2022

Maximum Profit Facility Location and Dynamic Resource Allocation for Instant Delivery Logistics

Darshan Chauhan
Portland State University, drc9@pdx.edu

Avinash Unnikrishnan
Portland State University, uavinash@pdx.edu

Stephen D. Boyles
The University of Texas at Austin, sboyles@austin.utexas.edu

Citation Details

Published as: Chauhan, D.R., Unnikrishnan, A., Boyles, S.D., (2022), “Maximum Profit Facility Location and Dynamic Resource Allocation for Instant Delivery Logistics", Transportation Research Record. DOI: 10.1177/03611981221082574

Maximum Profit Facility Location and Dynamic Resource Allocation for Instant Delivery Logistics

Darshan R. Chauhan
Graduate Research Assistant, Department of Civil and Environmental Engineering, Portland State University, Portland, OR 97201
Email: drc9@pdx.edu (Corresponding Author)

Avinash Unnikrishnan
Professor, Department of Civil and Environmental Engineering, Portland State University, Portland, OR 97201
Email: uavinash@pdx.edu

Stephen D. Boyles
Associate Professor, Department of Civil, Architectural, and Environmental Engineering, The University of Texas at Austin, Austin, TX 78712
Email: sboyles@mail.utexas.edu

Cite as:
Chauhan, D.R., Unnikrishnan, A., Boyles, S.D., (2022), “Maximum Profit Facility Location and Dynamic Resource Allocation for Instant Delivery Logistics", Transportation Research Record.
DOI: 10.1177/03611981221082574

Word Count: 7653 words + 1 table(s) × 250 = 7903 words

Submission Date: March 24, 2022

ABSTRACT

Increasing e-commerce activity, competition for shorter delivery times, and innovations in transportation technologies have pushed the industry towards instant delivery logistics. In this paper, we study a facility location and online demand allocation problem applicable for a logistics company expanding to instant delivery service using unmanned aerial vehicles (UAV) or drones. The problem is decomposed into two stages. During the planning stage, the facilities are located, and product and battery capacity are allocated. During the operational stage, the customers place orders dynamically and real-time demand allocation decisions are made. We explore a multi-armed bandit framework for maximizing the cumulative reward realized by the logistics company subject to various capacity constraints and compare it with other strategies. The multi-armed bandit (MAB) framework provides about 7% more rewards than the second-best strategy when tested on standard test instances. A case study based in Portland Metro Area showed that MAB can outperform the second-best strategy by more than 20%.

INTRODUCTION

E-commerce usage has become ever-ubiquitous now, especially due to social isolation requirements during the COVID-19 pandemic. A shift towards digital shopping has resulted in double-digit retail e-commerce growth rates (32.1% growth from 2019 Q4 to 2020 Q4) compared to single-digit in total retail growth (6.9% growth from 2019 Q4 to 2020 Q4) (*1*). In recent years, the delivery time thresholds for online purchases have also become intensive, with various e-commerce platforms providing same-day delivery options. Now, even options for 2-hour (Walmart, Amazon Prime Now, Walmart, Space NK) and 1-hour delivery (Instacart Express, Shipt, Alibaba Fresh Hema, buymie.eu) exist, with the industry gearing towards an instant (30-minute or better) delivery goal (Amazon Prime Air, Getir, Wolt). One of the viable alternatives for instant delivery right now is a UAV/drone. With the numerous large corporations including Amazon, Walmart, UPS, DHL, and Kroger heavily investing in drone technology, the growth in this sector has surpassed previous forecasts (*2*). The higher operational speed and better cost-effectiveness of drones compared to traditional ground vehicles (*3*) would be beneficial as extant logistics systems are stressed with increased demand from e-commerce growth.

This work delves into facility location and resource allocation for including instant delivery logistics into a company's operations. We assume that the instant deliveries (or, time-sensitive deliveries) are fulfilled through a battery-operated drone. While the non-time-sensitive orders can either be fulfilled by a drone from the located facilities or a truck from the central warehouse. The system consists of a set of facilities that can act as both "dark" stores (or micro fulfillment centers), and drone operations sites. During the planning stage, the facilities are located and resources (product and battery capacity) are allocated such that it maximizes the total profit based on the deterministic information available. Once the facilities are set up, during the operational stage, the orders are received and real-time decisions are made regarding which facility and mode of transport would be used for fulfillment on an order-wise basis. Therefore, the goal in the operational stage is the adaptive learning of allocation of each order to maximize the cumulative profits while respecting resource capacity constraints.

Powell (*4*) summarizes the work done by various communities on stochastic optimization. Focusing on online sequential decision-making communities, Markov decision processes, Q-learning, and approximate dynamic programming are formed based on state transition functions like Bellman's equation. These methods learn over time through multiple iterations over these states. In our problem, we model consumption of non-replenishable resources over time, and therefore, a state of the system does not recur. Additionally, the above methods use maximization of terminal reward as the objective function which is not the case for our problem. These nuances make the above approaches not suitable for our problem.

Multi-armed bandits is an online decision-making framework that maximizes the cumulative reward over the learning period (*4*). Multi-armed bandits can also be equipped with "context" that provide information available before making decisions, and "knapsacks" that can account for globalized resource consumption associated with the decisions (*5*). The above characteristics make multi-armed bandits a suitable approach for our problem.

The key contributions of this study are: (i) formulating the instant delivery logistics problem as a two-stage problem – offline facility location with online resource allocation; (ii) while most of the previous logistics research has focused on time-aggregated dynamic resource allocation (*6*), we consider dynamic resource allocation at an order-level, which could be potentially beneficial for reducing delivery times (because of no lag in decision-making); and (iii) exploring a multi-armed bandit approach to effectively learn how to allocate orders to facilities in real-time and maximize the cumulative profits, and comparing it with other strategies.

The rest of the paper is organized as follows: the next section covers the relevant literature spanning the fields of facility location for stochastic demand, and dynamic resource allocation problems. Next, the problem description and formulation along with the dynamic resource allocation strategies are discussed. Later, computational experiments are conducted on test datasets. The final section concludes the work and provides avenues for further research.

LITERATURE REVIEW

In this section, we focus on primarily on facility location problems and dynamic resource allocation.

Facility Location Problems

Facility location has been one of the classical Operations Research problems and is one of the first prominent decisions that impact tactical and operational strategies for all organizations. An extensive number of books as well as review articles have been dedicated to facility location research: (*7–9*). Further, application-specific facility location reviews for humanitarian relief (*10*), healthcare and emergency location (*11*), and urban-based applications (*12*) are also available.

Mukundan and Daskin (*13*) is one of the earliest works to explicitly consider maximizing profit (others implicitly considered maximizing profit as an alternative to minimizing cost). They consider joint location and sizing problems while considering cover-based constraints for facilities based on their size. Profit is defined as the difference between revenue and cost. We extend the problem considered by Mukundan and Daskin (*13*) in two ways: firstly, by considering a continuum of facility size. This is achieved by converting costs in the objective functions to budget constraints. Therefore, the objective function in our work consists of only revenue-based terms. Secondly, we enforce capacity constraints derived from product and battery capacity allocations. Further, our problem does not consider coverage as a function of investment level as the facilities considered here are “dark-stores" which only cater to internet-based orders. The range of the facility is determined by explicitly modeling energy consumption in battery-operated drones.

Ambulance location literature is rich in two-stage facility location models where ambulances are located offline and their allocations to demand points (and ambulance relocation) are made in real-time (*14, 15*). For the real-time allocation, either offline policies are used as in Gendreau et al. (*14*), or adaptive online policies using methods like approximate dynamic programming are developed as in Schmid (*15*). The above works model allocation decision for ambulances to a request and then, their relocation decisions. The unavailability of ambulances for a request due to being busy or excessive travel time is modeled, but the researchers do not consider the modeling of non-replenishable resource consumption (like cost budgets). In this work, facility location decisions are made offline and we develop an adaptive online policy using a multi-armed bandits approach for allocation of requests to appropriate fulfillment facility. Additionally, the unavailability of drones (due to range considerations) and trucks (for instant fulfillment requests) is modeled, and non-replenishable resource consumption related to routing costs and product capacity at individual facilities is considered.

Dynamic Resource Allocation

Resource allocation problems are widely observed: assigning a vehicle to a demand point in vehicle routing problems (VRP), a personnel to a job, etc. In the context of VRP, Bektaş et al. (*16*) concisely defines dynamic problems as problems in which information is revealed gradually over time, rather than all at once (static problems). Here, we use the same definition for resource allocation problems, and use the words "dynamic" and "online" interchangeably.

In operations research, dynamic resource allocation problems are generally tackled by formulating them as multi-stage stochastic programs (MSSP) (*17*). Several AI-based techniques have also been explored for online resource allocation problems which include Q-learning (*18, 19*), multi-armed bandits (*20, 21*), online algorithms (*22*), and approximate dynamic programming (ADP) (*23*). Recently, Powell (*4*) summarized the commonalities of various communities of stochastic and/or dynamic optimization and noted that adaptive learning algorithms based on dynamic programming (Q-learning, ADP, stochastic dual dynamic programming for MSSP) fall under the category of state-dependent problems with a terminal reward objective. Additionally, they observed that contextual bandits lend themselves well to several state-dependent problems with a cumulative reward objective. In this study, we explore a multi-armed bandit framework for dynamic resource allocation. Badanidiyuru et al. (*21*) first proposed a multi-armed framework with universal budget constraints, which explicitly accommodates budget constraints in decision-making. Specifically, we use linear contextual bandits with knapsacks proposed by Agrawal and Devanur (*24*) and extrapolate results to accommodate restricted "arm" availability arising from drone range constraints.

