Crowdshipping with Dynamic Workers Availability – Restless-Bandit-Based Prioritization

Amin Karimi
School of Mathematics and Statistics
The University of Melbourne
Melbourne, Australia
Email: karimia@student.unimelb.edu.au

Jing Fu
School of Engineering
Royal Melbourne Institute of Technology
Melbourne, Australia
Email: jing.fu@rmit.edu.au

Lele Zhang
School of Mathematics and Statistics
The University of Melbourne
Melbourne, Australia
Email: lele.zhang@unimelb.edu.au

Hadi Ghaderi
School of Business, Law & Entrepreneurship
Swinburne University of Technology
Melbourne, Australia
Email: hghaderi@swin.edu.au

***Abstract*—Exponential growth for last mile delivery demand has created several challenges for retailers and couriers, at the same time forcing the development of efficient and sustainable delivery solutions. One of the emerging solutions is crowd-sourced delivery, also known as crowdshipping. In a crowdshipping system, the general public participates in parcel delivery (known as crowdshippers) and then rewarded with remunerations. To develop sustainable and commercially viable crowdshipping solutions, capable of handling large-scale delivery tasks, effective assignment of tasks to crowdshippers is critical. Particularly when both tasks and crowdshippers dynamically arrive and depart the system, it becomes challenging to complete deliveries, while maximizing the total profit of the platform. This paper models the dynamic crowdshipping system using a Markov decision process and proposes a restless-bandit-based capacity relaxation technique to facilitate the task-to-crowdshipper assignment. Simulation results show that the proposed technique is superior over two baseline policies with respect to higher average profits and lower task rejection rates. The learning of this research provides important directions for the design and development of crowdshipping systems that are subject to both crowdshipper and task uncertainty.**

***Index Terms*—stochastic process; restless bandits; crowdsourcing; task assignment; crowdshipping; parcel delivery**

I. Introduction

Fueled by exponential growth in e-commerce, more consumers are opting to purchase goods and services online [1]. This tendency has led to a surge in urban freight activity, particularly the Last Mile Delivery (LMD) of parcels to the doorstep of customers [2]. Furthermore, changing consumer expectations for fast, convenient and low-cost delivery options have forced retailers and logistics organizations to further expand their service configurations through increased coverage, frequency and speed, all of which lead to higher numbers of trips and vehicle activity in highly populated urban areas [3]. In tandem with these developments, efforts are being made to leverage digital technologies that could lead to efficient logistics operations, while minimizing negative environmental impacts [4]. One example of such advancements is the widespread use of mobile devices and app-based solutions that allow logistics works to be outsourced to individuals, also known as *crowd logistics* [5]. In the past decade, crowd logistics has received substantial momentum as an innovative solution that could potentially address some of the infrastructure and resource constraints of business logistics [6] [7]. In this paper, we study a form of crowd logistics service known as *crowdshipping (CS)*. Despite the absence of a uniform conceptualization, crowdshipping can be defined as a decentralized logistics system, in which individuals from the general public perform parcel delivery for an agreed compensation amount [8]. Predominantly, in such systems, the assignment of tasks to individuals, information sharing, and financial transactions are facilitated by a digital platform [9] [10].

Crowdshipping is inherently a complex and multifaceted problem that involves disciplines such as mathematics, computer science and transport engineering [11]. For example, the many-to-many nature of crowdshipping presents model complexity when considering the temporal and spatial considerations of delivery tasks and the pool of available *crowdshippers* [12]. Specifically, crowdshippers' mobility pattern is subject to constant variations. Second, crowdshipping relies on a pool of individuals who are connected via mobile devices that produce large mobility data, requiring sophisticated trajectory tools to understand and predict movement patterns [13]. Therefore, constant capturing and analyzing mobility data from a large number of individuals is computationally difficult. Third, the literature shows that crowdshippers are noticeably sensitive to their reimbursement [14] [15]. More specifically, crowdshipping platforms employ a wide range of people with varying compensation sensitivity. Therefore, a successful crowdshipping system should utilize compensation and pricing strategies that are fair to the crowd, yet, yield profit for the platform [16]. More specifically, higher compensations

 ISBN: 978-1-68558-102-2

could attract more crowdshippers, while leaving little revenue for the platform. On the other hand, low compensation could discourage participation and eventually lead to lower revenue. Finally, relevant to the objective of this research, effectively assigning jobs to crowdshippers remains a key decision-making process to the long-term sustainability of crowdshipping platforms [17]. Optimized assignments could not only minimize operational costs by identifying the right crowd with lower compensation amounts, but they also have important implications for service quality (e.g., reliability, availability and speed) and customer retention [18]. For example, an effective task assignment mechanism is capable of identifying crowdshippers that are more aligned with the delivery task requirement, temporally and spatially, leading to a quicker and cost-effective process.

In light of the above considerations, we study a large-scale crowdshipping assignment problem with a long-run optimization objective, considering dynamic assignments of available and eligible crowdshippers to different delivery requests. In contrast to static optimization approaches that aim to maximize profit at one point in time, we aim to maximize average profit over a period of time with dynamic parameters. Specifically, we take into account the dynamic registration and de-registration of crowdshippers and the dynamic arrivals of delivery requests with various service levels. Such assumptions substantially complicate the formulation of the problem and prevent conventional optimization techniques from being applied here. We formulate the crowdshipping assignment problem as a stochastic system consisting of parallel bandit processes. In a special case with a fixed number of crowdshippers at all the time, the *Restless-Bandit-Based (RBB)* resource allocation technique studied in [19] leads to near-optimal results without consuming excessive computational power. However, for the general case with a dynamic crowdshippers pool, extended or new techniques are required. In this paper, we propose the *Restless-Bandit-Based Capacity Relaxation* (RBB-CR) technique to approximate the marginal profit of assigning an incoming delivery request to a registered crowdshipper. Then we develop and evaluate an assignment strategy, referred to as the RBB-CR policy, that always prioritizes the assignments of crowdshippers with the highest approximated marginal profits. Extensive numerical results demonstrate that the RBB-CR always outperforms two baseline policies with respect to the average revenue, cost, profit, and customer satisfaction. Such superiority is consistent for a variety of system sizes and loads. After a brief literature review in Section II, we explain the crowdshipping assignment problem and its model in Section III and then discuss strategies applied in Section IV. We discuss the settings and results of the numerical study in Section V, which is followed by a brief conclusion in Section VI.

