body of literature on last-mile delivery. The earliest crowd-shipping last-mile delivery model is introduced by Archetti, Savelsbergh, and Speranza (2016). They extend the capacitated vehicle routing problem by incorporating in-store customers as crowd-shippers. In-store customers undertake a part of the last-mile delivery operation in exchange for a small compensation proportional to their deviation from their original route. Some of the limitations in that study are addressed by Macrina et al. (2017), who incorporate time-windows and allow multiple deliveries by crowd-shippers. However, both of these studies are static and do not incorporate stochasticity in crowd-shipper availability into their models.

Incorporating uncertainty into last-mile delivery related models, especially in the vehicle routing problem, has received significant attention in the literature (see the reviews by Gendreau, Laporte, and Séguin (1996) and Oyola, Arntzen, and Woodruff (2018)), where the most commonly considered uncertainties have been in customer demand, travel times, service times and presence of customers. However, few studies consider uncertainty in crowd-shipping models.

Arslan et al. (2019) introduce a peer-to-peer model that matches delivery tasks to crowd-shippers (referred to as ad-hoc drivers) in real-time. Their model also considers backup vehicles (i.e., delivery trucks) to handle deliveries that are not viable or cost-efficient by crowd-shippers. They develop a rolling-horizon framework that re-optimizes the decisions of their model each time new information about crowd-shippers and delivery tasks is revealed. They also introduce an exact algorithm and a heuristic to design routes for crowd-shippers.

Dayarian and Savelsbergh (2017) propose a dynamic same-day delivery crowd-shipping model where in-store customers are considered as possible crowd-shippers along with a company-owned fleet. They propose a myopic policy, which does not consider any information about the future, as well as a sample-scenario based rolling-horizon framework that considers stochastic information about the arrival of in-store customers and online orders. They show that incorporation of stochastic future information can increase service quality drastically with only a modest increase in total delivery operation cost.

Gdowska, Viana, and Pedroso (2018) incorporate the probability of willingness of crowd-shippers to accept or reject assigned delivery tasks into the model of Archetti, Savelsbergh, and Speranza (2016), and propose a bi-level heuristic to minimize the expected total delivery cost. In contrast to their one-stage stochastic modeling approach, we incorporate uncertainty into a two-stage stochastic model due to the two-tier nature of the delivery system.

Dahle, Andersson, and Christiansen (2017) develop a two-stage stochastic vehicle routing model with dynamically appearing crowd-shippers. In the first stage, they decide on routing of company vehicles before the appearance of any crowd-shippers. In the second stage, they assign customers' packages to crowd-shippers or company vehicles, or decide to postpone some deliveries and incur a penalty. They show that, compared to a deterministic model with re-optimization, the stochastic model can yield cost-savings. However, their stochastic analysis does not consider the show-up probability for crowd-shippers. Instead they enumerate all possible scenarios for availability of crowd-shippers and consider equal probability for all scenarios, which may not reflect individuals' availability probability correctly. Furthermore, the proposed approach is practical only when a few crowd-shippers are considered (three crowd-shippers are considered in the paper). Otherwise, scenario generation and scenario evaluation techniques are required to obtain valid solutions.

The mobile depots have advantage over fixed facilities in their flexibility to move for adapting to temporal and spatial change of demand. For this purpose, Halper and Raghavan (2011) introduce the mobile facility routing problem. (Note that mobile facilities are referred to mobile depots for last-mile delivery applications.) They model the routing of mobile facilities while demand in the service region locations changes over time. They aim to maximize the total serviced demand by considering the dependency of demand on the arrival and departure of mobile facilities in each stop. They develop three heuristics algorithms that produce high-quality solution based on their computational results, especially for instances with a significant temporal and spatial demand change. Similarly, our study can benefit from the concept of mobile depots in crowd-shipping since customer locations and crowd-shippers commuting patterns change day by day. In contrast to the study of Halper and Raghavan (2011), our study considers a new business model, i.e., a two-tier delivery operation with mobile depots and crowd-shippers, and proposes a two-stage stochastic model without considering routing of mobile depots.

Although transshipment nodes for truck delivery are well-studied (e.g., see Dellaert et al. (2019) and references therein), the literature on the role of *transshipment nodes in crowd-shipping* is still sparse. Kafle, Zou, and Lin (2017) has made the first attempt to introduce transshipment nodes (referred to as relay points) in a crowd-shipping operation. They propose a new crowd-shipping system incorporating truck deliveries to cyclist and pedestrian crowd-shippers willing to receive packages from trucks at transshipment nodes and deliver them to customers. These local crowd-shippers submit bids and the truck carrier decides on bids, transshipment nodes, truck routes and schedules. The generated bids in their study only involve routes of the crowd-shippers that start and end at a specific transshipment node. Crowd-shippers are assumed to conform with the time window of customers and arrival time of trucks if their bids are selected by the truck carrier.

Raviv and Tenzer (2018) study transshipment nodes as automatic drop-off and pick-up points. Crowd-shippers transfer packages from one transshipment node to another. They develop a stochastic dynamic program which provides an optimal routing of packages in the network, under simplifying assumptions such as ignoring capacity constraints at transshipment nodes. Macrina et al.

(2020) extend the study of Kafle, Zou, and Lin (2017) by modeling individual crowd-shippers and incorporating their time-windows. Crowd-shippers are allowed to pick up customer packages from the central depot as well as transshipment locations. They develop a highly efficient variable neighborhood search heuristic solution algorithm.

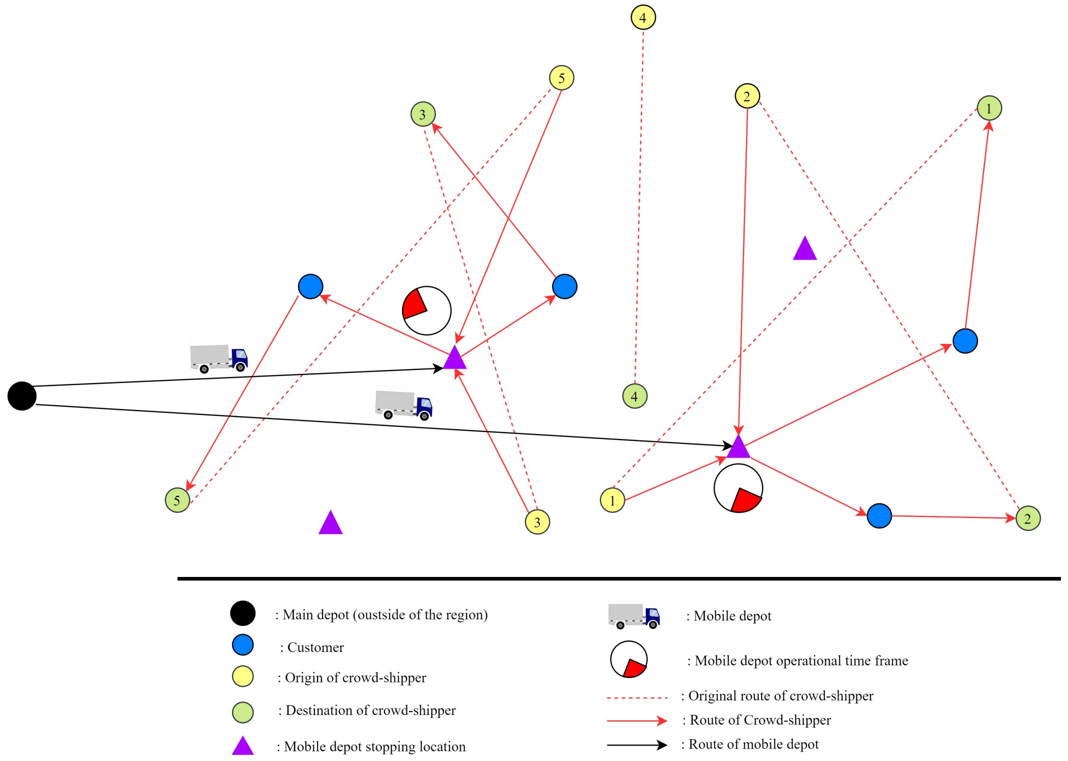
3. Problem Description

We consider a two-tier last-mile delivery system, as illustrated in Figure 1, in which the first tier consists of movement of mobile depots from the main depot to stopping locations, while the second tier consists of delivery of packages from mobile depots to customers by crowd-shippers. In other words, the trucks loaded with customer packages are dispatched from a main depot to selected urban locations where they spend time for crowd-shippers to pick up packages and deliver them to customers. We assume that the availability of crowd-shippers is uncertain, and the assignment of customer packages to mobile depots together with the selection of mobile depot stopping locations must be made before this uncertainty is revealed. For instance, consider mobile depots that are loaded with customer packages the night before and sent to the selected locations in the early morning, after which crowd-shippers' availability is revealed and package-to-crowd-shipper assignment is made. Crowd-shippers indicate (e.g., via a mobile app) their intention to make a delivery the day before to help the company in the first-tier planning, but have a chance not to show up due to last-minute changes in their schedule or commute. In the case of no show, the courier company may assign the package to a crowd-shipper with a higher compensation or incur a penalty for not serving the customer.

The SMDCP is defined as follows. Given a set of potential locations for a capacitated mobile depot, a set of customer locations, and a set of potential crowd-shippers with their commute and availability (i.e., the probability of showing up) information, the SMDCP decides on mobile depot locations, their operational time frames (e.g., morning and evening), and the assignment of customers to selected mobile depots, in order to minimize the *expected* total cost of operating mobile depots, compensating crowd-shippers and postponing customer package deliveries.

In contrast to the most closely-related studies from the literature, namely by Kafle, Zou, and Lin (2017) and Macrina et al. (2020), which primarily model truck routes, we focus on incorporating the uncertainty of crowd-shipper availability and their spatial commuting pattern. Due to this added level of complexity, we do not consider routing of mobile depots (i.e., trucks), and we do not allow mobile depots to make a direct delivery to customers. In other words, in our model, mobile depots have only one stop in their assigned location where they pass customer packages to available crowd-shippers. However, since our modeling framework is flexible to incorporate such extra features, we leave their exploration for future research. Lastly, we note that the first-tier of

Figure 1 Last-mile Delivery Operation with Crowd-shipping and Mobile Depots.


4
5
3
2
1
5
4
1
3
2
: Main depot (outside of the region)
: Customer
: Origin of crowd-shipper
: Destination of crowd-shipper
: Mobile depot stopping location
: Mobile depot
: Mobile depot operational time frame
: Original route of crowd-shipper
: Route of Crowd-shipper
: Route of mobile depot

our delivery operation is highly aggregated. Since we require trucks as mobile depots, which are assumed to be less sustainable for making delivery to customers in urban areas, ignoring routing of such mobile depots is deemed appropriate.

4. Mathematical Model

We formulate the SMDCP as a two-stage stochastic integer program. We assume that crowd-shipper availability is uncertain and modeled as a random vector $\xi = (\xi_1, \ldots, \xi_K)$ where ξ_k represents the availability of crowd-shipper $k \in K$. The first-stage decisions consist of mobile depot stopping location selection, customer package selection for potential delivery via crowd-shipping, and package assignment to mobile depots, represented by binary variables $\{z_i\}_{i\in I}$, $\{y_j\}_{j\in J}$ and $\{w_{ij}\}_{i\in I, j\in J}$, respectively. These decisions are all deterministic, i.e., they need to be made before the crowd-shipper availability is observed. The second-stage decisions are made after observing the crowd-shipper availability information. The second-stage problem assigns packages to available crowd-shippers or sends the packages back to the main depot, using the binary variables $\{p_{ijk}\}_{i\in I, j\in J, k\in K}$ and $\{v_j\}_{j\in J}$, respectively. The notation used in the model is provided in Table 1.

The SMDCP formulation is as follows:

$$\nu^* := \min \; F(z,y) + \mathbb{E}_\xi[Q(y,w,\xi)] \tag{1a}$$

Table 1 Notation Used in the Stochastic Programming Model.