Focusing on delivery-related applications, Mallick et al. (*25*) discuss recommendation systems for a carrier in truckload freight exchange marketplaces with constraints on arriving back at the origin within the planning period. Guo et al. (*17*) studied drone-truck combined instant delivery logistics (90-minute grocery delivery) and proposed to solve it as a multi-stage stochastic program. Here, all the orders in a 30-minute interval were considered for allocation and routing. The truck and drone resources were available at each time interval for allocation (i.e., they are replenished). Similar assumptions have been made in humanitarian logistics applications (*18, 23*), where a constant amount of resources to be distributed (food, medical kits) are made available every time period. However, in our study, a total product inventory and battery capacity are allocated to each open facility for the planning period and these are not replenished during the operational stage. Additionally, the above research addresses catering demand in each time interval. Our study differs by allocating demand at an order level, which would improve delivery time performance for instant orders, and is also adequate considering the limited payload capacity of drones (*26*).

Availability-related constraints are adequately tackled in dynamic fleet management literature (*27*), where drivers available at any time period are varying. However in this study, we assume that a sufficient amount of drones are placed at facilities and trucks are central warehouses that vehicle-related availability constraints and congestion effects can be safely ignored, to emphasize more on the allocation of facility and mode of delivery to a request. Shavarani et al. (*28*) studied a congested facility location problem for drone delivery with a case study implementation in San Francisco. The results show that for the system with 30 minute wait time, drone acquisition costs only accounted for about 12% of recharging infrastructure setup and operation costs. We consider that the drone acquisition costs are already considered in the drone operations facility opening and operation costs.

PROBLEM DESCRIPTION

This section presents the maximum profit facility location problem for online demand satisfaction which is applicable for a logistics company employing UAVs/drones for last-mile delivery. We propose a two-stage framework wherein the planning stage, we locate facilities and allocate resources (product and battery capacity) to them with an a priori estimate of consumer demand. With the facilities located, and resources allocated, in the operational stage, the consumers place orders dynamically and we use multi-armed bandits for making real-time decisions to maximize the cumulative revenue while consuming resources. Note that the resources allocated at the end of the planning stage are not replenished in the operational stage. We consider two types of consumers: time-sensitive and regular. For time-sensitive customers, the demand must be met instantly, else, the demand is lost. For the regular customers, the demand must be fulfilled but instant deliveries are not mandatory.

Stage 1: Planning Stage

The online demand satisfaction problem is defined over a finite planning horizon. There are two modes of delivery available: drone delivery and traditional ground-based delivery. The drone-based deliveries are achieved by locating facilities that act as both products holding storerooms as well as drone launching stations. The ground-based deliveries are fulfilled using trucks located at a central warehouse. We assume that there are enough drones/trucks at each facility that congestion effects can be neglected. The time-sensitive orders are fulfilled instantly (for example, 30 minutes or less), and the regular orders are fulfilled within a predetermined level of service (example, 1-day delivery or 2-day delivery). The time-sensitive orders are, therefore, assumed to be satisfied only using drones, while the regular orders can be delivered by either drones or trucks.

Let G denote the set of demand points. During the planning stage, each demand point can be considered as a small geographical region (for example, ZIP code, Census tract, etc.). Let n_g^S and n_g^R denote the anticipated number of time-sensitive and regular orders from point $g \in G$, respectively. Let F_g denote the set of all deliveries to point $g \in G$, i.e. $F_g = \{1, \dots, n_g^S, n_g^S + 1, \dots, n_g^S + n_g^R\}$. During the facility location (or, planning) stage, we do not take temporal aspects into consideration, and therefore, can consider both types of deliveries together.

Let o_{gf} denote the order weight of f^{th} $(\in F_g)$ order from demand point $g \in G$. We assume each order can weigh up to o_{max}. During the planning stage, o_{gf} could take the form of using representative weights for different weight categories (therefore, obtaining the number of orders for different weight categories), and during the operational stage, the demand can assume the continuum of values up to o_{max}. The parameters c_{gf}^D and c_{gf}^T denote the estimated profit for satisfying the f^{th} order of demand point $g \in G$ using drones and trucks, respectively.

The drone-based deliveries are carried out from potential facility locations spread through the service area. The set of all potential facility locations is represented by H. A maximum of p number of facilities can be opened. This parameter is prescribed considering leasing costs, recharging infrastructure setup costs, drone acquisition costs, and other related costs. A total of α amount of commodity and β amount of battery capacity (for drones) are available for distribution among open facilities. If a facility is opened, then a minimum of α_{min} and β_{min} amount of product and battery capacity must be allocated to it.

We assume that the drones only make one-to-one deliveries (from facility to demand point and back), as drones would be allocated at an order level during the operational stage. Let, b_{ghf}

be the battery consumption order to travel from a facility $h \in H$ to a point $g \in G$ and back while delivering the f^{th} order, and B_{drone} be the battery capacity of the drone. To be on a bit on the conservative side, we simplify b_{ghf} to b_{gh} (the battery consumption obtained using order weight of o_{max}). The set $G_h = \{g \in G : b_{gh} \leq B_{drone}\}$ describes the set of demand points that are accessible from a facility $h \in H$.

The truck-based deliveries are satisfied from a central warehouse. During the planning horizon, there are a maximum of $|M|$ truck trips available (M denotes the set of truck trips) which are determined based on the predetermined level of service, for example, an average of 2 truck trips per day over the planning period of 300 days yields $|M| = 600$. The capacity of each truck (in the number of packages that can be delivered in each trip) is B_{truck} which is determined based on actual truck capacity, operator working hours constraints, etc. Each truck trip can be considered as a traveling salesman problem, and therefore, truck trip routing costs can be estimated using continuous approximation (*29*). For a warehouse located 'd_0' distance away from the center of the service area (of size A), the continuous approximation trip cost is given as:

$$m^{th} \text{ Truck Trip Cost} = k_1 n_m + k_2 d_0 \left(\mathbf{1}_{n_m > 0}\right) + k_3 \sqrt{n_m A}$$

where, n_m is the number of customers serviced on the $m^{th} (\in M)$ trip, A is the size of service area, and k_1, k_2, k_3 are the proportionality constants. The term $(\mathbf{1}_{n_m > 0})$ is the Heaviside step function with value 1 when $n_m > 0$, and 0 otherwise. The first term represents time costs (related to customer service times), and the second and third terms combined are distance related costs.

As the truck trip costs are concave, the best allocation is by consolidating as many assignments on a trip as possible. It is given that each truck can serve up to B_{truck} customers per trip, and a total routing cost budget is $B_{routing}$. Then, the best allocation occurs when the first m^* truck trips each serves B_{truck} customers, and the remaining routing budget is utilized on the $(m^* + 1)^{th}$ truck trip. The value of m^* is given as:

$$m^* := \left\lfloor \frac{B_{routing}}{k_1 B_{truck} + k_2 d_0 + k_3 \sqrt{B_{truck} A}} \right\rfloor$$

Therefore, instead of incorporating a non-linear non-convex routing cost constraint, we can incorporate a simpler (and slightly conservative) linear constraint by limiting the total number of truck-based deliveries to $\omega = (\min\{|M|, m^*\} \cdot B_{truck})$.

Now, we discuss the decision variables for the planning stage optimization problem. The binary variable y_h is 1 if a facility is opened at location $h \in H$. Let the variables u_h and z_h denote the product and battery capacity allocated to facility $h \in H$, respectively. The binary variable x_{hgf} is 1 if f^{th} $(\in F_g)$ order of point $g \in G$ is met by facility $h \in H$ using drone delivery, and the binary variable w_{gf} be 1 if f^{th} $(\in F_g)$ order of point $g \in G$ is met using truck delivery. Finally, the facility location problem can be described as:

$$\max_{u,w,x,y,z} \quad \sum_{h \in H} \sum_{g \in G} \sum_{f \in F_g} c^D_{gf} x_{hgf} + \sum_{g \in G} \sum_{f \in F_g} c^T_{gf} w_{gf} \tag{1}$$

$$\sum_{h \in H} y_h \leq p \tag{2}$$

$$\sum_{h \in H} u_h \leq \alpha \tag{3}$$

$$u_h \leq \alpha y_h \quad \forall\ h \in H \tag{4}$$

$$u_h \geq \alpha_{min} y_h \quad \forall\ h \in H \tag{5}$$

$$\sum_{g \in G_h} \sum_{f \in F_g} o_{gf} x_{hgf} \leq u_h \quad \forall\ h \in H \tag{6}$$