II. Related Work

While the literature on crowdshipping has grown substantially in recent years, the task assignment remains a topical area among researchers [20]. Compared to the ride-sharing task assignment, where the temporal considerations of the trip are simplified by the traveler's origin and destination points, in crowdshipping both sender's and receiver's availability are subject to variations [21]. Similarly, delivery requests could arrive at an unknown pace, a feature comparable to passenger travel requests in ride-sharing systems [22]. On the supply side, many crowdshipping platforms rely on a pool of casual workers with varying temporal and spatial availability [8]. In other words, the properties of the crowdshippers pool vary dynamically in terms of size, temporal and spatial availability, but also their ability to move goods considering the weight and size [23].

Various modeling and solution techniques have been applied to solve the crowdshipping task assignment problems in mainly three settings, deterministic, dynamic and stochastic [17]. In the deterministic environment when all information is deterministic and available in advance, methods such as mixed-integer programming and benders decomposition are widely used to find the optimal assignment [24] [17]. Efforts have been made to study the stochasticity in crowdshippers' availability or willingness to deliver [25] [26]. For example, Mousavi et al. [25] developed a two-stage stochastic integer program and decomposition algorithms to match crowdshippers to delivery tasks and demonstrated the superiority of the stochastic approach over the deterministic. The crowdshipping system is essentially a dynamic system as delivery requests and crowdshippers dynamically arrive in and leave the system. Nevertheless, the literature on the dynamic crowdshipping problem is limited. Similarly, Ghaderi et al. [24] followed a mixed-integer programming approach, in which the objective was to maximize the profit of the platform by minimizing the reimbursement, while minimizing the trip detour required by crowdshippers to complete parcel delivery. In this work, authors relied on normal people accepting to deliver a parcel as part of their daily travels, therefore, relying on extensive trajectory analytics to understand mobility patterns for optimized task assignment. Farazi et al. [27] applied heuristics-embedded Deep Q-Network (DQN) algorithms to assign dynamically arriving requests to available crowdshippers. Agent-based simulations are another useful way to model the crowdshipping system as they can easily incorporate the dynamic arrivals and departures of delivery requests and resources as well as the intelligent, stochastic decision-making processes of crowdshippers [28]. Nevertheless, compared to the proposed approach, agent-based simulations would be limited by their convergence, scalability, transparency and interpretability.

III. A Crowdshipping Assignment Problem

Define $\mathbb{N}_+$ and $\mathbb{N}_0$ as the sets of positive and non-negative integers, respectively, and for any $N \in \mathbb{N}_+$, let $[N]$ represent the set $\{1, 2, \ldots, N\}$. Let $\mathbb{R}$, $\mathbb{R}_+$ and $\mathbb{R}_0$ be the sets of all, positive and non-negative real numbers, respectively.

Consider L different types of customers that are characterized by their origins (i.e., parcel collection points), destinations (i.e., delivery addresses), parcel sizes (weights and/or volumes), and other specific requirements of the delivery services (e.g., delivery urgency – same day, next day, etc.). The customers keep generating delivery requests to a crowdshipping system (platform) with various registered crowdshippers, also correspondingly classified into L types. Each delivery request may include multiple parcels. Crowdshippers are divided into M different classes based on their locations and eligibility of serving certain requests. For instance, big parcels can only be carried by crowdshippers equipped with vans or trucks; urgent parcels prefer nearby crowdshippers; and crowdshippers in different locations only agree to detour within certain geographical distances. For $\ell \in [L]$, let $\mathscr{M}_\ell$ represent the class set of crowdshippers that are eligible and willing to deliver parcels for type-ℓ requests.

When a delivery request arrives, an available crowdshipper will be selected to deliver the associated parcel(s); if there is no such crowdshipper, the delivery request will be *rejected*. In this research, crowdshippers are not allowed to decline the offer as long as they are available and eligible for a certain type of delivery task. While this could present a limitation for this study, at the same time, it allows for a higher level of service quality and system profitability. A crowdshipper may be able to serve more than one delivery request. We consider a *delivery weight*, $w_{\ell,m} \in [0,1]$, for a type-ℓ request matched with a class-m crowdshipper, meaning that the $w_{\ell,m}$ proportion of the crowdshipper's carrying capacity, such as vehicle's storage space, is used and occupied for the delivery request. For example, if a crowdshipper in class m is able to take three delivery requests of type ℓ, then we can set $w_{\ell,m} = \frac{1}{3}$ to formulate this case. In this case, the $w_{\ell,m}$ proportion of the crowdshipper's capacity is *occupied* and becomes *unavailable* for future requests until all the associated parcels are delivered. Upon successful delivery of a request, the crowdshipper becomes partially/fully *available* again to serve other requests. The crowdshippers dynamically join and leave the crowdshipping system, resulting in time-varying numbers of registered crowdshippers in each class. In other words, crowdshippers declare their availability for work randomly. Once a crowdshipper has been occupied by request(s), he or she will not leave the crowdshipping system until all the requests are fully completed.

Let $\bar{C}_m(t)$ represent the number of class-m crowdshippers that are assigned with some delivery requests at time t, and let $\bar{\boldsymbol{C}}(t) \coloneqq (\bar{C}_m(t) : m \in [M])$. Furthermore, for $m \in [M]$, let $C_m(t)$ represent the total number of registered crowdshippers of class m in the system at time $t \geq 0$, which is affected by both people's willingness of becoming crowdshippers (i.e., signing up for crowdshipping) and the underlying strategies of matching them with different delivery requests. Formally, we define

$$C_m(t) \coloneqq E_m(t) + \Delta_m(t), \tag{1}$$

where $E_m(t)$ is a random variable considered as a hyper-parameter reflecting people's willingness of staying in and joining the crowdshipping system at time t, and $\Delta_m(t) \coloneqq \max\{0, \bar{C}_m(t) - E_m(t)\}$ is used to ensure that $C_m(t) \geq \bar{C}_m(t)$ all the time. In particular, $E_m(t)$ is bounded and takes values in $\mathbb{N}_+$. While some crowdshippers, who are on the way to deliver parcels, may wish to leave the system, causing $E_m(t) < \bar{C}_m(t)$ for some t, adding the second item $\Delta_m(t)$ in (1) aims to keep $C_m(t) = \bar{C}_m(t)$ when $E_m(t) < \bar{C}_m(t)$ so that these crowdshippers are required to finish their jobs before de-registration. Assume that $C_m(t)$ is observable all the time and is uniformly bounded for all $t \geq 0$. Let $\boldsymbol{C}(t) \coloneqq (C_m(t) : m \in [M])$.