Sets:	
I	Set of mobile depot stopping locations (indexed by i)
J	Set of customers (indexed by j)
K	Set of crowd-shippers (indexed by k)
Parameters:	
c_i^z	Cost of sending a mobile depot to location i
c_{ijk}^p	Cost of serving customer j with crowd-shipper k through location i
c_j^y	Cost of postponing customer j's delivery in the first stage
c_j^v	Cost of not serving customer j's delivery in the second stage
$t(i)$	0 if the operation window of location i is in the morning, 1 if it is in the evening
$t(k)$	0 if crowd-shipper k is available in the morning, 1 if available in the evening
C	Capacity of (homogeneous) mobile depots
Random variables:	
ξ_k	1 if crowd-shipper k is available, 0 otherwise
Decision variables:	
z_i	1 if a mobile depot is sent to location i, 0 otherwise
y_j	1 if customer j's delivery is postponed in the first stage, 0 otherwise
w_{ij}	1 if customer j's package is sent by mobile depot to location i, 0 otherwise
p_{ijk}	1 if customer j is served through mobile depot at location i by crowd-shipper k, 0 otherwise
v_j	1 if customer j is not served in the second stage, 0 otherwise

$$\text{s.t. } w_{ij} \leq z_i \qquad \forall i \in I, \forall j \in J \tag{1b}$$

$$\sum_{i \in I} w_{ij} + y_j = 1 \qquad \forall j \in J \tag{1c}$$

$$\sum_{j \in J} w_{ij} \leq C \qquad \forall i \in I \tag{1d}$$

$$z \in \{0,1\}^{|I|}, y \in \{0,1\}^{|J|}, w \in \{0,1\}^{|I||J|}, \tag{1e}$$

where the first-stage cost function is defined as

$$F(z,y) = \sum_{i \in I} c_i^z z_i + \sum_{j \in J} c_j^y y_j,$$

while the second-stage cost function is

$$Q(y,w,\xi) = \min \sum_{i \in I} \sum_{j \in J} \sum_{k \in K} c_{ijk}^p p_{ijk} + \sum_{j \in J} c_j^v v_j \tag{2a}$$

$$\text{s.t. } \sum_{i \in I} \sum_{j \in J} p_{ijk} \leq \xi_k \qquad \forall k \in K \tag{2b}$$

$$\sum_{i \in I} \sum_{k \in K} p_{ijk} + v_j = 1 - y_j \qquad \forall j \in J \tag{2c}$$

$$\sum_{k \in K} p_{ijk} \leq w_{ij} \qquad \forall i \in I, \forall j \in J \tag{2d}$$

$$p_{ijk} = 0 \qquad \forall j \in J,\ \forall (i,k) \in I \times K \text{ s.t. } t(i) \neq t(k) \tag{2e}$$

$$p \in \{0,1\}^{|I||J||K|}, v \in \{0,1\}^{|J|}. \tag{2f}$$

The objective function (1a) minimizes the sum of (i) mobile depot operational cost, which includes truck, gas, and parking costs, (ii) penalty cost for postponing a customer's delivery, and (iii) the expected second-stage cost associated with the crowd-shipping operation, that includes compensation to crowd-shippers and penalty for not serving a customer (which is larger than the postponing penalty) as provided in (2a). Constraints (1b) ensure that customer packages are only assigned to dispatched mobile depots. Constraints (1c) either postpone a customer delivery to a future operational day or assign the package to a mobile depot. Constraints (1d) are the capacity constraints indicating the maximum number of customers that can be assigned to a mobile depot, where we assume that the company has a fleet of homogeneous mobile depots. We note that routing of mobile depots between stopping locations can be incorporated into the first-stage problem, however, we choose to keep the model simpler to accommodate larger instance cases in our stochasticity and sensitivity analysis that are based on optimal solutions.

The so-called recourse problem (2) aims to minimize the compensation to crowd-shippers and penalty for sending undelivered customer packages back to the main depot. Constraints (2b) ensure that a customer can only be assigned to an available crowd-shipper. Constraints (2c) assign customer packages shipped through mobile depots to crowd-shippers, also recording those that are not delivered due to lack of crowd-shipper availability/suitability. Constraints (2d) make sure that a package can only be delivered by a crowd-shipper through a mobile depot if it is assigned to that depot in the first stage.

We also introduce morning and evening time frames to implicitly incorporate crowd-shippers' temporal commuting patterns and mobile depots' operation times. Crowd-shippers can only pick up customers' packages from mobile depots that have the same time frame since there is no option of drop-off for mobile depots in stopping locations, which is enforced by constraints (2e). We note that if a crowd-shipper is available for both morning and evening time frames, then we represent that person as two different crowd-shippers. Customers' time frames are not considered, i.e., we assume that crowd-shippers can drop off packages at customers' locations at any time in the morning or evening; however, this assumption can be easily relaxed by considering implicit time frames for customers as well.

5. Solution Methodology

Stochastic programs are notoriously difficult to solve to optimality, thus are most commonly solved by means of approximation. The most commonly used approach is the Sample Average Approximation (SAA), which we also use, as explained in Section 5.1. This approach transforms the stochastic program into a large-scale mixed-integer program. In Section 5.2, we describe our methodology to solve the obtained deterministic problem via a decomposition algorithm for which we outline some enhancements in Section 5.3.

5.1. Sample Average Approximation

The (theoretical and computational) challenge in solving the two-stage stochastic programming formulation (1) stems from the expectation term in the objective function (1a). Since we have K random variables, exact evaluation of this expectation requires a K-dimensional integration as we need to consider all possible realizations of the random vector ξ. In order to overcome this computational difficulty, the SAA method considers a finite number of realizations for the random vector, each of which is called a *scenario*, and approximates the expected second-stage cost with the average cost of the scenarios. This results in a deterministic problem referred to as the *SAA problem*.

Let $\{\xi^s\}_{s\in S}$ be a sample of scenarios, which we call a *solution sample*. We use Monte Carlo sampling to generate (independent and identically distributed) scenarios, assigning equal probability, i.e., $1/|S|$, to each one. Replacing the expectation term in (1a) with the sample average, we obtain the SAA problem for the SMDCP as follows:

$$\hat{\nu}(S) := \min_{z,y,w} F(z,y) + \frac{1}{|S|}\sum_{s\in S} Q(y,w,\xi^s) \tag{3a}$$

$$\text{s.t. } (1b) - (1e). \tag{3b}$$

In what follows, we explain how to choose a reasonable sample size to be used in building the SAA problem to obtain near-optimal solutions to the original problem, by deriving statistically valid upper and lower bounds on the optimal objective value of the original stochastic program, and how to reformulate the SAA problem as a mixed-integer program. We refer the reader to the book by Shapiro, Dentcheva, and Ruszczyński (2014) and references therein for more details.

5.1.1. Obtaining upper bounds. Let $(\hat{z}(S), \hat{y}(S), \hat{w}(S))$ be an optimal solution of the SAA problem, (3). Because it satisfies all the first-stage constraints, (1b)-(1e), it is a feasible solution for the original SMDCP problem (1), i.e., can be implemented in practice. However, its true cost

$$U(S) := F(\hat{z}(S), \hat{y}(S)) + \mathbb{E}_\xi[Q(\hat{y}(S), \hat{w}(S), \xi)]$$

which constitutes an upper bound on the SMDCP optimal value (i.e., $\nu^* \leq U(S)$), is not necessarily correctly reflected by the SAA optimal value $\hat{\nu}(S)$. Again, due to the complexity in evaluating high-dimensional integrals, the true cost is estimated via sampling as

$$U^{mean}(S) := F(\hat{z}(S), \hat{y}(S)) + \frac{1}{|S^{eval}|}\sum_{s\in S^{eval}} Q(\hat{y}(S), \hat{w}(S), \xi^s) \tag{4}$$

where S^{eval} is a large *evaluation sample* of equally likely scenarios independently generated from the probability distribution of the random vector ξ, (i.e., $|S| << |S^{eval}|$). More specifically, a confidence interval (CI) with a level of $100*(1-\alpha)$ can be constructed on the upper bound statistic $U(S)$ as

$$U^{mean}(S) \pm \texttt{z-score}_{\alpha/2} U^{std}(S) \tag{5}$$

where $U^{std}(S)$ represents the standard deviation of the evaluation sample (individual) scenario costs: $\{F(\hat{z}(S),\hat{y}(S))+Q(\hat{y}(S),\hat{w}(S),\xi^s)\}_{s\in S^{eval}}$.

5.1.2. Obtaining lower bounds. As mentioned above, given a solution sample of N scenarios, S_N, the optimal value $\hat{\nu}(S_N)$ of the SAA problem (3) is not meaningful. However, its expected value constitutes an unbiased estimator on the true optimal value of the SMDCP model, i.e., $\mathbb{E}[\hat{\nu}(S_N)]\leq\nu^*$, which can be estimated by solving multiple SAA problems.

Let $S_N^1,\ldots,S_N^M$ be independent samples of scenarios, each of cardinality N. Solving the corresponding SAA problem for each solution sample, we obtain M objective function values $\{\hat{\nu}(S_N^1),\ldots,\hat{\nu}(S_N^M)\}$. Denoting their mean and standard deviation by $L_{M,N}^{mean}$ and $L_{M,N}^{std}$, respectively, we can build a CI with a level of $100*(1-\alpha)$ on the lower bound statistic $\mathbb{E}[\hat{\nu}(S_N)]$ as

$$L_{M,N}^{mean}\pm\texttt{t-score}_{\alpha/2,M-1}L_{M,N}^{std}. \tag{6}$$

5.1.3. Choosing the number of scenarios. The recommended use of the SAA method is to vary the sample size, $N=|S|$; compute the corresponding lower and upper bound CIs, respectively given in (5) and (6); choose a sufficiently large sample size to achieve a reasonable optimality gap also considering the computational effort required to solve the SAA problem; and then conduct problem-specific analyses based on the solutions obtained from the SAA problem built with the appropriate number of scenarios.

For a given sample size N, the *worst-case optimality gap* (%) is calculated as the relative gap between the lowest endpoint of the lower bound interval and the highest endpoint of the upper bound interval:

$$100\times\frac{\left|\left(U^{mean}(\tilde{S})+\texttt{z-score}_{\alpha/2}U^{std}(\tilde{S})\right)-\left(L_{M,N}^{mean}-\texttt{t-score}_{\alpha/2,M-1}L_{M,N}^{std}\right)\right|}{U^{mean}(\tilde{S})+\texttt{z-score}_{\alpha/2}U^{std}(\tilde{S})}$$

where the solution sample used for the upper bound estimation, $\tilde{S}$, can be selected from the solution samples used in the lower bound estimation, namely $S_N^1,\ldots,S_N^M$. In our experiments, as detailed in Section 7.4, we use a mid-size evaluation sample, S^{pick}, (i.e., $|S^{pick}|>N$) to compute $U^{mean}(S_N^1),\ldots,U^{mean}(S_N^M)$ via (4), and pick $\tilde{S}$ as the one sample with the lowest upper bound mean estimate, that is, $\tilde{S}=S_N^\ell$ with $\ell\in\underset{i=1,\ldots,M}{\arg\min}\,U^{mean}(S_N^i)$.