$$\sum_{h \in H} z_h \leq \beta \tag{7}$$

$$z_h \leq \beta \cdot y_h \quad \forall\ h \in H \tag{8}$$

$$z_h \geq \beta_{min} y_h \quad \forall\ h \in H \tag{9}$$

$$\sum_{g \in G_h} \sum_{f \in F_g} b_{gh} x_{hgf} \leq z_h \quad \forall\ h \in H \tag{10}$$

$$\sum_{h \in H} x_{hgf} + w_{gf} \leq 1 \quad \forall\ f \in F_g, g \in G \tag{11}$$

$$\sum_{g \in G} \sum_{f \in F_g} w_{gf} \leq \omega \tag{12}$$

$$x_{hgf} \in \{0,1\} \quad \forall\ f \in F_g, g \in G, h \in H \tag{13}$$

$$w_{gf} \in \{0,1\} \quad \forall\ f \in F_g, g \in G \tag{14}$$

$$y_h \in \{0,1\} \quad \forall\ h \in H \tag{15}$$

$$u_h, z_h \geq 0 \quad \forall\ h \in H \tag{16}$$

The objective function 1 aims to maximize the cumulative profit achieved by the delivery system. Equation 2 constrains the number of open facilities to a maximum of p. Equations 3 and 7 ensure that a total amount of product and battery capacity allocated is less than α and β (the overall budgets), respectively. Equations 4 and 5 ensure that product is only inventoried at open facilities, and that at least a minimum of α_{min} amount of product is inventoried when a facility is opened. Equations 8 and 9 enforce similar constraints to battery capacity, i.e., battery capacity can only be allocated to open facilities and at least a minimum of β_{min} amount of battery capacity is allocated at an open facility. Equation 6 ensures that no more demand than the product inventory at a facility is met. Similarly, equation 10 constrains the battery consumption to be less than the battery capacity of the facility. Note that equations 6 and 10 take drone delivery range into consideration. Equation 11 ensures that a particular order can be satisfied by at most one facility using a drone or by using truck-based delivery. Constraint 12 limits the total number of truck based deliveries to ω. Equations 13-16 are variable definitions.

After finding an optimal solution, it is modified by allocating the remaining slack in product (equation 3) and battery capacity (equation 7). The slack is allocated such that it maximizes the minimum value of product and battery capacity at an open facility.

Stage 2: Operational Stage

At the end of stage 1, we have a solution for the planning stage problem, given by the tuple $(x^*, w^*, y^*, u^*, z^*)$. Let, the set of all opened facilities be represented by set $H' := \{h \in H : y^*_h = 1\}$. The product inventory and battery capacity located at an open facility $h \in H'$ are given by u^*_h and z^*_h, respectively, which would be used for drone-based deliveries. The regular (i.e., non-time-sensitive) orders can also be fulfilled through the central warehouse using truck-based delivery (a maximum of ω number of regular order deliveries can be fulfilled). Order fulfillment leads to consumption of these resources (product, battery, truck) which are not replenished during the entirety of the operational stage. The goal during the operational stage is to maximize cumulative profit by allocating orders arriving in an online manner to either one of the located facilities for drone delivery, or to the central warehouse for a truck delivery (only for regular orders), such that no more than available resources are utilized.

During the operational stage, uncertainties stem from various sources: the probability of an order being from demand location $g \in G$, the probability of an order being time-sensitive, the order weight distribution from demand location $g \in G$, the battery consumption distribution while using drones for delivering an order, and the profit distributions of time-sensitive and regular orders. However, in any market, these values cannot be known with absolute certainty and need to be learned by implementing field trials. We explore three strategies for this online operational stage problem: first, a multi-armed bandits-based approach; second, an allocation heuristic designed from the solution of the planning stage optimization problem; and third, two random choice heuristics based on the random choice among available options for an order. Based on the context of our problem, we use the words “profits” and “rewards” synonymously here.

Currently, there is no option of non-fulfillment of an order (possibly due to network congestion) and can be a part of future research. As a performance measurement here, we study the maximization of cumulative profit until the first resource constraint is violated considering the allocation of each order independent of its profit and resource consumption. Alternatively put, the episode ends at the first instance of a facility (or central warehouse) running out of a resource. Instant delivery logistics would typically be adopted for a relatively small geographical area (like a metropolitan area). In such cases, orders received after the first instance of exhaustion of resources could lead to denial of service based on geography, which cannot be the case for practical applications. Also, this stopping criterion would indicate the need for resources to be replenished for uninterrupted service.

Multi-armed bandits

Multi-armed bandits are a reinforcement learning framework wherein the agent learns by exploring given set of options (a.k.a. “arms") such that the cumulative reward achieved is maximized. Here, specifically, we use linear contextual bandits with knapsacks (linCBwK), proposed by Agrawal and Devanur (*24*). The linCBwK allows us to choose only from a subset of arms (as all open facilities are not available to each order placed) while accounting for constraints consisting of the product inventory and battery consumption budgets at each facility, and overall truck routing.

A linCBwK problem has five components to it. The first is the K number of arms or actions. Here, these actions represent options for each order: drone delivery from a located facility, or truck delivery from the central warehouse. Therefore, we have $K = |H'| + 1$ arms, and $[K] := \{h \, \forall \, h \in H'\} \cup \{truck\}$ is the set of all arms.

The second is time horizon or total number of decision-making events T. Here, each time/event $t \in \{1, 2, \ldots, T\}$ represents an order placed in real-time by demand point $g_t \in G$ (assumed i.i.d. to unknown distribution $\mathscr{D}^g$, abbreviated as $g_t \overset{iid}{\sim} \mathscr{D}^g$). Let, λ_t (derived from $\lambda_t | g_t \overset{iid}{\sim} \mathscr{D}^{\lambda}_{g_t}$) be 1 if the order is time-sensitive and requires instant delivery, and 0, otherwise. Note that the demand points outside the drone-based coverage region can only place regular orders. Then, for each $h \in H'$, taking the current environmental factors into account, we observe the battery consumption, $b^t_{g_t h}$ ($b^t_{g_t h} | g_t \overset{iid}{\sim} \mathscr{D}^b_{g_t h}$) for all $h \in H'$. All of the above information is available before making the allocation decision for event t.

The third is the context. Context represents all information that we have prior to making a decision. At each event t, we observe an m-dimensional context vector for each arm $a \in [K]$, represented by $\mathbf{x}_t(a) \in [0,1]^m$. Let the context matrix $X_t := \{\mathbf{x}_t(a) \forall a \in [K]\} \in [0,1]^{m \times K}$. For our case, we observe a $m = K$ dimensional context for each arm. The $K \times K$ diagonal context matrix is constructed as:

$$X_t[h,h] := \begin{cases} 1 & \text{if } b^t_{g_t h} \leq B_{drone} \\ 0 & \text{otherwise} \end{cases}, \quad \forall\ h \in H'$$

$$X_t[truck, truck] := \begin{cases} 1 & \text{if } \lambda_t = 0 \\ 0 & \text{otherwise} \end{cases}$$

Agrawal and Devanur (*24*) state that when the context matrix is a K-dimensional identity matrix, linCBwK emulates the bandits with knapsacks (BwK) problem (*21*). We extrapolate this result to consider a BwK problem with restricted arm availability. The above definition of context only allows arms with context equal to 1 to be available for selection. While defining context, we ensure that the *truck* arm is only available for regular deliveries. Additionally, we consider that the demand points outside the drone-based coverage can only place regular orders. The above definition allows for the availability of at least one arm for selection by the algorithm.

The fourth component of linCBwK problem is reward. At time t, a scalar reward $r_t(a_t) \in [0,1]$ is realized after playing action $a_t \in [K]$. Without loss of generality, we assume that the reward for fulfilling a time-sensitive delivery is $c^S \in (0,1]$, and a regular delivery is $c^R \in [0, c^S)$ irrespective of which action is chosen. This assumption can be relaxed to model rewards that are a function of the ordering demand point g_t and the action a_t chosen.

The fifth component of linCBwK is knapsacks constraints, or globalized budget constraints. For our problem, there are $d = (2 \cdot |H'| + 1)$ universal knapsack constraints. The first $|H'|$ constraints represent the product consumption at facility $h \in H'$ with budgets u^*_h, the second $|H'|$ constraints represent the battery consumption at facility $h \in H'$ with budgets z^*_h, and the last knapsack constrains the total number of truck deliveries to a budget ω (as described in Stage 1). If at time t, the *truck* arm is chosen, then, $\mathbf{0}$ amount of product (U_h), $\mathbf{0}$ amount of battery (B_h) resources, and 1 unit of truck delivery resources are used. If the truck-delivery arm is not chosen, then let h_t denote the chosen facility for fulfillment of demand. When h_t is chosen, $(o^t_{g_t} \cdot \mathbf{e}_{h_t})$ amount of product, $(b^t_{g_t h_t} \cdot \mathbf{e}_{h_t})$ amount of battery resource, and 0 unit of truck delivery resources are consumed. Here, $\mathbf{e}_{h_t}$ is a $|H'| \times 1$ matrix with value 1 for the row where $h = h_t$, and 0, otherwise. Let I^{truck}_t be 1 if the truck-delivery arm is chosen at time t, and 0, otherwise.