As mentioned in Section I, the long-run optimization problem proposed in this research is essential for operating a real-world crowdshipping system. Such an assumption significantly complicates the task assignment module and related analysis. To model the delivery requests with different types, we consider mean rates λ_ℓ ($\ell \in [L]$), each arriving with pre-determined origin, destination, parcel profiles, and some delivery preferences. Delivery time for each request is a random time that formulates uncertainties along the travel route. Request arrivals follow a Poisson distribution (see [29]). In Section V, extensive numerical results will be presented with time-variant Poisson arrivals that further capture the dynamic features of the system workloads in both busy and idle hours. For $\ell \in [L]$ and $m \in \mathscr{M}_\ell$, the duration of serving a type-ℓ request by a crowdshipper in class m is independently and identically distributed with mean $1/\mu_{\ell,m}$, where $\mu_{\ell,m} \in \mathbb{R}_+$. For $m \notin \mathscr{M}_\ell$, define $\mu_{\ell,m} \equiv 0$.

We make decisions upon the arrival of requests in an online manner without assuming given requests that will come in the future. Define $N_{\ell,m}(t)$ as the number of type-ℓ requests that are being served by a class-m crowdshipper at time t. Let $\boldsymbol{N}(t) \coloneqq \big(N_{\ell,m}(t) : \ell \in [L], m \in [M]\big)$, where $N_{\ell,m}(t) \equiv 0$ if $m \notin \mathscr{M}_\ell$. The number of occupied crowdshippers in class m at time t is given by $\bar{C}_m(t) = \sum_{\ell \in [L]} N_{\ell,m}(t) w_{\ell,m}$, which should not exceed the value of $C_m(t)$. More specifically, define $a_{\ell,m}\big(\boldsymbol{N}(t), \boldsymbol{C}(t)\big) \in \{0,1\}$ as an action variable indicating whether a type-ℓ request newly arrived at time t is going to be served by a crowdshipper in class m or not. If $a_{\ell,m}\big(\boldsymbol{N}(t), \boldsymbol{C}(t)\big) = 1$, the newly arrived type-ℓ request is served by a crowdshipper in class m; otherwise, not served by class-m crowdshippers. Define $a_{\ell,m}(\cdot,\cdot) \equiv 0$ if $m \notin \mathscr{M}_\ell$. The action variables are determined based on knowledge of $\boldsymbol{N}(t)$ and $\boldsymbol{C}(t)$ at time t and are subject to

$$\sum_{m \in \mathscr{M}_\ell \cup \{m_\ell\}} a_{\ell,m}\big(\boldsymbol{N}(t), \boldsymbol{C}(t)\big) = 1, \ \forall \ell \in [L], t \geq 0, \tag{2}$$

where m_ℓ is a dummy crowdshipper standing for rejection of a type-ℓ request, and

$$\sum_{\ell \in [L]} N_{\ell,m}(t) w_{\ell,m} + \sum_{\ell \in [L]} a_{\ell,m}\big(\boldsymbol{N}(t), \boldsymbol{C}(t)\big) w_{\ell,m} \leq C_m(t), \quad \forall m \in [M], t \geq 0. \tag{3}$$

In (2), for a newly arrived type-ℓ delivery request, we always select a crowdshipper to serve it or reject it - setting $a_{\ell,m_\ell}\big(\boldsymbol{N}(t), \boldsymbol{C}(t)\big) = 1$. Inequality (3) ensures that we only assign the registered crowdshippers to serve newly arrived delivery requests. In particular, upon a decision-making epoch t, we will check the number of registered crowdshippers. If

 ISBN: 978-1-68558-102-2

$C_m(t) = \bar{C}_m(t)$ at time t (i.e., all registered crowdshippers are occupied by earlier arrived parcels), then (3) enforces $a_{\ell,m}\big(\boldsymbol{N}(t), \boldsymbol{C}(t)\big) = 0$ for all $\ell \in [L]$. Unlike the canonical restless-bandit-based resource allocation problem discussed in [19], the right-hand side of constraint (3) (i.e., $C_m(t)$) is dependent on the employed crowdshipping assignment strategies. It prevents past results in [19], which assumes $C_m(t) \equiv C$ for some constant $C \in \mathbb{N}_+$, from being directly applied here.

Here, we adapt the RBB resource allocation technique in the request-crowdshipper assignment problem. More importantly, we extend the technique to practical scenarios with time-varying and strategy-dependent resource capacities. The stochastic process $\big\{\big(\boldsymbol{N}(t), \boldsymbol{C}(t)\big), t \geq 0\big\}$ evolves according to a transition kernel that is affected by the above-defined action variables $a_{\ell,m}(\boldsymbol{N}(t), \boldsymbol{C}(t))$ for all $\ell \in [L]$ and $m \in [M]$. Define a set Φ of crowdshipping assignment strategies determined by such action variables $a_{\ell,m}\big(\boldsymbol{N}(t), \boldsymbol{C}(t)\big) \in \{0, 1\}$. To emphasize the dependencies between the employed strategy, the action variables, and the underlying stochastic process, we rewrite $a_{\ell,m}(\cdot,\cdot)$, $\boldsymbol{N}(t)$, $\boldsymbol{C}(t)$ and $\bar{\boldsymbol{C}}(t)$ as $a^{\phi}_{\ell,m}(\cdot,\cdot)$, $\boldsymbol{N}^{\phi}(t)$, $\boldsymbol{C}^{\phi}(t)$ and $\bar{\boldsymbol{C}}^{\phi}(t)$ for $\phi \in \Phi$, respectively. We aim to maximize the long-run average profit of the crowdshipping system

$$\max_{\phi\in\Phi} \lim_{T\to+\infty} \frac{1}{T}\mathbb{E}\int_0^T \sum_{\ell\in[L]} \sum_{m\in[M]} R_{\ell,m}\big(N^{\phi}_{\ell,m}(t)\big)dt, \tag{4}$$

subject to (2) and (3), where $R_{\ell,m}\big(N^{\phi}_{\ell,m}(t)\big)$ is the expected profit rate of the process $\big\{N^{\phi}_{\ell,m}(t), t \geq 0\big\}$. In particular, define the profit rate