5.1.4. Reformulating the SAA problem. The SAA problem, (3), can be reformulated as a mixed-integer program by introducing copies of the second-stage decision variables and constraints, one set of copies per scenario (by adding scenario subscripts and "for all scenarios" statements, respectively) as follows:

$$\min\ \sum_{i\in I}c_i^z z_i+\sum_{j\in J}c_j^y y_j+\frac{1}{|S|}\sum_{s\in S}\left(\sum_{i\in I}\sum_{j\in J}\sum_{k\in K}c_{ijk}^p p_{ijks}+\sum_{j\in J}c_j^v v_{js}\right) \tag{7a}$$

$$\text{s.t. } w_{ij} \le z_i \qquad \forall i \in I, \forall j \in J \tag{7b}$$

$$\sum_{i\in I} w_{ij} + y_j = 1 \qquad \forall j \in J \tag{7c}$$

$$\sum_{j\in J} w_{ij} \le C \qquad \forall i \in I \tag{7d}$$

$$\sum_{i\in I}\sum_{j\in J} p_{ijks} \le \xi_k^s \qquad \forall k \in K, \forall s \in S \tag{7e}$$

$$\sum_{i\in I}\sum_{k\in K} p_{ijks} + v_{js} = 1 - y_j \qquad \forall j \in J, \forall s \in S \tag{7f}$$

$$\sum_{k\in K} p_{ijks} \le w_{ij} \qquad \forall i \in I,\ \forall j \in J, \forall s \in S \tag{7g}$$

$$p_{ijks} = 0 \qquad \forall j \in J, \forall s \in S,\ \forall (i,k) \in I \times K \text{ s.t. } t(i) \ne t(k) \tag{7h}$$

$$z \in \{0,1\}^{|I|}, y \in \{0,1\}^{|J|}, w \in \{0,1\}^{|I||J|} \tag{7i}$$

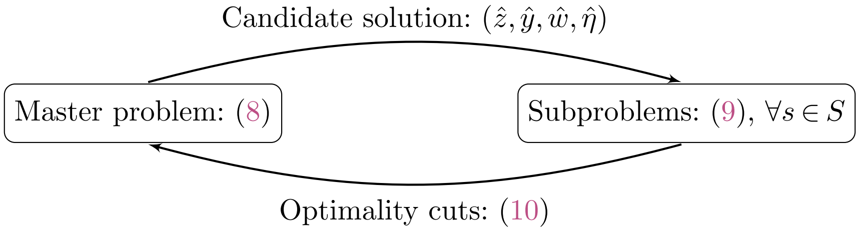
$$p \in \{0,1\}^{|I||J||K||S|}, v \in \{0,1\}^{|J||S|}. \tag{7j}$$

This formulation is known as the *extensive form* (EF) of the SAA problem. When the number of scenarios, $|S|$, is large, this model usually takes a prohibitively long time to solve, thus is solved by means of decomposition. In the next section, we present a decomposition algorithm to solve (7).

5.2. Decomposition Framework

The EF given in (7) has a special block structure making it very amenable for decomposition. Once the first-stage variables z, y and w are fixed in (7), then the remaining problem decomposes by scenario. Based on this observation, the EF can be decomposed into a *master problem* and a set of *subproblems*, one per scenario, and solved via a cutting plane algorithm as illustrated in Figure 2.

Figure 2 Decomposition and Cutting Plane Algorithm Scheme.


Candidate solution: $(\hat{z}, \hat{y}, \hat{w}, \hat{\eta})$
Master problem: (8)
Subproblems: (9), $\forall s \in S$
Optimality cuts: (10)

5.2.1. Master problem. Introducing continuous variables $\{\eta_s\}_{s\in S}$ to represent the second-stage cost under each scenario, we obtain the master problem (MP) as follows:

$$\min_{z,y,w,\eta} \sum_{i\in I} c_i^z z_i + \sum_{j\in J} c_j^y y_j + \frac{1}{|S|}\sum_{s\in S} \eta_s \tag{8a}$$

$$\text{s.t. } (7b), (7c), (7d), (7i) \tag{8b}$$

$$[\text{Cuts linking } \eta, y \text{ and } w \text{ variables}] \tag{8c}$$

$$\eta \in \mathbb{R}_+^{|S|} \tag{8d}$$

Cuts in (8c) are generated on-the-fly to provide a gradually improving approximation of the set

$$\bigcap_{s\in S}\left\{(\eta_s,y,w)\in\mathbb{R}\times\{0,1\}^{|J|}\times\{0,1\}^{|I||J|}:\eta_s\geq Q(y,w,\xi^s)\right\},$$

and eventually to ensure that $\eta_s=Q(y,w,\xi^s)$ for all $s\in S$.

5.2.2. Subproblems. The MP, which is a relaxation of the EF, provides a candidate solution, $(\hat{z},\hat{y},\hat{w},\hat{\eta})$, to the subproblems to be checked for feasibility and optimality. If one of the checks fail, then a new cut is generated for the MP to eliminate the candidate solution. More specifically, we have one subproblem (SP) for each scenario $s\in S$, whose right-hand-side vector is modified according to the scenario and the master candidate solution:

$$\begin{aligned}
Q(\hat{y},\hat{w},\xi^s)=\min\ & \sum_{i\in I}\sum_{j\in J}\sum_{k\in K}c^p_{ijk}p_{ijk}+\sum_{j\in J}c^v_j v_j && && \text{(9a)}\\
\text{s.t.}\ & \sum_{i\in I}\sum_{j\in J}p_{ijk}\leq\xi^s_k && \forall k\in K && \text{(9b)}\\
& \sum_{i\in I}\sum_{k\in K}p_{ijk}+v_j=1-\hat{y}_j && \forall j\in J && \text{(9c)}\\
& \sum_{k\in K}p_{ijk}\leq\hat{w}_{ij} && \forall i\in I,\forall j\in J && \text{(9d)}\\
& p_{ijk}=0 && \forall j\in J,\ \forall(i,k)\in I\times K \text{ s.t. } t(i)\neq t(k) && \text{(9e)}\\
& p\in\{0,1\}^{|I||J||K|},v\in\{0,1\}^{|J|}. && && \text{(9f)}
\end{aligned}$$

5.2.3. Optimality cuts. Note that our two-stage stochastic program (1) for the SMDCP has relatively complete recourse, since for any given first-stage feasible solution, the second-stage problem is feasible thanks to the v variables corresponding to not serving customers. In other words, our SPs are feasible for any given MP candidate solution. As a result, we only need to generate optimality cuts.

Optimality cuts are generated and added to the MP when there is a mismatch between the guess of the MP for the second-stage cost of a scenario, $\hat{\eta}_s$, and the true cost of the scenario, $Q(\hat{y},\hat{w},\xi^s)$. As the second-stage is a binary program and all the first-stage variables that are linked to the second-stage problem are all binary, one option is to use the so-called integer L-shape cuts (Laporte and Louveaux 1993) to fix such a mismatch. However, these cuts are usually very weak, thus their use in the cutting plane algorithm (known as the integer L-shaped method) has a slow convergence to an optimal solution.

On the other hand, we observe that the binary restrictions in the SPs can be relaxed in our case, thus we can resort to Benders cuts, and in turn employ the Benders decomposition algorithm (Benders 1962). In Proposition 1, we prove that the constraint coefficient matrix of (9) is totally unimodular (TU), which combined with the facts that (i) the right-hand-side vector is integral, and (ii) (9) is feasible, implies that the polyhedron $\{(p,v) \in \mathbb{R}_+^{|I||J||K|} \times \mathbb{R}_+^{|J|} : (9b)-(9e)\}$ is integral (Wolsey and Nemhauser 1999, Corollary 2.4). The proof of the proposition is provided in the Appendix A.

Proposition 1. *Let $\mathbb{A}$ be the constraint coefficient matrix of* (9). *Then, $\mathbb{A}$ is TU.*

Let γ, π and β be dual variables associated with constraints (9b), (9c) and (9d), respectively. Then, for a scenario with the mismatched cost, $\hat{\eta}_s < Q(\hat{y}, \hat{w}, \xi^s)$, we generate the Benders cut

$$\eta_s \geq \sum_{k\in K} \hat{\gamma}_k \xi_k^s + \sum_{j\in J} \hat{\pi}_j (1-y_j) + \sum_{i\in I}\sum_{j\in J} \hat{\beta}_{ij} w_{ij} \tag{10}$$

for the MP, where $\hat{\gamma}$, $\hat{\pi}$ and $\hat{\beta}$ are obtained at an optimal dual solution of the linear programming relaxation of (9).

5.3. Enhancements

We introduce many algorithmic enhancements to improve the performance of our Benders decomposition algorithm. All of these ideas are inspired by Bodur and Luedtke (2017) and Bodur et al. (2017). As such, in this section we only summarize the strategies we implemented, and refer the reader to the aforementioned studies for more details.

5.3.1. Branch-and-cut (B&C) implementation. As our MP is a binary program, solving it to optimality from scratch at every iteration of the cutting plane algorithm is very time consuming, especially considering the fact that its size significantly increases with the addition of Benders cuts (up to $|S|$ per iteration). Therefore, we incorporate the cutting plane algorithm into the branch-and-bound algorithm to obtain a B&C algorithm. This is also referred to as the one-tree implementation in the literature, as only a single branch-and-bound tree is created.

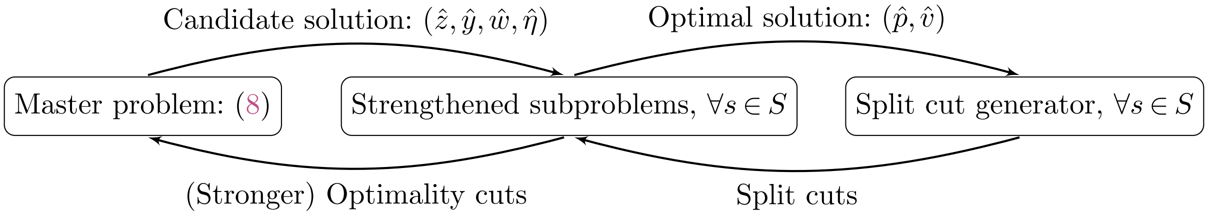
5.3.2. Cuts from fractional solutions. We intervene to the branch-and-bound tree and also generate Benders cuts (10) at nodes with a fractional solution, i.e., using candidate MP solutions violating some binary restrictions, as they are valid as well. These cuts are quite helpful in improving lower bounds.

5.3.3. Heuristic solutions inside branch-and-bound. When we are at an integral node in the branch-and-bound tree and add some Benders cuts due to the mismatches in second-stage guessed and true costs, we indeed cut off the candidate integral solution. However, this candidate first-stage solution combined with the optimal solutions of the SPs constitutes a feasible solution to the EF, thus provides an upper bound on its optimal value.

5.3.4. Cut management strategies. We follow the same strategies suggested by Bodur et al. (2017), in their Implementation Details section, in order to carefully select which cuts to add in terms of different criteria such as the depth of the search tree, cut violation amount and the round of cuts added at every node.

5.3.5. Cut-and-project (C&P) framework. Observing that our SP structure is similar to the capacitated facility location problem, for which the C&P framework proposed by Bodur et al. (2017) is shown to be very effective, we also generate the so-called split cuts to strengthen our SPs via the C&P approach. An illustration of this approach is provided in Figure 3.

Figure 3 **Cut-and-project Framework.**


Candidate solution: $(\hat{z}, \hat{y}, \hat{w}, \hat{\eta})$
Optimal solution: $(\hat{p}, \hat{v})$
Master problem: (8)
Strengthened subproblems, $\forall s \in S$
Split cut generator, $\forall s \in S$
(Stronger) Optimality cuts
Split cuts

In this framework, a master candidate solution is passed to the subproblems as in the cutting plane algorithm (in Figure 2), however, SPs are first strengthened with the addition of some cuts (linking p, v, y and w variables) before the generation of the optimality cuts for the MP. More specifically, when an SP is solved for a fixed candidate master solution, its optimal solution, $(\hat{p}, \hat{v})$, is fed into a cut generator to add new constraints (i.e., split cuts) to the SP for making it tighter. Then, the strengthened SPs are re-solved, and potentially stronger optimality cuts are obtained for the MP. The reader is referred to the study of Bodur et al. (2017) for more implementation details (e.g., split cut generation heuristic, and its integration into a branch-and-cut setting) which are followed in our study.

6. Extension to a Risk-averse Setting

The SMDCP operation is susceptible to a large penalty for failing to serve a customer in the second-stage. Minimizing only the expected value of this penalty, the risk-neutral SMDCP model is more suitable when delivery system conditions remain approximately the same, e.g., when crowd-shipper availability and customer locations do not change significantly every day. In such a stable operation setting, the risk-neutral model solutions would perform well in the long run. Thus, this model can be useful for a courier company that adopts the SMDCP operation for a fixed set of zones (i.e., customers are represented as a zone) and has a stable daily crowd-shipper pool. However, in our last-mile delivery system for an urban area, the set of customer locations and the pool of signed up crowd-shippers vary day by day. As such, the solutions obtained from the risk-neutral model may perform poorly under some scenarios. Therefore, incorporating risk to minimize the high cost of unfavorable outcomes in the SMDCP may be appealing.