The bandit optimization problem that we are tackling here is given as:

$$\max_{\mathbf{e}, I^{truck}} \quad \sum_{t=1}^{T} \left[\left\{ c^S \lambda_t + c^R (1-\lambda_t) \right\} (1 - I^{truck}_t) + c^R (1-\lambda_t) I^{truck}_t \right] \tag{17}$$

$$\text{s.to.} \quad \sum_{t=1}^{T} o^t_{g_t} \cdot \mathbf{e}_{h_t} \cdot (1 - I^{truck}_t) \leq u^*_h \quad \forall\ h \in H' \tag{18}$$

$$\sum_{t=1}^{T} b^t_{g_t h_t} \cdot \mathbf{e}_{h_t} \cdot (1 - I^{truck}_t) \leq z^*_h \quad \forall\ h \in H' \tag{19}$$

$$\sum_{t=1}^{T} I_t^{truck} \leq \omega \tag{20}$$

At time t, upon selection of arm a_t, let $\mathbf{v}_t(a_t)$ be the d-dimensional resource consumption vector. As a reminder, we allow only payloads up to o_{max}. Also note that for chosen arm, the value of $b^t_{g_t h_t}$ is always less than or equal to B_{drone}. Therefore, we can use the above values to normalize resource consumption vector $\mathbf{v}_t(a_t)$ in the range [0,1], a requirement to implement linCBwK. The first set of transformations are given as:

Transformations I:

Product consumption knapsacks: $o^t_{g_t} \leftarrow \dfrac{o^t_{g_t}}{o_{max}}$ $\qquad u^*_h \leftarrow \dfrac{u^*_h}{o_{max}}$

Battery consumption knapsacks: $b^t_{g_t h_t} \leftarrow \dfrac{b^t_{g_t h_t}}{B_{drone}}$ $\qquad z^*_h \leftarrow \dfrac{z^*_h}{B_{drone}}$

The other required transformation for linCBwK is a uniform value of budget for each knapsack constraint. Therefore, we scale each knapsack so that its budget to the lowest value after the above transformations. The new budget, B, is given as:

$$B = \min\big\{\min\{u^*_h : h \in H'\}, \min\{z^*_h : h \in H'\}, \omega\big\}$$

The transformations to make the budget the same for all knapsack constraints are:

Transformations II:

Product consumption knapsacks: $o^t_{g_t} \leftarrow \dfrac{B}{u^*_h} o^t_{g_t};$ $\qquad u^*_h \leftarrow B$

Battery consumption knapsacks: $b^t_{g_t h_t} \leftarrow \dfrac{B}{z^*_h} b^t_{g_t h_t};$ $\qquad z^*_h \leftarrow B$

Truck delivery knapsack: $I^{truck}_t \leftarrow \dfrac{B}{\omega} I^{truck}_t;$ $\qquad \omega \leftarrow B$

We make the following two assumptions about context, rewards, and resource consumption vectors in linCBwK (*24*):

- In every round t, the tuple $\{\mathbf{x}_t(a), r_t(a), \mathbf{v}_t(a)\}_{a=1}^K$ is generated from an unknown distribution $\mathscr{D}$, independent of everything in previous rounds. The procedure used for generating contexts, rewards, and resource consumption for our instant delivery logistics problem satisfies this assumption.
- There exists an unknown vector $\mu_* \in [0,1]^{m\times 1}$ and a matrix $W_* \in [0,1]^{m\times d}$ such that for every arm a, given contexts $\mathbf{x}_t(a)$, and history H_{t-1} before time t,

$$\mathbb{E}[r_t(a)|x_t(a), H_{t-1}] = \mu_*^{\mathsf{T}} x_t(a), \qquad \mathbb{E}[\mathbf{v}_t(a)|x_t(a), H_{t-1}] = W_*^{\mathsf{T}} x_t(a) \tag{21}$$

The decision-making flow of the linCBwK algorithm consists of five major steps: observing the context matrix X_t, obtaining optimistic estimates of μ_* and W_* using the l_2-regularized norms of previously observed values of rewards and resource consumption, arm selection using an expected reward penalized with expected resource consumption, realizing the values of reward and resource consumption for the selected arm, and finally, updating the penalty weights using multiplicative weight update (MWU) algorithm (*24*). The detailed algorithm is in Algorithm 1. Like all multi-armed bandit algorithms, the performance is measured by the complexity of the cumulative regret. For linCBwK, the regret is measured from the optimal static policy (*24*), obtained from solving a static stochastic optimization problem.

Algorithm 1 Algorithm for linCBwK

```
Input parameters: $B$, $T_0$, $T$, $(1-\delta)$ confidence level, MWU algorithm parameter $\varepsilon$
Compute $Z$ which satisfies assumptions presented in Agrawal and Devanur (24)
Initialize $t = 1$, $\theta_{1,j} = \frac{1}{1+d}$, $\forall\, j \in \{1,2,\ldots,d\}$, $radius_t = \sqrt{m \log\left(\frac{d+tmd}{\delta}\right)} + \sqrt{m}$
Initialize $B' = B - T_0$, $T' = T - T_0$
while $t \leq T'$ do
    Observe context $X_t$
    For every $a \in [K]$, compute $\tilde{\mu}_t(a)$ and $\tilde{W}_t(a)$ (the optimistic estimates of $\mu_*$ and $W_*$) as:
        $\tilde{\mu}_t(a) := \arg\max_{\mu \in C_{t,0}} \mathbf{x}_t(a)^\intercal \mu, \quad \text{where, } \hat{\mu}_t := M_t^{-1} \sum_{i=1}^{t-1} \mathbf{x}_i(a_i) r_i(a_i)^\intercal \qquad (22)$
    where, $C_{t,0} := \left\{\mu \in \mathbb{R}^{m\times 1} : \|\mu - \hat{\mu}_t\|_{M_t} \leq radius_t\right\}$
        $\tilde{W}_t(a) := \arg\min_{W \in \mathscr{G}_t} \mathbf{x}_t(a)^\intercal W \theta_t, \quad \text{where, } \hat{W}_t := M_t^{-1} \sum_{i=1}^{t-1} \mathbf{x}_i(a_i) \mathbf{v}_i(a_i)^\intercal \qquad (23)$
     where, $\mathscr{G}_t := \left\{\mathbb{R}^{m\times d} : \mathbf{w}_j \in C_{t,j}\right\}$ and $C_{t,j} := \left\{\mathbf{w} \in \mathbb{R}^{m\times 1} : \|\mathbf{w} - \hat{\mathbf{w}}_{tj}\|_{M_t} \leq radius_t\right\}$
    Play the arm $a_t := \arg\max_{a\in[K]} \mathbf{x}_t(a)^\intercal \left(\tilde{\mu}_t(a) - Z\tilde{W}_t(a)\theta_t\right)$
    Observe reward $r_t(a_t)$ and resource consumption $\mathbf{v}_t(a_t)$
    If for some $j \in \{1,\ldots,d\}$, $\sum_{t'\leq t} \mathbf{v}_{t'}(a_{t'}) \cdot \mathbf{e}_j \geq B$, then EXIT. (Note: $\mathbf{e}_j$ is a $d \times 1$ matrix with value 1 for $j^{th}$ row, and 0, otherwise)
    Update $\theta_{t+1}$ using MWU algorithm, and with $g_t(\theta_t) := \theta_t \cdot \left(\mathbf{v}_t(a_t) - \frac{B}{T}\mathbf{1}\right)$, as:
    $\theta_{t+1,j} = \frac{w_{t,j}}{1+\sum_j w_{t,j}}, \text{ where } w_{t,j} = \begin{cases} w_{t-1,j}(1+\varepsilon)^{g_{t,j}} & \text{if } g_{t,j} > 0, \\ w_{t-1,j}(1-\varepsilon)^{-g_{t,j}} & \text{if } g_{t,j} \leq 0. \end{cases}, \quad \forall\, j \in \{1,2,\ldots,d\} \qquad (24)$
    $t += 1$
end while
```

Planning Stage Optimization Allocation (PSOA) Heuristic

The planning stage optimization allocation (PSOA) heuristic uses the insights obtained from the solution of the optimization problem solved in Stage 1 to derive a policy. Particularly, $2|G|$ different policies are derived depending upon the demand point $g \in G$ placing an order, and the order being time-sensitive or not. The policies are given as:

For all $g \in G$

$$p^{PSOA}_{g,\lambda}(h) = \begin{cases} \dfrac{\sum_{f\in F} x^*_{hgf}}{\sum_{a\in H'} \sum_{f\in F} x^*_{agf}} & \text{if, } \sum\limits_{a\in H'} \sum\limits_{f\in F} x^*_{agf} > 0,\ \lambda = 1 \\ \dfrac{\sum_{f\in F} \frac{n^R_g}{n^S_g+n^R_g} x^*_{hgf}}{\sum_{a\in H'} \sum_{f\in F} \frac{n^R_g}{n^S_g+n^R_g} x^*_{agf} + \sum_{f\in F} w^*_{gf}} & \text{if, } \sum\limits_{a\in H'} \sum\limits_{f\in F} x^*_{agf} > 0,\ \lambda = 0 \\ 0 & \text{otherwise} \end{cases}, \forall\, h \in H' \tag{25}$$

$$p^{PSOA}_{g,\lambda}(truck) = \begin{cases} 1 - \sum\limits_{h\in H'} p^{PSOA}_{g,\lambda}(h) & \text{if, } \lambda = 0 \\ 0 & \text{otherwise} \end{cases} \tag{26}$$

where, $\lambda = 1$ for time-sensitive deliveries, and $\lambda = 0$ for regular deliveries. Note that as the type of orders (time-sensitive or regular) are not differentiated in the Stage 1 optimization problem, some demand points which can be covered through drone-based deliveries may only be served using the truck delivery option. As a result, these demand point could place a time-sensitive order and the PSOA heuristic would not know what to do. In such cases, the PSOA heuristic collects a reward of 0, and consumes **0** units of all the resources (product, battery, and truck-delivery). The algorithm for PSOA is presented in Algorithm 2.