$$R_{\ell,m}\Big(N^{\phi}_{\ell,m}(t)\big)\Big) := \mathscr{p}_{\ell}\mu_{\ell,m}N^{\phi}_{\ell,m}(t) - \mathscr{c}_{\ell,m}N^{\phi}_{\ell,m}(t), \tag{5}$$

where $\mathscr{p}_{\ell}$ is the monetary income of successfully delivering the parcel(s) for a request of type ℓ, $\mathscr{c}_{\ell,m}$ is the cost rate per unit time per crowdshipper that is specified by the administration expenditure, the material and labor costs, etc., and $1/\mu_{\ell,m}$ is the expected delivery time for a class-m crowdshipper to complete the type-ℓ request. The product $\mu_{\ell,m}N^{\phi}_{\ell,m}(t)$ thus represents the expected number of type-ℓ requests completed by class-m crowdshippers per unit time at time t when the strategy ϕ is employed. We consider bounded $C_m(t)$ for all $m \in [M]$, hence the crowdshipping system is always stable with existing equilibrium average objective in the long-run regime. We refer to the problem described in (4), (2) and (3) as the *crowdshipping assignment problem* and also the *original problem*.

IV. Crowdshipping Assignment Strategies

The crowdshipping assignment problem is modeled by a Markov Decision Process (MDP) with state space

$$\begin{aligned}\mathscr{N} &:= \Bigg(\prod_{\ell\in[L]}\prod_{m\in\mathscr{M}_{\ell}}\Big\{0,1,\ldots,\lfloor\frac{C}{w_{\ell,m}}\rfloor\Big\}\Bigg)\times\{0,1,2\ldots,C\}^M \\ &= \Bigg(\prod_{\ell\in[L]}\prod_{m\in\mathscr{M}_{\ell}}\mathscr{N}_{\ell,m}\Bigg)\times\{0,1,2\ldots,C\}^M, \end{aligned}\tag{6}$$

where $C \in \mathbb{N}_+$ is an upper bound of $C_m(t)$ for $m \in [M]$ satisfying $C \geq \max_{m\in[M]} C_m(t)$ for all $t \geq 0$, and $\mathscr{N}_{\ell,m} := \{0, 1, \ldots, \lceil C/w_{\ell,m}\rceil\}$. Observing that the crowdshipping assignment problem exhibits large state space that increases exponentially in the number of L and M. In general, an optimal solution is intractable for such problems, and conventional optimizers, including off-the-shelf reinforcement learning solutions, cannot be directly applied here due to the exposing state space.

We resort to near-optimal strategies that are applicable to large-scale systems without consuming excessive computational power. The RBB resource allocation technique in [19] proposes to approximate optimality by decomposing the high-dimensional MDP into $\sum_{\ell\in[L]}|\mathscr{M}_{\ell}|$ sub-processes, with each considered as an MDP with binary actions evolving within remarkably reduced state space. We refer to such sub-process as a *bandit process* that is associated with a request-crowdshipper (RCS) pair $(\ell, m) \in [L]\times[M]$. The marginal profit of selecting a class-m crowdshipper to serve a type-ℓ request is evaluated and quantified through a real number, whenever $N^{\phi}_{\ell,m}(t)$ and $C^{\phi}_m(t)$ are given. Following the tradition of the restless-bandit community, we refer to such real number as the *index* of the associated action of selecting a certain RCS pair, and, in each decision-making epoch, always prioritize the actions with the highest indices. The indices for each bandit process are computed independently from those of other bandit processes, which consume a limited amount of computational power and enable the applicability of the resulting crowdshipping strategy in the original problem (4), (2) and (3).

Since the crowdshipping problem studied here is complicated due to the strategy-dependent capacity $C^{\phi}_m(t)$ of crowdshipper resources, we approximate the marginal profits of selecting certain crowdshippers through appropriate randomization of the action variables and the capacities of crowdshippers in the asymptotic regime.

A. Restless-Bandit-Based Capacity Relaxation (RBB-CR)

Randomise the action variables and relax the constraints (2) and (3) to

$$\lim_{t\to+\infty}\sum_{m\in\mathscr{M}_{\ell}\cup\{m_{\ell}\}}\mathbb{E}\Big[a^{\phi}_{\ell,m}\big(\boldsymbol{N}(t),\boldsymbol{C}(t)\big)\Big] = 1, \forall\ell\in[L], \tag{7}$$

and

$$\begin{aligned}&\lim_{t\to+\infty}\sum_{\ell\in[L]}w_{\ell,m}\mathbb{E}\Big[N^{\phi}_{\ell,m}(t)\Big] \\ &\quad+\lim_{t\to+\infty}\sum_{\ell\in[L]}w_{\ell,m}\mathbb{E}\Big[a^{\phi}_{\ell,m}\big(\boldsymbol{N}^{\phi}(t),\boldsymbol{C}^{\phi}(t)\big)\Big] \\ &\qquad\leq\lim_{t\to+\infty}\sum_{\ell\in[L]}w_{\ell,m}\mathbb{E}\Big[C^{\phi}_m(t)\Big], \forall m\in[M], \end{aligned}\tag{8}$$

respectively. We note that in a stable system, for a given strategy ϕ, the steady-state distributions of the random variables $\boldsymbol{N}^{\phi}(t)$ and $\boldsymbol{C}^{\phi}(t)$ exist as $t \to +\infty$. Define a set $\tilde{\Phi}$ of all crowdshipping strategies that are determined by the randomized action variables, and consider a relaxed version of the crowdshipping assignment problem

$$\max_{\phi\in\tilde{\Phi}} \lim_{T\to+\infty} \frac{1}{T}\mathbb{E}\int_0^T \sum_{\ell\in[L]}\sum_{m\in[M]} R_{\ell,m}\big(N^{\phi}_{\ell,m}(t)\big)dt, \tag{9}$$

which achieves an upper bound to the maximum of the original crowdshipping assignment problem. We refer to the problem described in (9), (7) and (8) as the *relaxed problem*.