In this section, we extend our risk-neutral SMDCP model into a risk-averse setting. More specifically, we incorporate the risk measure conditional-value-at-risk (CVaR) into (1) by modifying its objective function as

$$\min \; F(z,y) + (1-\lambda)\mathbb{E}_{\xi}[Q(y,w,\xi)] + \lambda \mathrm{CVaR}_{\alpha}[Q(y,w,\xi)]$$

to obtain a two-stage mean-risk stochastic programming formulation of SMDCP. Here, CVaR_{α} denotes the CVaR at level α, and $\lambda \in [0,1]$ is the convex combination weight used to balance the mean and risk of the second-stage cost.

CVaR is a coherent risk measure. Thanks to its convexity, it can be easily incorporated into our existing decomposition framework by simply modifying the MP as follows:

$$\begin{aligned}
\min_{z,y,w,\eta,\delta,\theta} \; & \sum_{i\in I} c_i^z z_i + \sum_{j\in J} c_j^y y_j + (1-\lambda)\sum_{s\in S}\frac{1}{|S|}\eta_s + \lambda\left(\delta + \frac{1}{1-\alpha}\sum_{s\in S}\frac{1}{|S|}\theta_s\right) && \text{(11a)}\\
\text{s.t.} \; & (7\text{b}),(7\text{c}),(7\text{d}),(7\text{i}) && \text{(11b)}\\
& [\text{Benders cuts linking } \eta, y \text{ and } w \text{ variables}] && \text{(11c)}\\
& \theta_s \geq \eta_s - \delta \qquad \forall s \in S && \text{(11d)}\\
& \eta \in \mathbb{R}_+^{|S|}, \delta \in \mathbb{R}, \theta \in \mathbb{R}_+^{|S|}. && \text{(11e)}
\end{aligned}$$

As a result, we apply our B&C and C&P methods to solve (11), where we also incorporate all the enhancements that are discussed in Section 5.3. To the best of our knowledge, this is the first application of the C&P approach in a risk-averse setting. We note that for mean-risk models, a specific Benders decomposition algorithm is proposed by Noyan (2012), where constraints (11d) are not included in the initial MP and enforced with another family of cuts generated from the SPs. We leave the comparison of this algorithm with ours to future research.

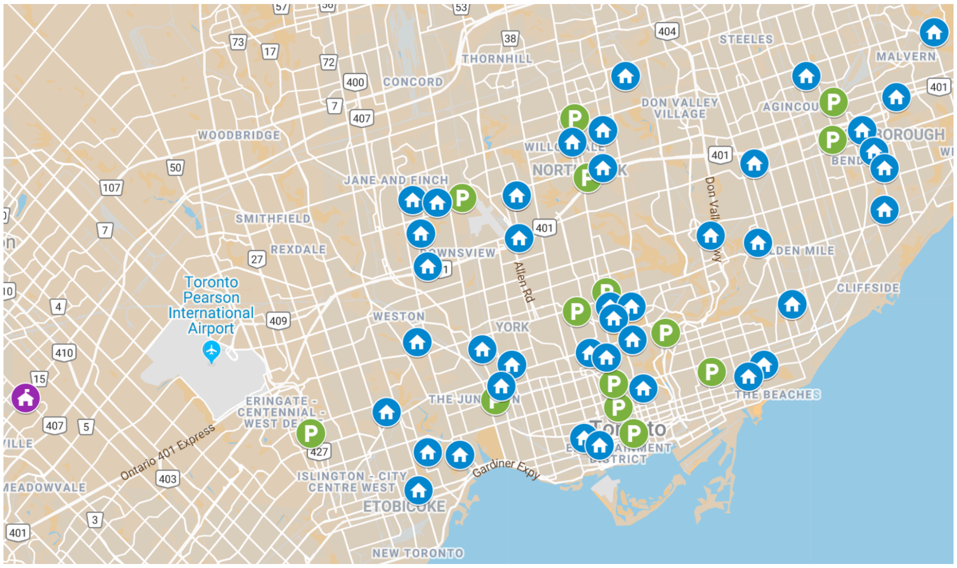
7. Computational Study

In this section, we present our computational results to provide insights on (i) cost savings of our mobile depot model over a traditional delivery model, i.e., the capacitated vehicle routing problem (CVRP), (ii) computational efficiency of the proposed enhanced Benders decomposition algorithm, and (iii) the importance of incorporating stochasticity and a risk measure into our model. More specifically, we consider a case study of the City of Toronto, where our instances are generated based on real data, as explained in detail in Section 7.1. After describing some implementation details in Section 7.2, we provide a comparison of our model with the CVRP by conducting a sensitivity analysis on the cost parameters in Section 7.3. In Section 7.4, we provide the results of the SAA analysis, for justifying the chosen number of scenarios in stochastic models in order to obtain high quality solutions. In Section 7.5, we compare the computational performance of alternative solution methods, namely directly solving the EF and applying the Benders decomposition in B&C and C&P frameworks. In Section 7.6, we present the value of incorporating crowd-shipper availability uncertainty into our model by comparing the solution quality of the stochastic model to the deterministic one. Lastly, in Section 7.7, we compare the solutions obtained from the risk-neutral and risk-averse models.

7.1. Instance generation

The City of Toronto is considered for generating our benchmark set. The 2016 version of the Transportation Tomorrow Survey (TTS), a comprehensive travel survey that is conducted every five years in the Toronto Area, is used to generate the instances. The City of Toronto has 625 traffic zones, for which almost 20 percent of all origin-destination pairs have non-zero demand. Our test instances are generated at the traffic zone level (i.e., traffic zones represent customer locations, mobile depot stopping locations, and origin or destination of crowd-shippers). Customers locations are randomly generated in proportion to the population of each traffic zone. Crowd-shippers are also randomly generated based on the TTS demand for origin-destination pairs. Potential mobile depot stopping locations are selected by considering proximity to neighborhood centers and the availability of a large outdoor parking lot in the selected mobile depot location. The main depot of our study is located in the Region of Peel, a major freight hub in North America, which is located to the west of the City of Toronto. Figure 4 shows the customer locations, potential mobile depot stopping location, and the main depot for our largest instance.

Figure 4 Map of Toronto with Customer Locations at (blue) Home Signs, Potential Mobile Depot Locations at (green) Parking Lot Signs, and the Main Depot at the (purple) Headquarter Sign with a Flag on Top.



In order to incorporate mobile depot and crowd-shipper time windows implicitly, two time frames representing morning and evening commutes are considered. Half of the crowd-shippers are randomly selected as morning commuters and the other half as evening commuters. Each mobile depot stopping location is duplicated to enable both morning and evening service to crowd-shippers. As such, our largest instance shown in Figure 4 has 40 customer locations and 30 mobile depot stopping locations.

The cost of opening a mobile depot is equal to μc^z, where c^z is equal to the shortest commuting distance by a truck from the main depot to the mobile depot stopping location, and μ is the relative cost ratio of a mobile depot (i.e, a truck) to a crowd-shipper compensation. We set $\mu = 1$ as the default value, otherwise it is noted (e.g., for the sensitivity analysis in Section 7.3). The capacity of a mobile depot, C, is set to $C = 10$ for all computational experiments in this paper. The cost of not serving a customer in the first stage (c^y) is equal to 0.6 times the shortest distance between the main depot and the customer location. The cost of not serving a customer in the second stage (c^v) is 1.5 times the cost of not serving in the first-stage (c^y). Crowd-shippers are paid based on a two-part compensation scheme, $c_{ijk} = c^1 + c^2 l_{ijk}$, where c^1 is the fixed compensation rate for accepting a delivery task, while c^2 is the compensation factor per unit of distance deviation from their original route, l_{ijk}. Dahle et al. (2019) show that the two-part compensation scheme yields the highest savings among the four compensation schemes in their study. For all computational experiments in this paper, $c^1 = 1$ and $c^2 = 0.6$, selected from the ranges considered by Dahle et al. (2019). The availability of each crowd-shipper is modeled as an independent Bernoulli distribution with the probability of showing up equal to p_{cs}. For simplicity in our sensitivity analysis, an equal probability of availability is assumed for all crowd-shippers.

We note that for all the aforementioned parameters values, after determining their reasonable ranges and relative magnitudes, we conducted some preliminary experiments to make sure that the generated instances are interesting in the sense that they do not have trivial solutions (such as no mobile depot is opened, only few crowd-shippers are chosen, or first-stage and second-stage costs are highly unbalanced).

7.2. Implementation Details

We run all experiments on a Linux workstation with 3.6GHz Intel Core i9-9900K CPUs and 128GB memory. We use IBM ILOG CPLEX 12.10 as the mixed-integer programming solver. We implement the B&C and C&P versions of the Benders decomposition algorithm via callback functions of CPLEX.

For algorithm performance comparison in Section 7.5, we use a time limit of two hours and set the number of threads to one, whereas for the other sections we do not impose any such limits.

Our B&C and C&P parameter settings are all borrowed from Bodur et al. (2017) except the following changes. We do not consider the “success” ratio for cut generation from scenarios, but rather add all of the violated cuts found at every iteration. We generate split (a.k.a. GMI) cuts for the subproblems only at every 20 nodes up to depth four in the branch-and-bound tree. Also, at the root node, we generate at most $10|S|$ split cuts in total for the subproblems.

7.3. Potential Benefits of Deterministic Mobile Depot with Crowd-shipping Problem over CVRP

We first analyze the potential cost-savings of our last-mile delivery business model in a deterministic setting where all the crowd-shippers are available compared to the traditional delivery model of the CVRP. This can be seen as the first step to motivate the SMDCP since the value in this most flexible deterministic setting bounds the value of the SMDCP. In the rest of the section, we refer to the Deterministic Mobile Depot with Crowd-shipping Problem as DMDCP.

The quantitative comparison is based on three instances with $(|I| = 10, |J| = 10)$, $(|I| = 20, |J| = 20)$ and $(|I| = 20, |J| = 30)$. The sensitivity analysis is done based on three factors including the ratio of crowd-shippers to customers $(|K|/|J|)$, the truck capacity in the CVRP (Q), and the relative cost of truck operation to crowd-shipper compensation (μ), for both the CVRP and DMDCP. Three levels of crowd-shippers to customers ratio, two levels of truck capacity, and three levels of the relative cost of truck operation to crowd-shipper compensation are considered, namely $|K|/|J| \in \{2, 3, 4\}$, $Q \in \{5, 10\}$, and $\mu \in \{1, 1.5, 2\}$. We do not consider capacity of the CVRP trucks to be more than mobile depot capacity, which is fixed to 10, since large trucks (i.e., larger than the mobile depots) are not suitable for customer delivery, especially in urban areas. For a fair comparison, since the operational cost of mobile depots is equal to the distance between the main depot and mobile depot locations in our model, the last leg of routes to the main depot is not considered in the operational cost of the CVRP. For each instance, five sets of customers are randomly generated and the average operational costs are reported.

Table 2 provides the quantitative results of the comparison between the DMDCP and CVRP for a variety of instances, specified by $(|I|, |J|)$, at different levels of relative cost of trucks to crowd-shippers (μ) and the ratio of crowd-shippers to customers $(|K|/|J|)$. The third column of the table provides the average operational cost of the CVRP solutions for two levels of truck capacity (Q). The columns labeled “DMDCP cost” report the operational cost of the DMDCP solutions for three levels of crowd-shippers to customers ratio. Lastly, the columns labeled “Cost saving %” provide the average percentage cost saving of the DMDCP solutions over the CVRP ones, where the individual instance percentages are averaged over five instances. The details on cost savings for each instance are provided in Table 10 in Appendix B.

Table 2 Sensitivity Analysis Results of DMDCP and CVRP. (Each Row Corresponds to Averages of Five Random Instances.)