Algorithm 2 PSOA Heuristic

```
Input parameters $B' = B - T_0$, and $T' = T - T_0$.
Initialize $t = 1$
while $t \leq T'$ do
    Observe ordering demand point $g_t$, time-sensitivity of order $\lambda_t$, and context $X_t$.
    Select an arm $a_t \in [K]$ chosen randomly with probability of choosing $a_t$ is $p^{PSOA}_{g_t,\lambda_t}(a_t)$
    if $a_t \in H'$ and $X_t[a_t, a_t] = 1$ then
        Play the selected arm $a_t$
        Observe reward $r_t(a_t)$ and resource consumption $\mathbf{v}_t(a_t)$
    else
        if $\lambda_t = 1$ then
            Randomly select and play an arm $a_t$ such that $X_t[a_t, a_t] = 1$
            Observe reward $r_t(a_t)$ and resource consumption $\mathbf{v}_t(a_t)$
        else
            Play the $truck$ arm, i.e, $a_t = truck$
            Observe reward $r_t(a_t)$ and resource consumption $\mathbf{v}_t(a_t)$
        end if
    end if
    If for some $j \in \{1, \dots, d\}$, $\sum_{t' \leq t} \mathbf{v}_{t'}(a_{t'}) \cdot \mathbf{e}_j \geq B$, then EXIT. (Note: $\mathbf{e}_j$ is a $d \times 1$ matrix with
    value 1 for $j^{th}$ row, and 0, otherwise)
    $t += 1$
end while
```

Random Choice (RC) Heuristic

The random choice (RC) heuristic chooses one of the available options randomly in a weighted manner upon observing the context based on random choice among the available alternative at each time t. The nominal probability of choosing the “$truck$" arm for a regular order is $p^{RC}_{truck} = \frac{\omega}{\sum_{g \in G} n^R_g}$. The algorithm is presented in Algorithm 3.

Algorithm 3 RC Heuristic

```
Input parameters $B'$, and $T'$.
Initialize $t = 1$
while $t \leq T'$ do
    Observe ordering demand point $g_t$, time-sensitivity of order $\lambda_t$, and context $X_t$.
    Calculate the set of facilities available for drone deliveries, i.e., $H_{avail} := \{h \in H' \,|\, X_t[h,h] = 1\}$

    if $|H_{avail}| > 0$ then
        if $\lambda_t = 1$ then
            Play one of the available facility arms each with probability $\frac{1}{|H_{avail}|}$
        else
            Play the arm “$truck$" with probability $p^{RC}_{truck}$, and one of the available facility arms each
            with probability $\frac{1}{|H_{avail}|}(1 - p^{RC}_{truck})$
        end if
    else
        Play arm “$truck$"
    end if
    Observe reward $r_t(a_t)$ and resource consumption $\mathbf{v}_t(a_t)$
    If for some $j \in \{1, \ldots, d\}$, $\sum_{t' \leq t} \mathbf{v}_{t'}(a_{t'}) \cdot \mathbf{e}_j \geq B$, then EXIT. (Note: $\mathbf{e}_j$ is a $d \times 1$ matrix with
    value 1 for $j^{th}$ row, and 0, otherwise)
    $t += 1$
end while
```

Blind Random Choice (BRC) Heuristic

The blind random choice (BRC) heuristic works like the RC heuristics, except that it does not have access to even the input parameters of the problem (n^S, n^R, ω). Therefore, at any time t, the BRC heuristic chooses one of the available arms randomly in a unweighted manner (i.e., each of the available arms has an equal probability of being selected).

COMPUTATIONAL EXPERIMENTS

The analysis is conducted on standard p-median test instances, adopted from Osman and Christofides (*30*), each consisting of 50 locations that act both as demand points (represented by set G) and potential facility locations (represented by set H) on a randomly generated on a 100 × 100 grid (here, units are assumed to be kilometers). For the current planning period, the anticipated number of times-sensitive (n^S_g) and regular (n^R_g) deliveries are random integers in the interval [8,12] and [8,12], respectively. The estimated demand for each order is randomly selected from a discrete uniform distribution from 0.5 kg to 2.25 kg in the interval of 0.25 kg. Euclidean distances are used for distance computations. The battery consumption for a trip from facility $h \in H$ to demand point $g \in G$ and back is calculated as in Figliozzi (*31*) assuming a payload of $o_{max} = 5$ lbs (2.27 kg). The overall energy efficiency and the lift-to-drag ratio of the drone are 0.66 and 2.89, respectively. The battery capacity of the drone is 1410 Wh, and a maximum battery utilization factor of 0.8 is used. Thereby, the effective battery capacity of the drone (B_{drone}) is 1128 Wh. The values of parameters α, β, α_{min}, β_{min}, and p are chosen to be 2500, $800 \cdot B_{drone}$, 800, $350 \cdot B_{drone}$, and 3, respectively. Considering the truck routing budget, the maximum number of orders that can be fulfilled by truck delivery (i.e., ω) is determined to be 400. We do not consider congestion effects in the current study and assume that enough drones/trucks are available at each operational facility.

The solution of the planning stage problem determines the initial state of the operational stage problem (i.e. K, d, m, and B can be calculated). For bandit learning using linCBwK, the $(1-\delta)$ confidence interval for estimating unknown parameters is taken to be 95%. The total number of orders (T) is assumed to be 1000, and the initial learning iterations (T_0) is assumed to be $m\sqrt{T}$ (rounded to the nearest integer). The above value of T_0 ensures that the linCBwK algorithm maintains the regret bound provided by Agrawal and Devanur (*24*). The online learning parameter, ε, is assumed to be $\sqrt{(d+1)/T}$, as proposed by Agrawal and Devanur (*24*).

Unknown to the linCBwK algorithm, for the nominal case, we assume that the estimated values of time-sensitive and regular deliveries used in the planning stage are off by at most $\rho^S =$ 30% and $\rho^R = 10\%$ compared to the actual simulation values observed during the operational stage. Therefore, for the simulations, the probability of a demand point ordering and the order being time-sensitive is determined by calculating simulation values of time-sensitive ($\tilde{n}^S_g$) and regular ($\tilde{n}^R_g$) deliveries from demand point $g \in G$ is set to:

$$\tilde{n}^S_g \in Uniform\left[\frac{1}{1+\rho^S}n^S_g, \frac{1}{1-\rho^S}n^S_g\right], \quad \forall\, g \in G$$

$$\tilde{n}^R_g \in Uniform\left[\frac{1}{1+\rho^R}n^R_g, \frac{1}{1-\rho^R}n^R_g\right], \quad \forall\, g \in G$$

$$P(g_t = g) = \frac{\tilde{n}^S_g + \tilde{n}^R_g}{\sum_{g \in G} \tilde{n}^S_g + \tilde{n}^R_g}, \quad \forall\, g \in G$$

$$P(\lambda_{g_t} = 1 \,|\, g_t = g) = \begin{cases} \dfrac{\tilde{n}^S_g}{\tilde{n}^S_g + \tilde{n}^R_g} & ;\, g \text{ can be served using drones} \\ 0 & ;\, g \text{ cannot be served using drones} \end{cases}, \quad \forall\, g \in G$$

At time $t \in \{1, 2, \ldots, T\}$, the ordering demand point $g_t = g$ with probability $P(g_t = g)$, and the order is time-sensitive with probability $P(\lambda_{g_t} = 1 \,|\, g_t = g)$. Unknown to the algorithms, we define parameters $\phi^1_g, \phi^2_g \in Uniform(0.5, 5)\ \forall\, g \in G$. The demand ($o_{g_t}$) is randomly chosen in the interval $[0, o_{max}]$ from the beta distribution $o_{max} \cdot Beta(\phi^1_{g_t}, \phi^2_{g_t})$, where, o_{max} is the maximum weight of an order. The battery consumption at time t (i.e., $b^t_{g_t h}$), between a demand point $g_t \in G$ and facility $h \in H'$ is assumed to vary in the interval $[b_{gh} - \hat{b}_{gh}, b_{gh} + \hat{b}_{gh}]$, where, b_{gh} is the nominal battery consumption (used in the planning stage), and $\hat{b}_{gh}$ is the maximum variation is battery consumption. The value of $\hat{b}_{gh}$ is assumed to be an integer in the interval $[0.1b_{gh}, 0.3b_{gh}]$. Similar assumptions are made in Chauhan et al. (*32*).