Consider a strategy $\phi^* \in \tilde{\Phi}$ that is optimal to the relaxed problem described in (9), (7) and (8). Based on the existence of the steady-state distributions, there exists the limit $C^*_m = \lim_{t\to+\infty}\mathbb{E}\big[\bar{\boldsymbol{C}}^{\phi^*}_m(t)\big]$. We construct a *surrogate* version of the relaxed problem by replacing the constraints in (8) with

$$\lim_{t\to+\infty}\sum_{\ell\in[L]} w_{\ell,m}\mathbb{E}\Big[N^{\phi}_{\ell,m}(t)\Big] + \lim_{t\to+\infty}\sum_{\ell\in[L]} w_{\ell,m}\mathbb{E}\Big[a^{\phi}_{\ell,m}\big(\boldsymbol{N}^{\phi}(t),\boldsymbol{C}^{\phi}(t)\big)\Big] \leq C^*_m, \forall m\in[M], \tag{10}$$

We refer to the problem described by (9), (7) and (10) as the *surrogate problem*. The strategy ϕ^* satisfies (7) and (10) and thus is also applicable to the surrogate problem. It follows that the maximum of the surrogate problem achieves an upper bound to that of the relaxed problem and the original crowdshipping assignment problem.

For $\ell\in[L]$ and $m\in[M]$, define

$$\alpha^{\phi}_{\ell,m}(n,c) \coloneqq \lim_{t\to+\infty}\mathbb{E}\Big[a^{\phi}_{\ell,m}\big(\boldsymbol{N}^{\phi}(t),\boldsymbol{C}^{\phi}(t)\big) \Big| N^{\phi}_{\ell,m}(t)=n, C^{\phi}_m(t)=c\Big], \tag{11}$$

which takes values in $[0,1]$. From [19, Proposition 4], there exist an optimal solution $\tilde{\phi}\in\tilde{\Phi}$ for the surrogate problem, $\boldsymbol{\nu}\in\mathbb{R}^L$ and $\boldsymbol{\gamma}\in\mathbb{R}^M_0$ such that, for $\ell\in[L]$, $m\in\mathscr{M}_\ell\cup\{m_\ell\}$ and $(n,c)\in\mathscr{N}_{\ell,m}\times\{0,1,\ldots,C\}$, if

$$\nu_\ell < \frac{\lambda_\ell}{\mu_{\ell,m}}R_{\ell,m}(n) - (1+\frac{\lambda_\ell}{\mu_{\ell,m}})w_{\ell,m}\gamma_m, \tag{12}$$

then $\alpha^{\tilde{\phi}}_{\ell,m}(n,c)=1$; if

$$\nu_\ell = \frac{\lambda_\ell}{\mu_{\ell,m}}R_{\ell,m}(n) - (1+\frac{\lambda_\ell}{\mu_{\ell,m}})w_{\ell,m}\gamma_m, \tag{13}$$

then $\alpha^{\tilde{\phi}}_{\ell,m}(n,c)=a$ where $a\in[0,1]$; otherwise, $\alpha^{\tilde{\phi}}_{\ell,m}(n,c)=0$. Note that, in (12) and (13), the value of $\alpha^{\tilde{\phi}}_{\ell,m}(n,c)$ is independent from c.

More importantly, such strategy $\tilde{\phi}$ can be constructed without assuming any knowledge of $\boldsymbol{\nu}$, $\boldsymbol{\gamma}$ and C^*_m. Define a ranking $\mathscr{o}$ of the RCS-State (RCSS) tuples $(\ell,m,n)\in\prod_{\ell\in[L]}\prod_{m\in\mathscr{M}_\ell\cup\{m_\ell\}}\mathscr{N}_{\ell,m}$ that are ranked according to the descending order of

$$\eta_{\ell,m}(n) \coloneqq \frac{\frac{\lambda_\ell}{\mu_{\ell,m}}R_{\ell,m}(n)}{w_{\ell,m}\big(1+\frac{\lambda_\ell}{\mu_{\ell,m}}\big)}, \tag{14}$$

where $\eta_{\ell,m_\ell}(n)\equiv 0$. For $(\ell,m)\in\prod_{\ell\in[L]}\mathscr{M}_\ell\cup\{m_\ell\}$, if

Input : Indices $\boldsymbol{\eta}$ and the state variables $\big(\boldsymbol{N}^{\text{RBB-CR}}(t),\boldsymbol{C}^{\text{RBB-CR}}(t)\big)$ at time t.
Output: $\boldsymbol{a}^{\text{RBB-CR}}\big(\boldsymbol{N}^{\text{RBB-CR}}(t),\boldsymbol{C}^{\text{RBB-CR}}(t)\big)$
Function *RBB-CR*
  $\boldsymbol{a}^{\text{RBB-CR}}\big(\boldsymbol{N}^{\text{RBB-CR}}(t),\boldsymbol{C}^{\text{RBB-CR}}(t)\big)\leftarrow\boldsymbol{0}$, $\boldsymbol{q}\leftarrow 0$
  Initialise a set $\mathscr{A}$ of all the RCSS tuples $(\ell,m,n)\in\prod_{\ell\in[L]}\prod_{m\in\mathscr{M}_\ell\cup\{m_\ell\}}\mathscr{N}_{\ell,m}$ with $N^{\text{RBB-CR}}_{\ell,m}(t)=n$.
  Build a maximum heap $\mathscr{H}$ of the set $\mathscr{A}$ according to the descending order of the RBB-CR indices.
  while $\mathscr{H}\neq\emptyset$ **do**
    $(\ell,m,n)\leftarrow$ the root node of the maximum heap $\mathscr{H}$
    if $q_\ell=0$ *and the constraints in* (2) *and* (3) *are not violated by setting* ϕ *=RBB-CR and* $a^{\text{RBB-CR}}_{\ell,m}(n)=1$ **then**
      $a^{\text{RBB-CR}}_{\ell,m}\big(\boldsymbol{N}^{\text{RBB-CR}}(t),\boldsymbol{C}^{\text{RBB-CR}}(t)\big)\leftarrow 1$
      $q_\ell\leftarrow 1$
    end
    $\mathscr{H}$ pops the root node
  end
  return $\boldsymbol{a}^{\text{RBB-CR}}\big(\boldsymbol{N}^{\text{RBB-CR}}(t),\boldsymbol{C}^{\text{RBB-CR}}(t)\big)$
End

Fig. 1: Pseudo-code for implementing the RBB-CR policy.