			Crowd-shippers to customers ratio					
			$\lvert K\rvert/\lvert J\rvert = 2$		$\lvert K\rvert/\lvert J\rvert = 3$		$\lvert K\rvert/\lvert J\rvert = 4$	
Instance ($\lvert I\rvert, \lvert J\rvert$)	μ	CVRP cost (Q)	DMDCP cost	Cost saving %	DMDCP cost	Cost saving %	DMDCP cost	Cost saving %
(10, 10)	1	115.31 (5)	85.8	28.5	76.9	33.3	73.2	36.5
		87.45 (10)		5.7		12.1		16.3
	1.5	172.96 (5)	104.9	39.4	96.5	44.2	92.2	46.7
		131.17 (5)		20.1		26.4		29.7
	2	230.62 (10)	123.9	46.3	115.8	49.8	111.2	51.7
		174.9 (10)		29.2		33.8		36.4
(20, 20)	1	200.52 (5)	187.3	6.3	162.9	18.5	153.3	23.3
		138.68 (10)		-35.5		-17.9		-10.9
	1.5	300.78 (5)	222.6	25.8	198.6	33.7	188.9	37.0
		208.02 (10)		-7.3		4.2		8.9
	2	401.04 (5)	257.8	35.6	233.8	41.5	224.0	44.0
		277.36 (10)		6.8		15.4		19.0
(20, 30)	1	290.45 (5)	253.7	12.3	228.0	21.2	210.9	27.1
		187.13 (10)		-36.1		-22.3		-13.1
	1.5	435.67 (5)	309.9	28.6	281.0	35.3	264.7	39.0
		280.69 (10)		-10.8		-0.4		5.4
	2	580 (5)	362.5	37.6	333.1	42.5	317.0	45.3
		374.26 (10)		3.1		10.8		15.1

The comparison shows significant cost savings of the DMDCP compared to the CVRP for most cases. Our analysis shows an increase in cost savings of the DMDCP with an increase in crowd-shippers to customer ratio. For ratios of 2, 3, and 4, the average cost savings across all the instances, Q values, and μ values are 13.1%, 21.2%, and 25.4%, respectively. As expected, the cost savings are larger when the capacity of the trucks in the CVRP is lower. For instance, when the crowd-shippers to customers ratio is $|K|/|J| = 3$, and capacity is $Q = 5$, the average cost savings of the DMDCP are above 18% for all combinations of instances and μ values (and, in fact, we observed more than 40% gain in half of the instances, with a maximum of 54% gain, see Table 10). This indicates that the aggregation in the first-tier operation of the DMDCP can be quite useful. Also, as expected, when the truck costs increase, then the savings by the DMDCP increase. Furthermore, we find that for the instance with the fewest customers, $|J| = 10$, the DMDCP has the largest savings compared to the instances with a higher number of customers. This shows the advantage of the DMDCP when there are only a few customers whose locations are scattered in a large network. Intuitively, mobile depots bring customer packages to urban centres and crowd-shippers can deliver the packages on their way, to the customers who live far from urban centres, with a low cost. In contrast, in the

CVRP, trucks are required to route through the whole city to make deliveries to a small number of scattered customer locations.

7.4. SAA Analysis

SAA analysis is performed, as explained in Section 5.1, in order to derive statistically valid bounds on the optimal value of the SMDCP model, and in turn to determine a sufficient number of scenarios, $|S|$, needed to obtain high quality solutions.

Two instances are chosen to conduct the SAA analysis with three levels of probability of crowd-shipper availability, $p_{cs} = \{0.4, 0.5, 0.6\}$. After some preliminary tests, four numbers of scenarios are considered, namely $|S| = N \in \{50, 100, 200, 300\}$. The other parameters of the SAA procedure are set as $M = 25$, $|S^{pick}| = 2|S|$ and $S^{eval} = 10000$.

The result summary for the SAA analysis is presented in Table 3. For each number of scenarios, $|S|$, and the crowd-shipper availability probability, p_{cs}, the worst optimality gap percentage, obtained via formula (7), is provided. The reported optimality gap is calculated from the 95% confidence intervals for lower and upper bounds on the SMDCP optimal value, that are respectively calculated using (6) and (5), which can be found in Table 9 in Appendix B.

Table 3 Optimality Gap Percentages of SAA Analysis.

Instance	$\lvert S\rvert$	$p_{cs} = 0.4$	$p_{cs} = 0.5$	$p_{cs} = 0.6$
(10,10,30)	25	3.65	3.83	2.18
	50	3.40	1.82	1.47
	100	1.61	1.28	0.75
	200	0.78	0.80	0.69
	300	0.68	0.53	0.44
(10,20,60)	25	2.62	1.86	0.80
	50	1.41	1.37	0.70
	100	0.94	0.93	0.62
	200	0.73	0.64	0.29
	300	0.70	0.61	0.53

For both instances in all levels of p_{cs}, the worst gap is below 1% for $S = 200$, which shows the sufficient number of scenarios for obtaining near-optimal solutions for the SMDCP. Therefore, for our computational experiments in Sections 7.6 and 7.7, the number of scenarios is fixed to $|S| = 200$.

7.5. Algorithmic Performance Comparison

This section is dedicated to the computational analysis of the performance of the three solution algorithms for the SMDCP, namely solving the EF directly and the two decomposition algorithms. More specifically, we look at the efficiency of the alternative methods for the original risk-neutral SMDCP model, as well as its risk-averse extension. The goal is to decide which method to use in finding exact solutions to the EF for instances of different characteristics to be analyzed in Sections 7.6 and 7.7 for the value of stochasticity and risk incorporation, respectively.

Four instances with $p_{cs} = 0.5$ are analyzed for four scenario levels for the risk-neutral case. The summary of the results is provided in Table 4. As expected, the EF does not scale well with the number of scenarios, and is only able to solve small instances in the given time limit of two hours. Despite the fact that the decomposition algorithms cannot solve large instances to optimality either, they provide significantly smaller optimality gaps. Nevertheless, we find that both of these methods have usually discovered near-optimal solutions by the end of the time limit. Regarding the addition of split cuts into subproblems in the C&P approach, we observe that they can indeed be helpful, however, in this problem class, subproblems take significant time to solve even in the B&C approach, as such the contribution of the split cuts is hindered by the difficulty in solving the growing subproblems and the time spent in generating the cuts. We also find that, for the instances whose deterministic version is easier to solve, the C&P is more helpful, especially for cases with a large number of scenarios.

Table 4 **Summary of the Algorithmic Performance Analysis for the Risk-neutral Model for Instances with $p_{cs} = 0.5$.**

		Time(sec) / Gap(%)		
Instance	$\|S\|$	EF	B&C	C&P
(10,10,30)	50	**1.60**	**0.20**	**3.1**
	100	**7.12**	**0.52**	**5.5**
	200	**19.06**	**0.96**	**10.1**
	300	**77.36**	**1.76**	**15.9**
(10,20,60)	50	**97.29**	**9.22**	**25.2**
	100	**819.08**	**12.65**	**42.9**
	200	**4511.35**	**47.83**	**108.9**
	300	3.27%	**47.50**	**193.4**
(20,30,90)	50	17.06%	**5305.45**	**6088.64**
	100	20.67%	7.41%	2.07%
	200	57.55%	7.48%	6.11%
	300	58.74%	9.97%	7.14%
(30,40,120)	50	21.93%	10.64%	13.10%
	100	62.18%	15.97%	14.59%
	200	100.00%	22.75%	16.93%
	300	100.00%	19.54%	18.71%

In Table 5, we compare the two decomposition frameworks in the risk-averse setting. We present the results for one medium-size instance varying the crowd-shipper availability probability (p_{cs}), as well as the CVaR risk level (α) and the convex combination parameter (λ) balancing the expected second-stage cost and its CVaR value.

Table 5 Effect of Adding Split Cuts to Subproblems for the Risk-averse Formulation for an Instance with ($|I| = 20, |J| = 20, |K| = 60$).

		$p_{cs} = 0.4$		$p_{cs} = 0.5$	
α	λ	B&C time	C&P time saving (%)	B&C time	C&P time saving (%)
0.7	0.00	1266.1	39.5	656.1	17.9
	0.25	928.4	33.7	691.9	-4.2
	0.50	1850.2	64.5	420.4	-28.9
	0.75	687.9	-12.5	1075.2	40.4
	1.00	690.6	36.6	453.7	20.4
0.8	0.00	1023.8	24.0	548.6	11.5
	0.25	1887.3	60.9	759.1	32.0
	0.50	1617.4	37.9	480.6	-72.2
	0.75	1421.4	54.7	528.1	3.0
	1.00	1080.6	69.9	318.5	12.5
0.9	0.00	972.2	25.1	353.2	-55.0
	0.25	1085.4	60.3	363.1	-86.0
	0.50	1061.1	27.7	439.2	-20.4
	0.75	1016.4	8.0	475.5	-19.4
	1.00	1406.8	85.4	455.4	46.8

The results from Table 5 show the substantial time savings of the C&P approach for p_{cs}= 0.4; however, the time savings for $p_{cs} = 0.5$ are mixed. The B&C approach solution time is lower for the case with p_{cs}=0.5 compared to the case with p_{cs}=0.5. The effort for generation of split cuts seems to pay off by accelerating the B&C approach significantly in the case with p_{cs} =0.4. We also observe that the time saving is generally larger for cases with $\lambda = 1$. This may show the effectiveness of split cuts when the stochasticity is given more weight. More computational experiments are needed to investigate the effectiveness of split cuts in the CVaR models, however, for our model the split cuts improve the solution time, especially for the instances that are computationally more demanding.

7.6. Comparison of Deterministic and Stochastic Models, Value of Stochasticity

To assess the value of incorporating stochasticity, we compare the quality of the implementable (i.e., first-stage) solutions from the stochastic model and a deterministic strategy by calculating the Expected Value of Perfect Information (EVPI) and Expected Value of Stochastic Solution (EVSS).

EVPI is the maximum cost a decision maker would like to pay to obtain the perfect information about the future uncertainty. Given a sample of scenarios $\{\xi^s\}_{s \in S}$, EVPI is defined in (12) as the difference between the optimal value of the stochastic model and the average of the optimal values of individual scenario deterministic problems.

$$\text{EVPI} := \hat{\nu}(S) - \frac{1}{|S|} \sum_{s \in S} \hat{\nu}(\{\xi^s\}) \tag{12}$$

Value of Stochastic Solution (VSS) is used to justify the extra effort for modeling and solving the stochastic model. When the uncertain parameters are continuous, i.e., the random variables have continuous probability distributions, VSS is defined as the cost difference for the solutions obtained from the stochastic model and the deterministic model that is built by considering the *expected scenario*, i.e., the scenario constructed by using the expected value of each random variable. However, the SMDCP requires a different approach for VSS calculation due to working with discrete distributions. In other words, the expected scenario does not make sense in our setting as the scenarios must be binary vectors representing crowd-shippers' availability status. This also implies that in this setting, considering the stochastic model is a much more reasonable approach than relying on a deterministic model since there is no intuitive way to choose a representative scenario, as the expected one, for the deterministic model.

One approach for VSS is to consider the most likely scenario for the deterministic problem (Liu, Fan, and Ordóñez 2009); however, the likelihood of scenario ξ^s in the SMDCP is $\prod_{k \in K}((1-p_{cs})(1-\xi^s_k)+p_{cs}\xi^s_k)$, which is low. So, instead, we define Expected VSS (EVSS) in (13) as a measure of the value of the stochastic solution:

$$\text{EVSS} := \nu^{DET}(S) - \hat{\nu}(S) \tag{13}$$

where

$$\nu^{DET}(S) := \frac{1}{|S|}\sum_{s \in S} \nu^{DET}(\{\xi^s\})$$

and for each $s \in S$,

$$\nu^{DET}(\{\xi^s\}) := F\left(\hat{z}(\{\xi^s\}), \hat{y}(\{\xi^s\})\right) + \frac{1}{|S|}\sum_{s' \in S} Q\left(\hat{y}(\{\xi^s\}), \hat{w}(\{\xi^s\}), \xi^{s'}\right).$$

In other words, we first solve the deterministic problem for each scenario ξ^s, obtain its optimal first-stage solution $(\hat{z}(\{\xi^s\}), \hat{y}(\{\xi^s\}), \hat{w}(\{\xi^s\}))$, and evaluate its true cost $\nu^{DET}(\{\xi^s\})$ using the original sample S. We then average those individual scenario costs to compute $\nu^{DET}(S)$, the expected value of the deterministic solution objective function quality, and compare it with the optimal value of the stochastic model, $\hat{\nu}(S)$, to compute EVSS.