During the simulations, each instance is run 10 times to account for randomness in demand, time sensitivity, order weight, and battery consumption generation. Table 1 shows the cumulative reward achieved. All instances opened 2 facilities for drone delivery and had a truck-delivery option for regular orders. The linCBwK provides the best rewards, slightly over 7% additional rewards with respect to the trailing PSOA heuristic. This result is as expected as the linCBwK dynamically updates the expected rewards and resource consumption, and weighs them appropriately for decision-making. The BRC heuristic follows PSOA, and RC has the worst outcome with respect to cumulative rewards. We hypothesized that the RC heuristic would perform better than BRC heuristic because of the weighted probability while choosing the delivery option. With a lower number of arms, for our computational experiments, the BRC heuristic uses the truck-based delivery option less intensively than RC which improves its performance. The same trend is observed for the successful number of allocations as seen in Figure 1.

TABLE 1: Cumulative reward obtained through successful allocations ($T = 1000, T_0 = 95$)

Instance	linCBwK			PSOA			BRC			RC		
	Min	Ave	Max	Min	Ave	Max	Min	Ave	Max	Min	Ave	Max
0	364.5	400.6	426.8	347.9	375.8	407.6	268.8	290	306.2	228	249.2	279.2
1	521.9	533.4	543.5	471.5	511.4	533.1	345.5	369.6	395.8	266.6	288.6	315.5
2	357.2	377.4	408.1	334.6	360	393.6	259.1	280.5	306.8	231.6	246.7	264.8
3	453.8	493.8	533.6	366.1	410.5	435	301.9	343.2	366	239.9	272.2	281.8
4	432.9	461.4	485.6	374.3	399.9	440.4	288.4	324.1	355.4	245.6	264.6	293.3
5	342.8	375.3	412.1	316.3	346.3	386	260.6	276.9	288.2	220.1	238.6	254
6	464.5	475.7	501	442.2	470.3	514.4	311.1	333.3	354.1	251.2	271.7	318.8
7	401.8	438.9	469.8	378.4	408.2	436.8	293.3	305.2	323	235.9	259.6	282.6
8	462.9	487.1	506.5	483.3	507.9	530	322.3	338.7	353.1	241.2	273.9	302.7
9	455.9	465.4	480	386	419.6	456.2	288.5	321.8	346.9	246.3	264.9	287.5
Overall	342.8	450.9	543.5	316.3	421	533.1	259.1	318.3	395.8	220.1	263	318.8

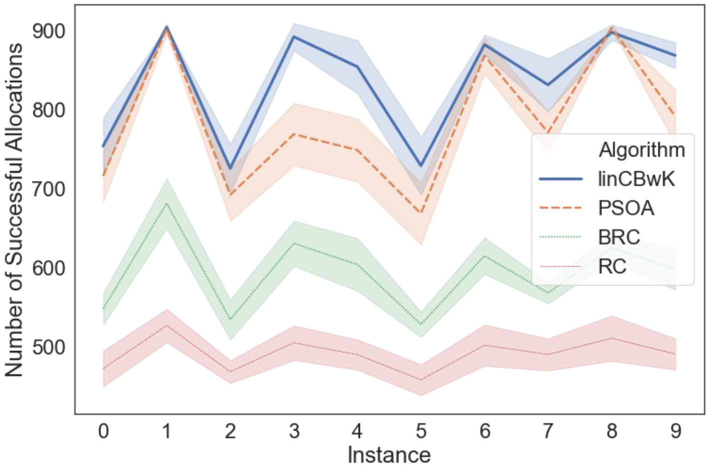


FIGURE 1: Number of successful allocations: average line with the standard deviation band ($T = 1000, T_0 = 95$)

The values ρ^S and ρ^R show the maximum deviation of the estimate used in the Stage 1 optimization problem from the simulated values used in Stage 2. The pair $(\rho^S, \rho^R) = (0.0, 0.0)$ implies that there was no estimation error during the planning stage. Figure 2 shows the effect of uncertainty on the performance of the heuristics, based on all 10 instances. As the amount of observed uncertainty increases, all heuristics perform slightly better. A likely reason for this observation is that the higher diversity in the demand generation provides more opportunities to facilities that are used less often. This would delay the resource consumption violation at a more intensively used facility.

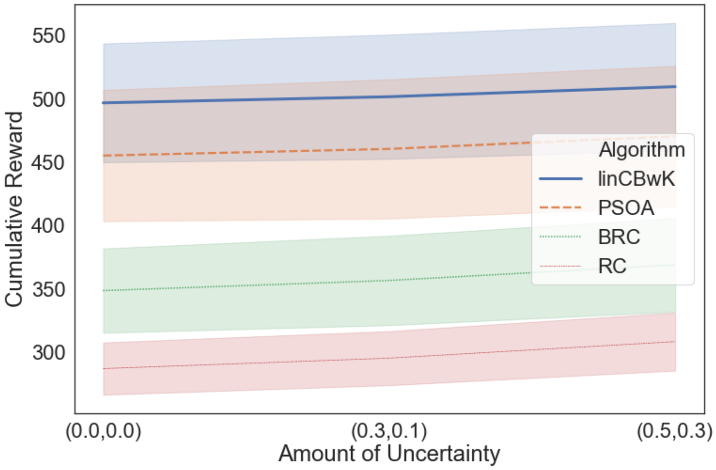
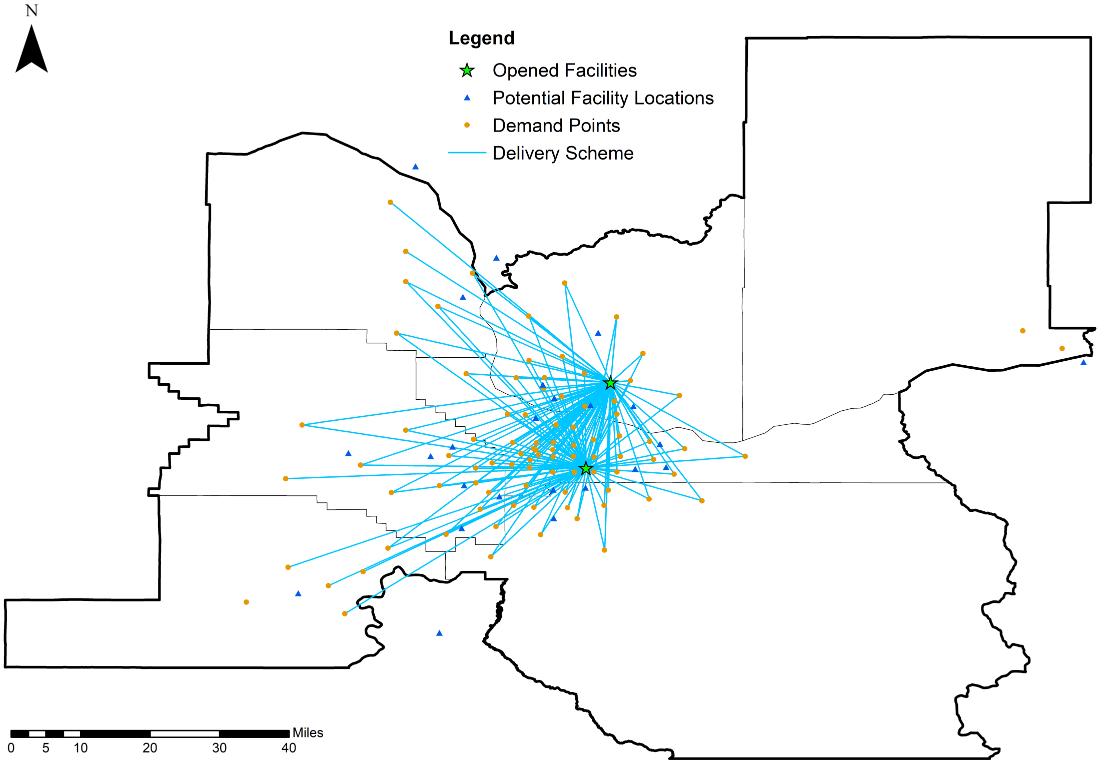