$\eta_{\ell,m}(n)=\eta_{\ell,m}(n')$ with $n<n'$, then the tuple (ℓ,m,n) precedes (ℓ,m,n'). Other tie cases are broken arbitrarily. Let $\boldsymbol{\eta}\coloneqq(\eta_{\ell,m}(n):\ell\in[L],m\in\mathscr{M}_\ell\cup\{m_\ell\},n\in\mathscr{N}_{\ell,m})$. Given $\mathscr{o}$, the strategy $\tilde{\phi}$ can be determined by plugging the ranking $\mathscr{o}$ in [19, Algorithm 1]. Since the policy $\tilde{\phi}$ is optimal to the surrogate problem, it achieves an upper bound of the maximum of the original crowdshipping assignment problem, which is the main concern of this paper. In general, $\tilde{\phi}$ is not applicable to the original problem because it does not necessarily satisfy (2) and (3). Nonetheless, the priority style of $\tilde{\phi}$ reveals and quantifies the importance of selecting certain RCS pairs through the $\boldsymbol{\eta}$-based tuple ranking $\mathscr{o}$.

B. *Restless-Bandit-Based (RBB) Crowdshipping Policy*

Intuitively, the policy $\tilde{\phi}$ (optimal to the surrogate problem) always prioritizes the tuples with higher $\eta_{\ell,m}(n)$, which represents the marginal profit of its associated action - selecting a crowdshipper in class m to serve a delivery request of type ℓ when $N^{\phi}_{\ell,m}(t)=n$. Although $\eta_{\ell,m}(n)$ exists in a closed form, it considers the past and future profits of the process $\{N^{\phi}_{\ell,m}(t),t\geq 0\}$ that are possibly gained by taking the associated selection. Following the tradition of the restless bandit problems, we refer to $\eta_{\ell,m}(n)$ as the *restless-bandit-based capacity relaxation (RBB-CR) index* of the bandit process $\{N^{\phi}_{\ell,m}(t),t\geq 0\}$ when $N^{\phi}_{\ell,m}(t)=n$.

Following the form of $\tilde{\phi}$ and the RBB resource allocation technique in [19], we propose a crowdshipping assignment strategy, applicable to the original problem described in (4), (2) and (3), by prioritizing selections of RCS pairs (ℓ,m) (selecting a crowdshipper in class m to serve a delivery request of type ℓ) according to the descending order of the RBB-CR indices $\boldsymbol{\eta}$. We refer to such crowdshipping assignment strategy as the RBB-CR policy.

At time $t \geq 0$, we observe the variables $\boldsymbol{N}^{\text{RBB-CR}}(t)$ and $\boldsymbol{C}^{\text{RBB-CR}}(t)$ and determine the action variables

$$\boldsymbol{a}^{\text{RBB-CR}}\big(\boldsymbol{N}^{\text{RBB-CR}}(t), \boldsymbol{C}^{\text{RBB-CR}}(t)\big) := \Big(a^{\text{RBB-CR}}_{\ell,m}\big(\boldsymbol{N}^{\text{RBB-CR}}(t), \boldsymbol{C}^{\text{RBB-CR}}(t)\big) : \ell \in [L], m \in \mathscr{M}_\ell \cup \{m_\ell\}\Big) \quad (15)$$

in the steps described in Fig. 1. In particular, we start with initializing $\boldsymbol{a}^{\text{RBB-CR}}\big(\boldsymbol{N}^{\text{RBB-CR}}(t), \boldsymbol{C}^{\text{RBB-CR}}(t)\big)$ to be $\mathbf{0}$, a vector set $\boldsymbol{q} \in \{0,1\}^L$ to be $\mathbf{0}$, and a set $\mathscr{A}$ of all the RCSS tuples $(\ell, m, n) \in \prod_{\ell\in[L]}\prod_{m\in\mathscr{M}_\ell\cup\{m_\ell\}} \mathscr{N}_{\ell,m}$ with $N^{\text{RBB-CR}}_{\ell,m}(t) = n$. All the tuples in $\mathscr{A}$ are ranked according to the descending order of their RBB-CR indices, where tie cases are broken by selecting smaller $N^{\text{RBB-CR}}_{\ell,m}(t)$. Let $(\ell_\iota, m_\iota, n_\iota)$ represent the ιth tuple in $\mathscr{A}$, where $\iota = 1, 2, \ldots, |\mathscr{A}|$. In Lines 5 to 11 of the RBB-CR policy in Fig. 1, we seek for the smallest $\iota \in [|\mathscr{A}|]$ such that $q_{\ell_\iota} = 0$ and the constraints in (2) and (3) are not violated by setting $\phi = $ RBB-CR and $a^{\text{RBB-CR}}_{\ell_\iota,m_\iota}\big(\boldsymbol{N}^{\text{RBB-CR}}(t), \boldsymbol{C}^{\text{RBB-CR}}(t)\big) = 1$. For this smallest ι complying with (2) and (3), set $a^{\text{RBB-CR}}_{\ell_\iota,m_\iota}\big(\boldsymbol{N}^{\text{RBB-CR}}(t), \boldsymbol{C}^{\text{RBB-CR}}(t)\big) = 1$, $q_{\ell_\iota} = 1$, and keep exploring the remaining elements in $\mathscr{A}$ (exploring larger $\iota \in [|\mathscr{A}|]$). Including ordering of the RCSS tuples in $\mathscr{A}$, the computational complexity of the RBB-CR policy in Fig. 1 is $O(LM\log(LM))$, which is fast for large L and M. Upon an arrival of a delivery request of type $\ell \in [L]$, we assign a crowdshipper in the class $m \in \mathscr{M}_\ell$ with $a^{\text{RBB-CR}}_{\ell,m}\big(\boldsymbol{N}^{\text{RBB-CR}}(t), \boldsymbol{C}^{\text{RBB-CR}}(t)\big) = 1$ or reject it.

In the unrealistic case with constant $C^\phi_m(t) \equiv C$ for all $m \in [M]$, from [19, Theorem EC.1], the underlying stochastic process under the RBB-CR policy converges to a fixed point, referred to as the *global attractor*, *almost surely* as the number of crowdshippers and the arrival rates of delivery requests tend to infinity, proportionately. It follows that the RBB-CR coincides with the policy $\tilde{\phi}$ in the asymptotic regime. More specifically, RBB-CR is *asymptotically optimal* to the crowdshipping assignment problem described in (4), (2) and (3) as the number of crowdshippers and the arrival rates of delivery requests tend to infinity proportionately.