Table 6 shows the results of the comparison for two instances with different probabilities for crowd-shipper availability, where the number of scenarios is fixed to $|S| = 200$. In the SMDCP column, we provide the optimal value of the considered SAA problem. In the fourth column, we reported "EVPI (%)", corresponding to a relative expected value of having a perfect information as $100(\text{EVPI}/\hat{\nu}(S))$. In the column labeled "EVSS (%)", we provide the relative gap between $\nu^{DET}(S)$ and $\hat{\nu}(S)$, calculated as $100(\nu^{DET}(S) - \hat{\nu}(S))/\nu^{DET}(S)$, to indicate the gain by the stochastic solutions. In the last two columns of the table, we respectively report the max and min of the percentage gain of the stochastic solution over the individual scenario values $\{\nu^{DET}(\{\xi^s\})\}_{s \in S}$.

The results indicate that the stochastic solutions provide significant benefits over the deterministic ones. Also, increasing the value of the crowd-shipper availability probability reduces the

Table 6 **Analysis of Quality of Stochastic Solution (with $|S| = 200$).**

Instance	p_{cs}	SDMCP	EVPI (%)	EVSS (%)	VSS-Max (%)	VSS-Min (%)
(20, 20, 60)	0.4	244.1	8.62	6.08	29.45	0.39
	0.5	206.4	5.83	4.45	35.13	0.00
	0.6	186.4	4.92	3.35	31.54	0.00
(20, 30, 90)	0.4	350.7	10.00	6.00	14.37	1.59
	0.5	304.4	7.82	4.82	11.82	1.05
	0.6	276.4	6.41	3.99	11.29	0.04

EVSS and EVPI considerably, which shows the sensitivity of the SMDCP to the availability of crowd-shippers.

We also assess the contribution of crowd-shippers to customers ratio and crowd-shippers availability probability to the value of stochastic solution, using an instance with $(|I| = 20, |J| = 20)$. Table 7 provides this analysis for different combinations of $|K|/|J|$ and p_{cs} values, at three specific levels of *expected* ratio of crowd-shippers to customers, $p_{cs} \cdot |K|/|J|$.

Table 7 **Ratio of crowd-shippers to customers v.s. crowd-shippers availability probability analysis for an instance with $(|I| = 20, |J| = 20)$**

$p_{cs} \cdot \lvert K\rvert/\lvert J\rvert$	$(\lvert K\rvert/\lvert J\rvert, p_{cs})$	Total Cost	EVSS(%)
1	(2, 0.500)	267.8	6.02
	(3, 0.333)	280.6	7.24
	(4, 0.250)	290.8	6.32
	(5, 0.200)	296.5	8.16
	(6, 0.167)	296.0	7.31
	(7, 0.143)	298.0	6.83
	(8, 0.125)	302.0	6.91
1.5	(2, 0.750)	190.2	3.78
	(3, 0.500)	206.4	4.45
	(4, 0.375)	218.0	5.30
	(5, 0.300)	229.1	6.86
	(6, 0.250)	227.7	5.80
	(7, 0.214)	230.2	5.42
	(8, 0.187)	232.2	6.27
2	(2, 1.000)	156.8	0.00
	(3, 0.666)	175.9	3.19
	(4, 0.500)	187.0	4.10
	(5, 0.400)	196.8	5.01
	(6, 0.333)	193.7	6.10
	(7, 0.286)	199.8	5.32
	(8, 0.250)	202.4	5.05

The results show the increasing trend of Total Cost with decrease in p_{cs} (or increase in $|K|/|J|$) at a specific level of expected crowd-shippers to customer ratio. However, the trend is different for

EVSS; decrease in the value of p_{cs} increases EVSS initially, and then (after potential fluctuation due to randomness) decreases EVSS. The reason is that the problem becomes more deterministic as the crowd-shipper availability probability gets closer to 0 or 1. Based on our analysis it would be better to have a lower number of potential crowd-shippers with high availability probability compared to higher number of potential crowd-shippers with lower availability probability. For example, for $p_{cs} \cdot |K|/|J| = 1.5$, the total cost and EVSS for $|K|/|J| = 3, p_{cs} = 0.5$ is 206.4 and 4.45, respectively. In contrast these values for $|K|/|J| = 8, p_{cs} = 0.187$ is 232.2 and 6.27. Companies may benefit more by investing in increasing crowd-shippers' availability probability rather than increasing the pool of potential crowd-shippers.

7.7. Assessing Risk in the Model via Conditional Value-at-Risk

In this section, the risk-averse approach is analyzed and compared to the risk-neutral approach based on two instances with $(|I| = 20, |J| = 20, |K| = 60)$ and $(|I| = 20, |J| = 30, |K| = 90)$, and $p_{cs} \in \{0.4, 0.5\}$. The sensitivity analysis is done on instances with parameters of $\lambda \in \{0, 0.5, 1\}$ and $\alpha \in \{0.7, 0.8, 0.9\}$. Table 8 provides the results by reporting the first- and second-stage costs (as well as their sum), and the CVaR value of the (first-stage) solutions.

We observe that when λ increases (i.e., when the CVaR cost gets a larger weight), the risk-averse formulation tends to invest more on the first-stage, i.e., sending more mobile depots or postponing customer deliveries, to decrease the reliance on the uncertainty of crowd-shippers and avoid a higher penalty for not serving the customers in the second-stage, which results in a decrease in the CVaR and second-stage costs. For example, for the instance with $(|I| = 20, |J| = 20, |K| = 60)$, and $p_{cs} = 0.4$, when $\alpha = 0.9$ and $\lambda = 0$, the first-stage cost is 82, but for $\lambda = 1$, the first-stage cost increases significantly to 144. In contrast, the value of the second-stage cost in the latter and the former are 125 and 162, respectively, which shows the decrease in the second-stage cost by increasing λ to 1.

We also find that using the risk-averse model can help in getting solutions that are near optimal but with a much lower risk value. For example, in the instance with $(|I| = 20, |J| = 20, |K| = 60)$, $p_{cs} = 0.4$, and $\alpha = 0.8$, for $\lambda = 0$, the expected total cost, and CVaR cost are 244.1, and 224.6, respectively. In contrast, for $\lambda = 0.5$, these costs are 246.2, and 199.8, respectively, which shows that only a 2.1 unit increase in the expected total cost leads to 24.8 units decrease in the CVaR value.

When only the CVaR cost considered in the objective function of the risk-averse model (i.e., λ $= 1$), the reverse is also true. Although the expected total cost is much higher than the case of $\lambda = 0.5$, the change in the CVaR cost is negligible. For example, in the instance with $(|I| = 20, |J| = 30, |K| = 90)$, $p_{cs} = 0.5$, and $\alpha = 0.7$, for λ=0.5, the expected total cost, and the CVaR cost are

Table 8 Risk Averse Analysis.

	Instance	p_{cs}	λ	First-stage cost	Second-stage cost	Expected total cost	CVaR cost
$\alpha = 0.7$	(20,20,60)	0.4	0.0	82.0	162.1	244.1	211.5
			0.5	82.0	162.2	244.2	208.4
			1.0	102.0	154.2	256.2	187.0
		0.5	0.0	75.0	131.4	206.4	160.2
			0.5	75.0	131.4	. 206.4	160.2
			1.0	82.0	134.5	216.5	151.5
	(20,30,90)	0.4	0.0	182.8	167.9	350.7	212.3
			0.5	182.8	167.9	350.7	212.3
			1.0	211.2	154.5	365.7	181.5
		0.5	0.0	134.8	169.6	304.4	207.7
			0.5	170.8	139.8	310.6	163.9
			1.0	170.8	151.2	322.0	163.8
$\alpha = 0.8$	(20,20,60)	0.4	0.0	82.0	162.1	244.1	224.6
			0.5	102.0	144.2	246.2	199.8
			1.0	102.0	158.5	260.5	199.5
		0.5	0.0	75.0	131.4	206.4	167.2
			0.5	75.0	131.4	206.4	167.2
			1.0	82.0	137.4	219.4	157.5
	(20,30,90)	0.4	0.0	182.8	167.9	350.7	227.2
			0.5	182.8	168.0	350.8	226.8
			1.0	226.0	145.5	371.5	176.9
		0.5	0.0	134.8	169.6	304.4	217.5
			0.5	170.8	139.8	310.6	169.9
			1.0	170.8	154.5	325.3	169.9
$\alpha = 0.9$	(20,20,60)	0.4	0.0	82.0	162.1	244.1	248.6
			0.5	102.0	144.8	246.8	220.8
			1.0	144.0	125.0	269.0	177.8
		0.5	0.0	75.0	131.4	206.4	179.4
			0.5	70.0	138.0	208.0	180.1
			1.0	82.0	134.8	216.8	166.5
	(20,30,90)	0.4	0.0	182.8	167.9	. 350.7	253.7
			0.5	225.6	137.3	362.9	190.9
			1.0	244.2	139.5	383.7	170.0
		0.5	0.0	134.8	169.6	304.4	235.3
			0.5	170.8	139.8	310.6	181.1
			1.0	170.8	155.5	326.3	181.0

310.6, and 163.9, respectively. However, for λ=1, the model decreases the CVaR cost only to 163.8 by increasing the expected total cost drastically to 322.0. We believe sensitivity analysis for the value of λ is necessary to get the best possible operational decisions in this model.

Since α defines a threshold for $100(1-\alpha)$ percent of the worst scenarios, the expected value for CVaR should intuitively increase with the increase in α, but our results are not consistent with our intuition. For example, in the instance with $(|I| = 20, |J| = 30, |K| = 90)$, $p_{cs} = 0.4$, and $\lambda = 1$, for

$\alpha = 0.7$, the CVaR cost is 181.5; in contrast, for $\alpha = 0.9$, the CVaR cost is 170. In this situation, the risk-averse model decides to invest more in the first-stage cost (e.g., not serving customers) in order to avoid the huge cost of the worst scenarios, which may lead to a lower CVaR cost since there is a smaller number of customers to service in the second-stage.

The risk-averse model also shows a higher sensitivity to lower values of p_{cs}. We notice a sharper decrease in the value of CVaR when shifting from risk-neutral to risk-averse model in the cases with a lower value of p_{cs}. In the instance with $(|I| = 20, |J| = 30, |K| = 90)$, $p_{cs} = 0.4$, and $\alpha = 0.9$, changing the λ from 0 to 1, the value of CVaR decreases by 73.7 units, whereas, for $p_{cs} = 0.4$, the decrease is 54.3 units.

The analysis shows the importance of considering risk in this crowd-shipping operation under uncertainty, which can provide managerial insights for crowd-shipping companies to hedge against the uncertainty in crowd-shippers' availability. We believe a more extensive study could be conducted, especially on the effect of the crowd-shipper compensation scheme, to shed more light on incorporating risk into a crowd-shipping operation model.

8. Conclusion

Growth in e-commerce and increasing demand for fast home delivery solutions is leading to a need for new solutions for last-mile deliveries in urban areas. This paper provides a methodology for optimizing a last-mile delivery solution that exploits the operational flexibility of mobile depots, and the relative cost efficiency of crowd-shipping for package delivery in urban areas. The two-tier delivery model presented in this paper optimally selects mobile depot locations in advance of full information about the availability of crowd-shippers, and then assigns to available crowd-shippers the final leg of the shipment to the customer. Uncertainty in the availability of crowd-shippers is incorporated by modeling the problem as a two-stage stochastic integer program and an enhanced decomposition solution algorithm is developed. Finally, a risk-averse approach is compared against a risk-neutral approach by assessing conditional value-at-risk. Scenarios are simulated for a City of Toronto case study, in which demand is a function of population density and crowd-shipper availability is a function of observed commuting patterns.

The results are promising. First, we compare a deterministic version of the two-tier model, DMDCP, to the CVRP. For most of the assessed instances (numbers of potential mobile depot locations, customers and crowd-shippers) and for varying relative costs of crowd-shipping and truck capacity, the DMDCP model outperforms the CVRP with an average cost savings of over 20%.