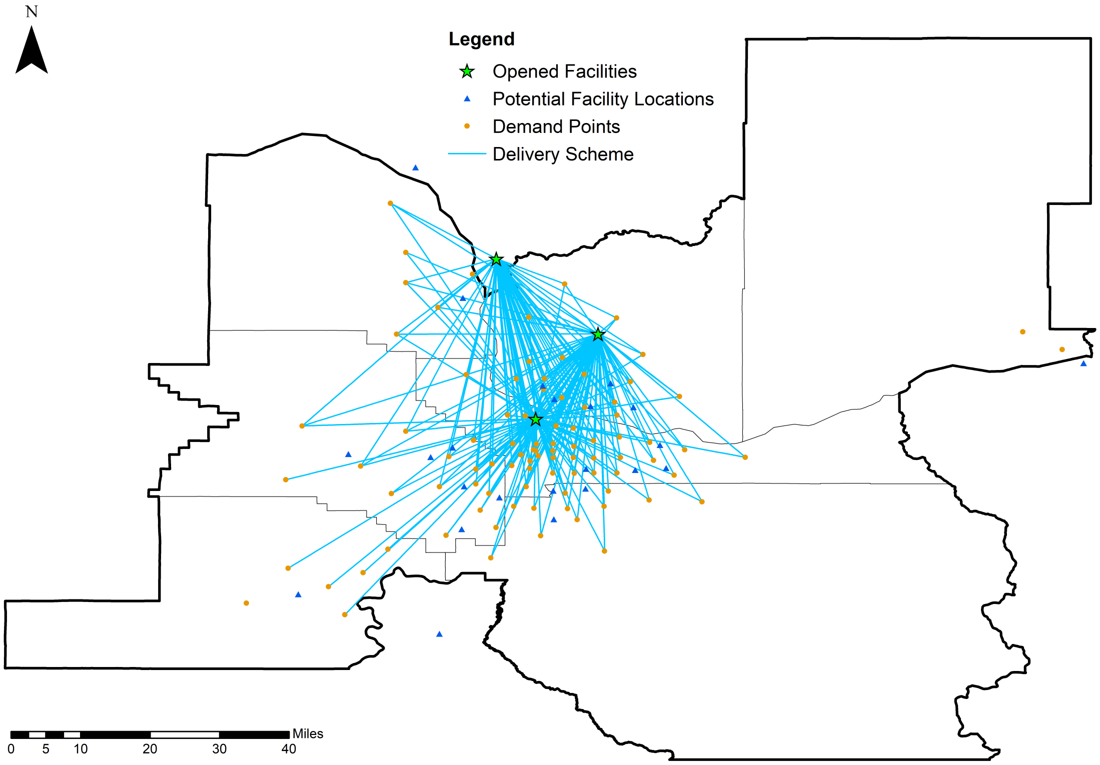
FIGURE 2: Cumulative rewards with varying the amount of uncertainty in estimating the number and type of deliveries (ρ^S,ρ^R): average line with the standard deviation band ($T = 1000, T_0 = 95$)

Portland Metro Area Case Study

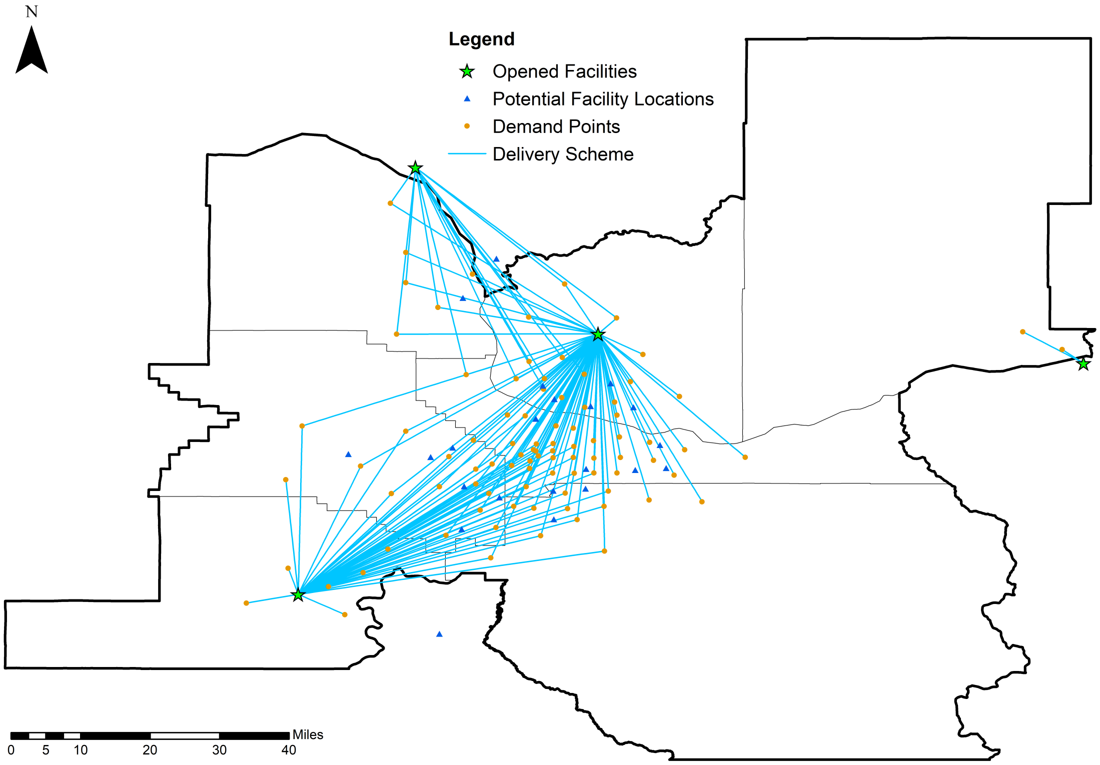
For the Portland Metro Area case study, we consider Walmart expanding its service options to offer instant delivery to its Walmart+ service subscribers (similar to Amazon's Prime subscription). The 26 Walmart stores in and around the Portland Metro are considered the potential drone-based fulfillment center candidates. The 90 centroids of the ZIP Code Tabulation Areas (ZCTAs) in the Portland Metro Area that can be serviced by drones are considered as the demand locations. The locations of Walmart stores and ZCTAs used in the study are available at https://github.com/drc1807/MPFL_DRA. The latest estimate of Walmart+ subscriber base in the USA is 60.78 million (*33*). Assuming geographically uniform subscriber base in the US and monthly ordering behavior results in 13840 anticipated daily deliveries in Portland Metro Area. We consider a planning period of one day. The proportion of time-sensitive deliveries at each demand point is randomly distributed in the interval [0.4,0.7]. The values of parameters α and β is set to 28000 and $13000 \cdot B_{drone}$, respectively. For the operational stage, the total number of orders (T) is set to 15000, and the amount of uncertainty is chosen as $\rho^S = 30\%$ and $\rho^R = 10\%$. We explore three cases by changing the number of opened facilities (p) from 2 to 4. The values of α_{min}/o_{max}, β_{min}/B_{drone}, and ω are selected to be the smallest multiple of 50 greater than $(p+1)^{0.5}T^{0.75}$. These values ensure that the regret bound conditions for linCBwK mentioned by Agrawal and Devanur (*24*) are met. Other parameters are the same as described for the p-median instances. The solutions of the planning stage optimization problem are shown in Figure 3.



(a) $p = 2$



(b) $p = 3$



(c) $p = 4$

FIGURE 3: Planning stage optimization problem solutions for Portland Metro Area

Figure 4 shows the variation in cumulative rewards achieved by various algorithms with the number of opened facilities. Here as well, the linCBwK algorithm performs best. A slight decrease in cumulative reward with increasing p is expected as the effective number of deliveries used for all algorithms, T' $(= T - T_0)$, decreases with increasing p. In this regard, PSOA gives a stable performance. However, a significant improvement in accumulated profits is experienced by linCBwK, BRC, and RC for $p = 3$. A primary reason for the improved performance is the availability of all facilities for almost every order (see Figure 3(b)) as well as proportionate distribution of product and battery resources among the facilities. As p increases, the minimum amount of resource allocation at facilities also increases. This means that the facilities available to a smaller proportion of demand points have more redundant capacity, and the more readily available facilities face scarcity of resources. This causes a drop in performance of linCBwK, BRC, and RC when $p = 4$ due to the outlying facility serving only two demand points. By numbers, linCBwK beats the second-best approach by 11.2%, 13.2%, and 25.1%, on average, as p increases from 2 to 4, respectively.

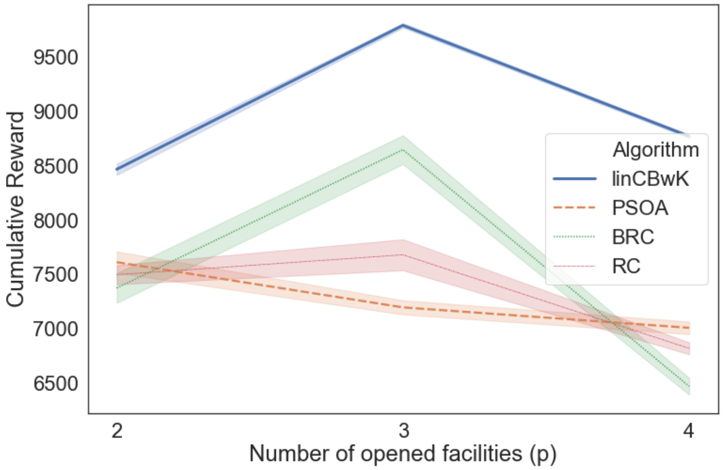


FIGURE 4: Cumulative rewards with varying the number of opened facilities (p): average line with the standard deviation band ($T = 15000,\ T_0 = (p+1)\sqrt{T}$)

CONCLUSIONS

This paper investigates a facility location and online demand allocation problem applicable to a logistics company expanding to instant delivery using UAV/drones. The problem consists of two stages: a planning stage, and an operational stage. During the planning stage, the company wishes to locate micro-fulfillment centers which serve the dual purpose of product storage and drone operations. We present a profit-maximizing mixed-integer linear program that accounts for product capacity, battery capacity, and routing cost constraints. During the operational stage, the orders arrive in an online manner and real-time decisions are made for the satisfaction of demand with an objective of maximizing cumulative profits while respecting the resource budget constraints. To the best of the authors' knowledge, this work is the first application in logistics considering non-replenishable resource consumption constraints in real-time decision-making.