We argue that the global attractor still exists in the case with the strategy-dependent $C^\phi_m(t)$. Observing that RBB-CR is independent of the exact values of C^*_m for $m \in [M]$, we can replace C^*_m in (10) with $\lim_{t\to+\infty}\mathbb{E}\big[C^{\text{RBB-CR}}_m(t)\big]$, and it will lead to the same RBB-CR indices and a priority style policy, similar to $\tilde{\phi}$ and denoted by $\tilde{\varphi}$, achieving an upper bound of the original maximization problem. When the number of crowdshippers and the arrival rates of the delivery requests increase proportionately to a parameter $h \to +\infty$ (i.e., approach the asymptotic regime with the *scaling parameter* h), the transition rates of the underlying process $\{\boldsymbol{N}^{\text{RBB-CR}}(t), \boldsymbol{C}^{\text{RBB-CR}}(t), t \geq 0\}$ will also increase correspondingly. The normalized state vector $\big(\boldsymbol{N}^{\text{RBB-CR}}(t)/h, \boldsymbol{C}^{\text{RBB-CR}}(t)/h\big)$ is likely to stay in a neighborhood of a fixed point as $h \to +\infty$, similar to the case with $C^\phi_m(t) \equiv C$, leading to the asymptotic optimality of the RBB-CR policy.

V. Numerical Results

This section is devoted to evaluating the performance of the proposed RBB-CR policy through some numerical experiments.

A. Simulation Settings

Consider $M = 10$ different classes of crowdshippers. The crowdshippers are distinguished by their transport modes: with and without vehicles. If crowdshippers in the class $m \in [M]$ of crowdshippers have no vehicles, then we normalize the moving speed v_m to be 1, if they have no vehicles and adopt other transport modes (e.g., public transport, bikes or e-bikes); otherwise, set the moving speed to be $v_m = 2.5$. Each crowdshipper class $m \in [M]$ corresponds to a geographical region with the *geographical center* (x_m, y_m), and the crowdshippers in the class are randomly distributed in this region with the location denoted by a random vector $(X_m, Y_m) \in \mathbb{R}_0 \times \mathbb{R}_0$ with the mean $(x_m, y_m) = (\mathbb{E}X_m, \mathbb{E}Y_m)$. The geographical centers are randomly distributed in a 2-D plane $[0, 50] \times [0, 50]$.

For class $m \in [M]$, the cost rates (per unit time) $\mathscr{c}_{\ell,m}$ for the crowdshipping system to hire the crowdshippers to serve an ℓ-request are set to be $\bar{\mathscr{c}}_m \in \mathbb{R}_+$ for all $\ell \in [L]$. If the crowdshippers of class $m \in [M]$ deliver parcels with vehicles, then $\bar{\mathscr{c}}_m$ is uniformly randomly generated from $[5, 10]$; otherwise, uniformly randomly taken from $[0.1, 3]$.

For each class $m \in [M]$, the number of crowdshippers that are willing to stay in the system at time t, $E_m(t)$, is set to $\big\lceil \max\big\{0, \bar{E}_m + \Delta_m(t)\big\}\big\rceil$, where $\bar{E}_m \in \mathbb{R}_+$ is uniformly randomly generated from $[5, 10]$, and $\Delta_m(t) \in \mathbb{R}$ is a normally distributed random number at time t. In Section V-C, we will explore the performance of systems with increasing $\bar{E}_m$ and compatibly many delivery requests per unit time.

In this preliminary numerical study, we consider $L = 2$ types of delivery requests: urgent (type $\ell = 1$) and regular (type $\ell = 2$) delivery. The proposed model scales well and can easily handle large L without consuming significantly more computational expenses. The monetary incomes $\mathscr{p}_1$ and $\mathscr{p}_2$ of completing delivery requests of the two types are uniformly distributed in $[200, 300]$ and $[100, 150]$, respectively. The delivery requests of type $\ell \in [L]$ keep arriving in the system with rates λ_ℓ that are randomly generated from $[1, 1.5]$. The origin position of an ℓ-request is randomly deployed in a region of the 2-D plane with expectation $(\bar{x}_\ell, \bar{y}_\ell) \in [0, 50] \times [0, 50]$. The deliver distance for a crowdshipper of class $m \in [M]$ to deliver the ℓ-request is set to $D_\ell = v_m \cdot \frac{1}{\mu_{\ell,m}}$, where v_m is the moving speed of the class-m crowdshippers and $\mu_{\ell,m}$ is the reciprocal of the expected time used to deliver a parcel from its origin position. For $\ell \in [L]$, consider $d_\ell = \mathbb{E}D_\ell = \lambda_\ell/\rho$, where $\rho \in \mathbb{R}_+$ is a given parameter representing the offered traffic intensity of the system. We adjust different values for ρ, indicating regular and busy periods of the crowdshipping system, in the following simulations. In this context, the total cost for a crowdshipper of class m to deliver an ℓ-parcel is equal to the cost rate $c_{\ell,m}$ multiplying the delivery time $\frac{D_\ell}{v_m}$.

For ℓ-requests, the eligibility of crowdshippers is based on

- the distance between the origin location of the parcel and the geographical center of the crowdshippers; and
- the relationship between the delivery type and the working modes of the crowdshippers.

In particular, urgent delivery requests can only be served by crowdshippers with vehicles while regular requests can be served by crowdshippers in either working mode. The exact settings of the eligibility between different requests and crowdshippers, specified as $\mathscr{M}_\ell$ ($\ell \in [L]$), are provided in Appendix A, together with the instance values of all the above mentioned random variables for the simulations. The model was coded using the C++ language and implemented on the high performance computing platform, Spartan [30], offered by The University of Melbourne.

B. Performance Evaluation

We demonstrate the effectiveness of the RBB-CR policy by comparing it to two baselines policies: Highest-Price (HP) and Shortest-Distance (SD). The HP and SD policies are greedy policies that always prioritize RCS pairs (ℓ, m) with the highest monetary incomes p_ℓ and the shortest travel distances, respectively. The travel distance for an RCS pair (ℓ, m) is defined as the sum of the expected delivery distance and the distance between the geographical center (x_m, y_m) of the crowdshipper class and the mean origin $(\bar{x}_\ell, \bar{y}_\ell)$ of the ℓ-requests. Tie cases are broken by selecting smaller $N^\phi_{\ell,m}(t)$ for ϕ =HP and SD. The HP and SD policies are constructed through the same steps as RBB-CR (with pseudo-code provided in Fig. 1) except that the RCS pairs (ℓ, m) should be ranked according to the descending order of the monetary incomes p_ℓ and the ascending order of the travel distances, respectively.