Second, for the stochastic version of the model, SMDCP, SAA analysis is performed to derive statistically valid bounds on the optimal value of the SMDCP solution, and to determine the number of required scenarios. We find that, for a variety of instances and parameter values, 200 scenarios are sufficient to attain a solution with a gap no worse than 1%. A performance analysis of EF, B&C and C&P solution algorithms finds that neither the EF nor the decomposition algorithms can solve large instances to optimality, but that the decomposition algorithms usually discover near-optimal solutions within two hours for instances up to a size of 30 mobile depot locations, 40 customers and 120 crowd-shippers.

Third, we show that the stochastic model provides higher quality solutions than the deterministic model. The EVSS is used to quantify the improvement in solution quality provided by introducing stochasticity. For the instances and degrees of uncertainty analyzed, the EVSS indicates a 3.35% to 6.08% improvement in the solution quality with stochasticity, and that improvement is greater when uncertainty in crowd-shipper availability is higher.

Finally, our assessment of the risk-averse approach shows that it would lead the operator to invest more in the first stage of the operation, by sending more mobile depots or postponing more customer deliveries, in order to reduce the risk of high penalties for non-delivery. In cases where the crowd-shipper availability is more uncertain, a risk-averse approach leads to greater reduction in CVaR. Understanding the trade-offs in applying a risk-averse approach is important for crowd-shipping companies that are attempting to minimize cost in an environment where crowd-shippers may be unreliable, and the penalty for non-delivery may be high.

Several areas of potential future research arise from this study. First, our analysis does not incorporate important aspects of real-life operations, including customer and crowd-shipper time-windows, congestion effects, and routing of mobile depots between multiple locations. Second, research should be conducted on the impacts of crowd-shipper compensation or incentive schemes that could reduce the costs associated with uncertainty in crowd-shipper availability. Third, there is a need to develop heuristics that can provide solutions for larger scale instances than those provided in this paper. Fourth, extending the proposed two-stage to multi-stage should be considered due to dynamic appearance of the crowd-shippers. Lastly, the effect of the C&P method should be assessed in the more proper decomposition algorithm for the CVaR risk-averse approach, introduced by Noyan (2012).

Acknowledgments

The authors wish to acknowledge the sources of financial support for the project, including the City of Toronto and the Region of York (through the Centre of Automated and Transformative Transportation Systems), the Region of Peel (through the Smart Freight Centre), and Natural Sciences and Engineering Research Council of Canada [Grant RGPIN- 2018-04984].

References

Archetti C, Savelsbergh M, Speranza MG (2016) *The vehicle routing problem with occasional drivers. Eur. J. Oper. Res.* 254(2):472–480.

Arslan AM, Agatz N, Kroon L, Zuidwijk R (2019) *Crowdsourced delivery—a dynamic pickup and delivery problem with ad hoc drivers. Transportation Sci.* 53(1):222–235.

Barr, Wohl (2013) *Exclusive: Wal-mart may get customers to deliver packages to online buyers.* https://www.reuters.com/article/us-retail-walmart-delivery/exclusive-wal-mart-may-get-customers-to-deliver-packages-to-online-buyers-idUSBRE92R03820130328.

Benders JF (1962) *Partitioning procedures for solving mixed-variables programming problems. Numerische Mathematik* 4(1):238–252.

Bodur M, Dash S, Günlük O, Luedtke J (2017) *Strengthened Benders cuts for stochastic integer programs with continuous recourse. INFORMS J. Comput.* 29(1):77–91.

Bodur M, Luedtke J (2017) *Mixed-integer rounding enhanced benders decomposition for multiclass service-system staffing and scheduling with arrival rate uncertainty. Management Sci.* 63(7):2073–2091.

Dahle L, Andersson H, Christiansen M (2017) *The vehicle routing problem with dynamic occasional drivers. International Conference on Computational Logistics*, 49–63 (Springer).

Dahle L, Andersson H, Christiansen M, Speranza MG (2019) *The pickup and delivery problem with time windows and occasional drivers. Comput. and Oper. Res.* 109:122–133.

Dayarian I, Savelsbergh M (2017) *Crowdshipping and same-day delivery: Employing in-store customers to deliver online orders.* https://pdfs.semanticscholar.org/675e/b192aff73261fa3124ded3959ba812fd491b.pdf.

Dellaert N, Dashty Saridarq F, Van Woensel T, Crainic TG (2019) *Branch-and-price–based algorithms for the two-echelon vehicle routing problem with time windows. Transportation Sci.* 53:463–479.

Gdowska K, Viana A, Pedroso JP (2018) *Stochastic last-mile delivery with crowdshipping. Transportation Research Procedia* 30:90–100.

Gendreau M, Laporte G, Séguin R (1996) *Stochastic vehicle routing. Eur. J. Oper. Res.* 88(1):3–12.

Halper R, Raghavan S (2011) *The mobile facility routing problem. Transportation Sci.* 45(3):413–434.

Insight Partners (2019) *Last mile delivery market to 2027 - global analysis and forecasts by technology (drones, autonomous ground vehicles, droids, semi-autonomous vehicles); type (b2b, b2c); application (3c products, fresh products, others).* https://www.theinsightpartners.com/reports/last-mile-delivery-market.

Kafle N, Zou B, Lin J (2017) *Design and modeling of a crowdsource-enabled system for urban parcel relay and delivery. Transportation Res. Part B: Methodological* 99:62–82.

Laporte G, Louveaux FV (1993) *The integer L-shaped method for stochastic integer programs with complete recourse. Oper. Res. Lett.* 13(3):133–142.

Lee J, Kim C, Wiginton L (2019) *Delivering last-mile solutions: A feasibility analysis of microhubs and cyclelogistics in the GTHA.* https://www.pembina.org/reports/delivering-last-mile-solutions-june-2019.pdf.

Lin J, Chen Q, Kawamura K (2016) *Sustainability SI: logistics cost and environmental impact analyses of urban delivery consolidation strategies. Networks and Spatial Economics* 16(1):227–253.

Liu C, Fan Y, Ordóñez F (2009) *A two-stage stochastic programming model for transportation network protection. Comput. and Oper. Res.* 36(5):1582–1590.

Macrina G, Pugliese LDP, Guerriero F, Laganà D (2017) *The vehicle routing problem with occasional drivers and time windows. International Conference on Optimization and Decision Science*, 577–587 (Springer).

Macrina G, Pugliese LDP, Guerriero F, Laporte G (2020) *Crowd-shipping with time windows and transshipment nodes. Comput. and Oper. Res.* 113:104806.

Marujo LG, Goes GV, D'Agosto MA, Ferreira AF, Winkenbach M, Bandeira RA (2018) *Assessing the sustainability of mobile depots: The case of urban freight distribution in rio de janeiro. Transportation Res. Part D: Transport and Environment* 62:256–267.

Noyan N (2012) *Risk-averse two-stage stochastic programming with an application to disaster management. Comput. and Oper. Res.* 39(3):541–559.

Oyola J, Arntzen H, Woodruff DL (2018) *The stochastic vehicle routing problem, a literature review, part I: models. EURO Journal on Transportation and Logistics* 7(3):193–221.

Raviv T, Tenzer EZ (2018) *Crowd-shipping of small parcels in a physical internet.* https://www.researchgate.net/publication/326319843_Crowd-shipping_of_small_parcels_in_a_physical_internet.

Rosenfield A, Lamers J, Nourinejad M, Roorda MJ (2016) *Investigation of commercial vehicle parking permits in toronto, ontario, canada. Transportation Research Record* 2547(1):11–18.

Shapiro A, Dentcheva D, Ruszczyński A (2014) *Lectures on stochastic programming: modeling and theory* (SIAM).

Van Duin J, Van Dam T, Wiegmans B, Tavasszy L (2016) *Understanding financial viability of urban consolidation centres. Transportation Research Procedia* 16:61–80.

Van Rooijen T, Quak H (2010) *Local impacts of a new urban consolidation centre–the case of binnenstadservice. nl. Procedia-Social and Behavioral Sciences* 2(3):5967–5979.

Verlinde S, Macharis C, Milan L, Kin B (2014) *Does a mobile depot make urban deliveries faster, more sustainable and more economically viable: results of a pilot test in brussels. Transportation Research Procedia* 4:361–373.

Wolsey LA, Nemhauser GL (1999) *Integer and combinatorial optimization*, volume 55 (John Wiley & Sons).

Appendix A: Proof of Proposition 1

Regarding the SP formulation given in (9), noting that (i) the upper bounds on the binary p and v in (9f) are redundant since the right-hand-side vector for the other constraints is binary, and (ii) the fixing constraints (9e) are not indeed constraints, but rather used to project out some variables from the formulation, thus do not impact the integrality results and can be ignored, we analyze the following simplified model:

$$\min \sum_{i\in I}\sum_{j\in J}\sum_{k\in K} c^{p}_{ijk} p_{ijk} + \sum_{j\in J} c^{v}_{j} v_j \tag{14a}$$

$$\text{s.t.} \sum_{i\in I}\sum_{j\in J} p_{ijk} \le \xi^{s}_{k} \qquad \forall k\in K \tag{14b}$$

$$\sum_{i\in I}\sum_{k\in K} p_{ijk} + v_j = 1 - \hat{y}_j \qquad \forall j\in J \tag{14c}$$

$$\sum_{k\in K} p_{ijk} \le \hat{w}_{ij} \qquad \forall i\in I, \forall j\in J \tag{14d}$$

$$p\in \mathbb{R}^{|I||J||K|}_{+}, v\in \mathbb{R}^{|J|}_{+}. \tag{14e}$$

We denote by $\mathbb{I}_{n\times n}$, $\mathbf{0}_{m\times n}$ and $\mathbf{1}_{m\times n}$ the identity matrix, matrix of zeros, matrix of ones, respectively, with dimensions provided as subscripts. Also, we use $(\cdot)^\top$ for the transpose operator.

Let $\mathbb{A} = [\mathbb{P}|\mathbb{V}]$ be the constraint coefficient matrix of (14b)-(14d), obtained by concatenating the coefficient matrix of the v variables, $\mathbb{V}$, to the right of the coefficient matrix of the p variables, $\mathbb{P}$. In what follows, we show that $\mathbb{P}$ is TU, which is sufficient for $\mathbb{A}$ to be TU since $\mathbb{V} = \left[\mathbf{0}_{|J|\times|K|}|\mathbb{I}_{|J|\times|J|}|\mathbf{0}_{|J|\times(|I|+|J|)}\right]^\top$ is a special matrix consisting of zero and identity matrix blocks, and appending it to $\mathbb{P}$ preserves totally unimodularity due to the properties of TU matrices, e.g., see (Wolsey and Nemhauser 1999, Proposition 2.1).

We further divide $\mathbb{P}$ into blocks corresponding to different constraint types as follows:

$$\mathbb{P} = \begin{bmatrix} \mathbb{P}_{(14b)} \\ \hline \mathbb{P}_{(14c)} \\ \hline \mathbb{P}_{(14d)} \end{bmatrix} = \left[\begin{array}{c|c|c|c|c|c|c|c|c|c}
\mathbb{I}_{|K|\times|K|} & \cdots & \mathbb{I}_{|K|\times|K|} & \mathbb{I}_{|K|\times|K|} & \cdots & \mathbb{I}_{|K|\times|K|} & \cdots & \mathbb{I}_{|K|\times|K|} & \cdots & \mathbb{I}_{|K|\times|K|} \\ \hline
\mathbf{1}_{1\times|K|} & & & \mathbf{1}_{1\times|K|} & & & & \mathbf{1}_{1\times|K|} & & \\
 & \ddots & & & \ddots & & \cdots & & \ddots & \\
 & & \mathbf{1}_{1\times|K|} & & & \mathbf{1}_{1\times|K|} & & & & \mathbf{1}_{1\times|K|} \\ \hline
\mathbf{1}_{1\times|K|} & & & & & & \cdots & & & \\
 & \ddots & & & & & \cdots & & & \\
 & & \mathbf{1}_{1\times|K|} & & & & \cdots & & & \\
 & & & \mathbf{1}_{1\times|K|} & & & \cdots & & & \\
 & & & & \ddots & & \cdots & & & \\
 & & & & & \mathbf{1}_{1\times|K|} & \cdots & & & \\
 & & & & & & \ddots & & & \\
 & & & & & & \cdots & \mathbf{1}_{1\times|K|} & & \\
 & & & & & & \cdots & & \ddots & \\
 & & & & & & \cdots & & & \mathbf{1}_{1\times|K|}
\end{array}\right]$$

We use the Ghouila-Houri's characterization of TU matrices (Wolsey and Nemhauser 1999, Theorem 2.7), for which we show that any subset of rows of $\mathbb{P}$, say R, can be partitioned into R_1 and R_2 such that for each column ℓ of $\mathbb{P}$, we have $\sum_{r\in R_1} \mathbb{P}_{r\ell} - \sum_{r\in R_2} \mathbb{P}_{r\ell} \in \{-1,0,1\}$.