We explore a multi-armed bandit framework that explicitly accounts for global knapsack constraints. We extrapolate results from extant literature to account for restricted “arm" availability in the framework arising from drone range constraints. The multi-armed bandit framework is compared with a heuristic policy derived from the planning stage optimization solution (PSOA heuristic), and two heuristics based on random choice. The analysis on standard test instances shows that the multi-armed bandit framework beats the second-best PSOA heuristic by accumulating 7% more profits, on average. An application of this problem to Portland Metro Area with a larger time horizon yields similar results with the multi-armed bandit framework performing the best, beating the second-best approach by at least 11.2%.

The present work can be expanded in various aspects. Currently, the model does not accommodate the non-fulfillment of time-sensitive orders or provide an incentive to switch to regular orders. This may be an important feature due to possible network congestion issues or drone availability issues. In this study, we assumed that enough drones are available at each facility, which may not be the case in many applications. Availability-related constraints are effectively tackled in the dynamic fleet management literature and can be a possible extension of this work.

ACKNOWLEDGMENTS

This work was supported by the National Science Foundation under grants CMMI-1562109/1562291: Collaborative Research: Non-Additive Network Routing and Assignment Models, and CMMI-1826320/1826337: Collaborative Research: Real-Time Stochastic Matching Models for Freight Electronic Marketplace.

AUTHOR CONTRIBUTIONS

The authors confirm contribution to the paper as follows: study conception and design: D.R. Chauhan, A. Unnikrishnan, S. Boyles; data collection: D.R. Chauhan; analysis and interpretation of results: D.R. Chauhan, A. Unnikrishnan; draft manuscript preparation and revisions: D.R. Chauhan, A. Unnikrishnan, S. Boyles. All authors reviewed the results and approved the final version of the manuscript.

REFERENCES

1. U.S. Census Bureau, *Quaterly Retail E-commerce sales – 4th Quarter 2020*, 2021, https://www2.census.gov/retail/releases/historical/ecomm/20q4.pdf, last accessed on 4/8/2021.
2. Aouad, A., *Walmart has filed for 97 drone patents – nearly double the amount of Amazon's*, 2019, https://www.businessinsider.com/walmart-files-double-amazons-drone-patents-2019-6, last accessed on 1/29/2021.
3. Chiang, W.-C., Y. Li, J. Shang, and T. L. Urban, Impact of drone delivery on sustainability and cost: Realizing the UAV potential through vehicle routing optimization. *Applied energy*, Vol. 242, 2019, pp. 1164–1175.
4. Powell, W. B., A unified framework for stochastic optimization. *European Journal of Operational Research*, Vol. 275, No. 3, 2019, pp. 795–821.
5. Slivkins, A., Introduction to multi-armed bandits. *arXiv preprint arXiv:1904.07272*, 2019.
6. Gu, Q., T. Fan, F. Pan, and C. Zhang, A vehicle-UAV operation scheme for instant delivery. *Computers & Industrial Engineering*, Vol. 149, 2020, p. 106809.
7. ReVelle, C. S. and H. A. Eiselt, Location analysis: A synthesis and survey. *European journal of operational research*, Vol. 165, No. 1, 2005, pp. 1–19.
8. Farahani, R. Z., M. SteadieSeifi, and N. Asgari, Multiple criteria facility location problems: A survey. *Applied mathematical modelling*, Vol. 34, No. 7, 2010, pp. 1689–1709.
9. Daskin, M. S., *Network and discrete location: models, algorithms, and applications*. John Wiley & Sons, 2011.
10. Dönmez, Z., B. Y. Kara, Ö. Karsu, and F. Saldanha-da Gama, Humanitarian Facility Location under Uncertainty: Critical Review and Future Prospects. *Omega*, 2021, p. 102393.
11. Ahmadi-Javid, A., P. Seyedi, and S. S. Syam, A survey of healthcare facility location. *Computers & Operations Research*, Vol. 79, 2017, pp. 223–263.
12. Farahani, R. Z., S. Fallah, R. Ruiz, S. Hosseini, and N. Asgari, OR models in urban service facility location: A critical review of applications and future developments. *European journal of operational research*, Vol. 276, No. 1, 2019, pp. 1–27.
13. Mukundan, S. and M. S. Daskin, Joint location/sizing maximum profit covering models. *INFOR: Information Systems and Operational Research*, Vol. 29, No. 2, 1991, pp. 139–152.
14. Gendreau, M., G. Laporte, and F. Semet, A dynamic model and parallel tabu search heuristic for real-time ambulance relocation. *Parallel computing*, Vol. 27, No. 12, 2001, pp. 1641–1653.
15. Schmid, V., Solving the dynamic ambulance relocation and dispatching problem using approximate dynamic programming. *European journal of operational research*, Vol. 219, No. 3, 2012, pp. 611–621.
16. Bektaş, T., P. P. Repoussis, and C. D. Tarantilis, Chapter 11: Dynamic vehicle routing problems. In *Vehicle Routing: Problems, Methods, and Applications, Second Edition* (P. Toth and D. Vigo, eds.), SIAM, 2014, pp. 299–347.
17. Guo, W., B. Atasoy, W. Beelaerts van Blokland, and R. R. Negenborn, Dynamic and stochastic shipment matching problem in multimodal transportation. *Transportation Research Record*, Vol. 2674, No. 2, 2020, pp. 262–273.

18. Yu, L., C. Zhang, J. Jiang, H. Yang, and H. Shang, Reinforcement learning approach for resource allocation in humanitarian logistics. *Expert Systems with Applications*, Vol. 173, 2021, p. 114663.

19. Jiang, Z., J. Gu, W. Fan, W. Liu, and B. Zhu, Q-learning approach to coordinated optimization of passenger inflow control with train skip-stopping on a urban rail transit line. *Computers & Industrial Engineering*, Vol. 127, 2019, pp. 1131–1142.

20. Villar, S. S., J. Bowden, and J. Wason, Multi-armed bandit models for the optimal design of clinical trials: benefits and challenges. *Statistical science: a review journal of the Institute of Mathematical Statistics*, Vol. 30, No. 2, 2015, p. 199.

21. Badanidiyuru, A., R. Kleinberg, and A. Slivkins, Bandits with knapsacks. In *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, IEEE, 2013, pp. 207–216.

22. Devanur, N. R., K. Jain, B. Sivan, and C. A. Wilkens, Near optimal online algorithms and fast approximation algorithms for resource allocation problems. *Journal of the ACM (JACM)*, Vol. 66, No. 1, 2019, pp. 1–41.

23. Yu, L., H. Yang, L. Miao, and C. Zhang, Rollout algorithms for resource allocation in humanitarian logistics. *IISE Transactions*, Vol. 51, No. 8, 2019, pp. 887–909.

24. Agrawal, S. and N. Devanur, Linear contextual bandits with knapsacks. In *Advances in Neural Information Processing Systems*, 2016, pp. 3450–3458.

25. Mallick, P., S. Sarkar, and P. Mitra, Decision recommendation system for transporters in an online freight exchange platform. In *2017 9th International Conference on Communication Systems and Networks (COMSNETS)*, IEEE, 2017, pp. 448–453.

26. Moshref-Javadi, M. and M. Winkenbach, Applications and Research Avenues for Drone-Based Models in Logistics: A Classification and Review. *Expert Systems with Applications*, 2021, p. 114854.

27. Lin, K., R. Zhao, Z. Xu, and J. Zhou, Efficient large-scale fleet management via multi-agent deep reinforcement learning. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2018, pp. 1774–1783.

28. Shavarani, S. M., S. Mosallaeipour, M. Golabi, and G. İzbirak, A congested capacitated multi-level fuzzy facility location problem: An efficient drone delivery system. *Computers & Operations Research*, Vol. 108, 2019, pp. 57–68.

29. Langevin, A., P. Mbaraga, and J. F. Campbell, Continuous approximation models in freight distribution: An overview. *Transportation Research Part B: Methodological*, Vol. 30, No. 3, 1996, pp. 163–188.

30. Osman, I. H. and N. Christofides, Capacitated clustering problems by hybrid simulated annealing and tabu search. *International Transactions in Operational Research*, Vol. 1, No. 3, 1994, pp. 317–336.

31. Figliozzi, M. A., Lifecycle modeling and assessment of unmanned aerial vehicles (Drones) CO2e emissions. *Transportation Research Part D: Transport and Environment*, Vol. 57, 2017, pp. 251–261.

32. Chauhan, D. R., A. Unnikrishnan, M. Figliozzi, and S. D. Boyles, Robust maximum coverage facility location problem with drones considering uncertainties in battery availability and consumption. *Transportation Research Record*, Vol. 2675, No. 2, 2021, pp. 25–39.

33. PYMTS, *Walmart+ Leverages Grocery Delivery; Gains 8 Million New Paid Subscribers*, 2021, https://www.pymnts.com/news/retail/2021/walmart-leverages-grocery-delivery-gains-8-million-new-paid-subscribers/, last accessed on 5/28/2021.