Let R consist of the rows corresponding to $\bar{K}\subseteq K$, $\bar{J}\subseteq J$ and $\cup_{i\in\hat{I}}\{(i,j): j\in\hat{J}(i)\}$ with $\hat{J}(i)\subseteq J$ for all $i\in\hat{I}$ from constraints (14b), (14c) and (14d), respectively. We propose the following partition of the rows in R:

- Rows corresponding to $k \in \bar{K}$: All in R_2
- Rows corresponding to $j \in \bar{J}$: All in R_1
- Rows corresponding to $(i,j) \in \bigcup_{i \in \hat{I}} \{(i,j) : j \in \hat{J}(i)\}$:

—If $j \notin \bar{J}$, then in R_1

—Otherwise, in R_2

Then, we observe that for each column of $\mathbb{P}$, the difference between the sum of the elements in R_1 and R_2 sets is -1,0 or 1.

Appendix B: Detailed Computational Results

Table 9 provides the details of SAA analysis. Confidence intervals (CIs) for upper and lower bounds are reported in the form of "Mean ± Half Width" of the interval.

Table 9 The Detailed Summary of SAA Analysis.

		$p_{cs} = 0.4$			$p_{cs} = 0.5$			$p_{cs} = 0.6$		
Instance	$\|S\|$	CI on LB	CI on UB	**Gap**	CI on LB	CI on UB	**Gap**	CI on LB	CI on UB	**Gap**
(10,10,30)	25	124.8 ± 3.0	125.7 ± 0.2	3.65	105.9 ± 1.5	108.2 ± 0.5	3.83	97.9 ± 1.0	98.9 ± 0.1	2.18
	50	124.1 ± 2.0	125.8 ± 0.8	3.40	107.5 ± 1.0	108.2 ± 0.3	1.82	98.2 ± 0.7	98.9 ± 0.1	1.47
	100	125.3 ± 1.3	125.6 ± 0.4	1.61	107.7 ± 0.8	108.1 ± 0.2	1.28	98.8 ± 0.5	98.9 ± 0.1	0.75
	200	125.9 ± 0.9	125.6 ± 0.5	0.78	108.1 ± 0.5	108.2 ± 0.4	0.80	98.9 ± 0.3	99.0 ± 0.3	0.69
	300	125.8 ± 0.5	125.7 ± 0.5	0.68	108.1 ± 0.3	108.2 ± 0.2	0.53	98.8 ± 0.3	98.9 ± 0.1	0.43
(10,20,60)	25	248.3 ± 3.8	250.3 ± 1.0	2.62	212.7 ± 2.2	213.8 ± 0.9	1.86	190.9 ± 1.7	190.4 ± 0.4	0.80
	50	249.2 ± 2.8	249.4 ± 0.6	1.4	212.7 ± 1.6	213.6 ± 0.4	1.37	190.6 ± 1.2	190.3 ± 0.6	0.70
	100	249.2 ± 2.0	249.1 ± 0.5	0.94	212.9 ± 1.1	213.3 ± 0.6	0.93	190.6 ± 0.7	190.3 ± 0.8	0.62
	200	249.0 ± 1.3	249.2 ± 0.4	0.73	213.1 ± 0.8	213.3 ± 0.5	0.64	190.5 ± 0.6	190.4 ± 0.1	0.29
	300	248.8 ± 0.9	249.4 ± 0.3	0.70	212.9 ± 0.5	213.7 ± 0.0	0.61	190.3 ± 0.4	190.7 ± 0.3	0.53

Table 10 provides the detailed comparison of the DMDCP and CVRP for all five randomly generated sets of customers for each instance.

Table 10 Detailed Sensitivity Analysis Results to Compare the DMDCP and CVRP.

		crowd-shippers to customers ratio														
		$\lvert K\rvert/\lvert J\rvert = 2$					$\lvert K\rvert/\lvert J\rvert = 3$					$\lvert K\rvert/\lvert J\rvert = 4$				
μ	Instance	DMCDP	CVRP cost		DMCDP cost saving		DMCDP	CVRP cost		DMCDP cost saving		DMCDP	CVRP cost		DMCDP cost saving	
			$Q = 5$	$Q = 10$	$Q = 5$	$Q = 10$		$Q = 5$	$Q = 10$	$Q = 5$	$Q = 10$		$Q = 5$	$Q = 10$	$Q = 5$	$Q = 10$
1	(10,10)	87.4	110.3	35.0	83.8	14.5	79.7	110.3	83.8	27.7	4.9	72.9	110.3	33.9	83.8	13.1
		71.7	112.4	36.2	84.6	15.3	65.4	112.4	84.6	41.8	22.7	64.9	112.4	42.2	84.6	23.3
		99.1	124.6	20.5	92.1	-7.7	83.4	124.6	92.1	33.1	9.4	76.9	124.6	38.3	92.1	16.5
		85.8	115.3	25.6	91.7	6.4	80.0	115.3	91.7	30.6	12.7	79.7	115.3	30.8	91.7	13.0
		85.0	113.9	25.4	85.0	0.0	75.9	113.9	85.0	33.4	10.7	71.5	113.9	37.2	85.0	15.8
	(20,20)	156.8	195.6	19.9	139.7	-12.2	140.2	195.6	139.7	28.3	-0.4	133.2	195.6	31.9	139.7	4.7
		189.2	212.8	11.1	146.9	-28.8	158.2	212.8	146.9	25.6	-7.7	150.3	212.8	29.3	146.9	-2.3
		176.5	208.7	15.4	138.7	-27.2	150.3	208.7	138.7	28.0	-8.3	147.8	208.7	29.2	138.7	-6.5
		222.1	194.4	-14.3	131.9	-68.4	192.9	194.4	131.9	0.8	-46.2	175.5	194.4	9.7	131.9	-33.0
		191.8	191.1	-0.4	136.1	-40.9	172.7	191.1	136.1	9.6	-26.9	159.7	191.1	16.4	136.1	-17.3
	(20,30)	234.2	303.3	22.8	191.6	-22.3	207.6	303.3	191.6	31.6	-8.4	194.6	303.3	35.8	191.6	-1.6
		239.1	302.0	20.8	191.5	-24.9	217.2	302.0	191.5	28.1	-13.4	200.2	302.0	33.7	191.5	-4.5
		293.1	276.2	-6.1	175.9	-66.6	262.2	276.2	175.9	5.1	-49.0	244.4	276.2	11.5	175.9	-38.9
		253.8	288.6	12.1	189.0	-34.3	224.2	288.6	189.0	22.3	-18.7	206.7	288.6	28.4	189.0	-9.4
		248.3	282.2	12.0	187.7	-32.3	228.6	282.2	187.7	19.0	-21.8	208.8	282.2	26.0	187.7	-11.3
1.5	(10,10)	106.4	165.5	35.7	125.8	15.4	98.7	165.5	125.8	40.3	21.5	91.9	165.5	44.5	125.8	27.0
		90.7	168.6	46.2	127.0	28.6	84.4	168.6	127.0	49.9	33.5	83.9	168.6	50.2	127.0	33.9
		118.1	187.0	36.8	138.1	14.5	102.4	187.0	138.1	45.2	25.9	95.9	187.0	48.7	138.1	30.6
		104.8	172.9	39.4	137.5	23.7	100.5	172.9	137.5	41.9	26.9	98.7	172.9	42.9	137.5	28.2
		104.4	170.8	38.9	127.5	18.1	96.4	170.8	127.5	43.6	24.4	90.5	170.8	47.0	127.5	29.0
	(20,20)	191.7	293.4	34.6	209.6	8.5	175.2	293.4	209.6	40.3	16.4	168.6	293.4	42.5	209.6	19.6
		224.1	319.2	29.8	220.3	-1.7	193.6	319.2	220.3	39.3	12.1	185.3	319.2	41.9	220.3	15.9
		213.0	313.1	32.0	208.1	-2.4	187.8	313.1	208.1	40.0	9.8	185.2	313.1	40.8	208.1	11.0
		257.2	291.6	11.8	197.9	-30.0	228.1	291.6	197.9	21.8	-15.3	210.5	291.6	27.8	197.9	-6.4
		226.8	286.6	20.9	204.2	-11.1	208.4	286.6	204.2	27.3	-2.0	194.7	286.6	32.1	204.2	4.7
	(20,30)	289.1	454.9	36.4	287.3	-0.6	262.6	454.9	287.3	42.3	8.6	249.6	454.9	45.1	287.3	13.1
		294.0	453.0	35.1	287.3	-2.3	272.2	453.0	287.3	39.9	5.2	256.1	453.0	43.5	287.3	10.9
		348.5	414.3	15.9	263.9	-32.1	314.8	414.3	263.9	24.0	-19.3	297.4	414.3	28.2	263.9	-12.7
		309.0	432.9	28.6	283.5	-9.0	279.3	432.9	283.5	35.5	1.5	262.3	432.9	39.4	283.5	7.5
		309.1	423.3	27.0	281.5	-9.8	276.1	423.3	281.5	34.8	1.9	258.3	423.3	39.0	281.5	8.2
2	(10,10)	125.4	220.7	43.2	167.7	25.2	117.7	220.7	167.7	46.6	29.8	110.9	220.7	49.7	167.7	33.9
		109.7	224.8	51.2	169.3	35.2	103.4	224.8	169.3	54.0	38.9	102.9	224.8	54.2	169.3	39.2
		137.1	249.3	45.0	184.2	25.5	121.4	249.3	184.2	51.3	34.1	114.9	249.3	53.9	184.2	37.6
		123.8	230.6	46.3	183.3	32.4	120.6	230.6	183.3	47.7	34.2	117.7	230.6	48.9	183.3	35.8
		123.4	227.8	45.8	170.0	27.4	115.9	227.8	170.0	49.1	31.8	109.5	227.8	51.9	170.0	35.6
	(20,20)	226.7	391.2	42.0	279.5	18.9	210.2	391.2	279.5	46.3	24.8	203.5	391.2	48.0	279.5	27.2
		259.1	425.5	39.1	293.8	11.8	228.6	425.5	293.8	46.3	22.2	220.3	425.5	48.2	293.8	25.0
		249.1	417.4	40.3	277.4	10.2	225.0	417.4	277.4	46.1	18.9	220.8	417.4	47.1	277.4	20.4
		292.1	388.9	24.9	263.8	-10.7	261.8	388.9	263.8	32.7	0.8	245.5	388.9	36.9	263.8	7.0
		261.8	382.2	31.5	272.3	3.9	243.4	382.2	272.3	36.3	10.6	229.8	382.2	39.9	272.3	15.6
	(20,30)	344.3	606.6	43.2	383.1	10.1	317.6	606.6	383.1	47.6	17.1	304.5	606.6	49.8	383.1	20.5
		348.9	604.1	43.0	383.0	10.1	327.2	604.1	383.0	45.8	14.6	310.8	604.1	48.5	383.0	18.9
		399.1	552.3	27.7	351.9	-13.4	365.3	552.3	351.9	33.9	-3.8	349.1	552.3	36.8	351.9	0.8
		363.7	577.2	37.0	377.9	3.8	334.3	577.2	377.9	42.1	11.5	317.3	577.2	45.0	377.9	16.1
		356.5	564.4	36.8	375.3	5.0	321.1	564.4	375.3	43.1	14.5	303.1	564.4	46.3	375.3	19.2