In Fig. 2, we demonstrate the effectiveness of the three policies, with respect to the average revenue, average cost, average profit, and rejection rate, against the time horizon, when $\rho = 5$. More precisely, the average revenue and average cost of a policy $\phi \in \Phi$ with time horizon $T \in \mathbb{R}_+$ are

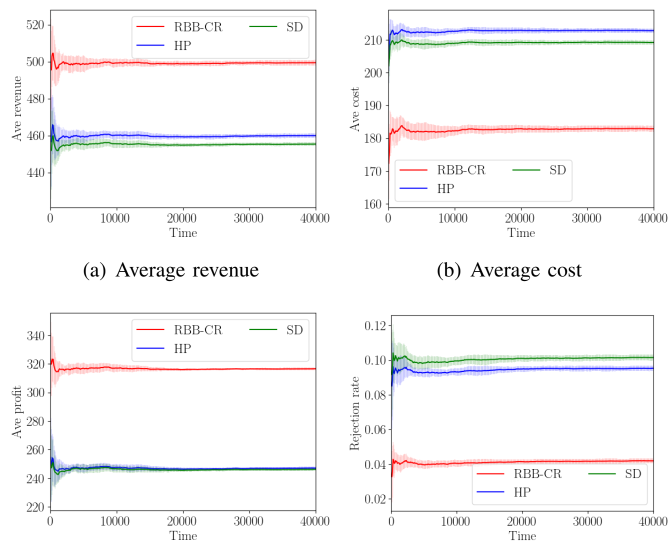
$$\mathfrak{R}^\phi_T := \frac{1}{T}\mathbb{E}\int_0^T \sum_{\ell\in[L]}\sum_{m\in[M]} p_\ell \mu_{\ell,m} N^\phi_{\ell,m}(t)dt, \qquad (16)$$

and

$$\mathfrak{C}^\phi_T := \frac{1}{T}\mathbb{E}\int_0^T \sum_{\ell\in[L]}\sum_{m\in[M]} c_{\ell,m} N^\phi_{\ell,m}(t)dt, \qquad (17)$$

respectively. The revenue $\mathfrak{R}^\phi_T$ is the amount of money that the crowdshipping platform receives from the senders for successfully delivering parcels, while the cost is what the platform pays the crowdshippers for performing deliveries until time T. The average profit $\mathfrak{P}^\phi_T$ is defined as the difference $\mathfrak{R}^\phi_T - \mathfrak{C}^\phi_T$. The rejection rate $\mathfrak{I}^\phi_T$ is the ratio of the number of rejected requests to the total under the policy ϕ during the time period $[0, T]$. It is an indicator of customer dissatisfaction; a higher rejection rate implies more rejected requests and lower customer satisfaction. The objective function in (4) is equal to $\lim_{T\to+\infty} \mathfrak{P}^\phi_T$.

In Fig. 2, RBB-CR always outperforms all the other policies, and the performance quickly becomes stable as T increases; that is, $\mathfrak{P}^{\text{RBB-CR}}_T$ is already close to its long-run average profit for relatively small T. RBB-CR achieves significantly better performance with respect to the four criteria. In particular, it achieves over 25% higher average profits compared to the other policies, implying its effectiveness with respect to the crowdshipping assignment problem described in (4), (2) and (3).



(a) Average revenue (b) Average cost

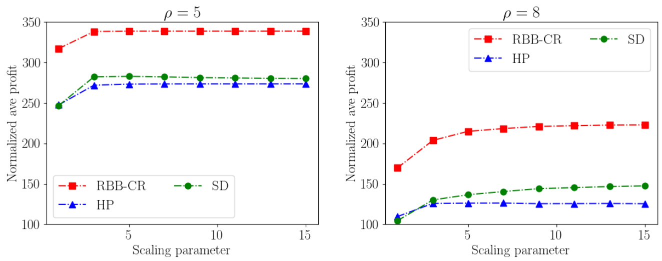
(c) Average profit (d) Average rejection rate

Fig. 2: Performance measures under different policies against time horizon, where the error bars are the standard deviations.

C. Performance Evaluation in Scaled Systems

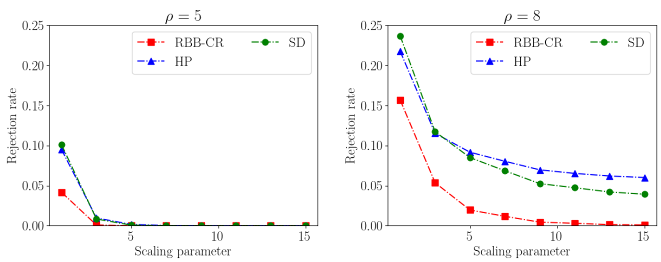
Consider a *scaling parameter* $h \in \mathbb{N}_+$. For class $m \in [M]$, re-consider the number of crowdshippers willing to stay at time t and the arrival rates as $E_m(t) = h\bar{E}^0_m + \Delta_m(t)$ and $\lambda_m = h\lambda^0_m$, respectively, where $\bar{E}^0_m, \lambda^0_m \in \mathbb{R}_+$ are given parameters set equal to $\bar{E}_m$ and λ_m for the simulations discussed in Section V-B. All the other system parameters remain unchanged as those for Section V-B. In this context, we can scale the size of the crowdshipping system with compatibly many delivery requests through h. We refer to such a crowdshipping system scaled with parameter h as the *scaled system*, and the system discussed in Section V-B is a special case with $h = 1$.

In Fig. 3, we compare the long-run average profits normalized by the scaling parameter h (that is, $\lim_{T\to+\infty} \mathfrak{P}^\phi_T/h$) under the three policies ϕ =RBB-CR, HP, and SD for $\rho = 5$ and 8, respectively. Similarly, in Fig. 4, we present the long-run average rejection rates of all the policies for both offered traffic intensities. From the definition, higher ρ implies longer delivery distances for the parcels and a heavier workload for



(a) (b)

Fig. 3: Normalized profits under different policies against the scaling parameter: (a) $\rho = 5$; and (b) $\rho = 8$.



(a) (b)

Fig. 4: Rejection rates under different policies against the scaling parameter: (a) $\rho = 5$; and (b) $\rho = 8$.

the crowdshipping system. The higher delivery distances lead to higher costs and lower profits, and the heavier workload means higher probabilities of rejecting arriving requests, which is consistent with the observations in Figures 3 and 4.

In Figs. 3(b) and 4(b), the normalized average profit and the rejection rate of RBB-CR becomes relatively stable as the scaling parameter h increases. In particular, the rejection rate significantly decreases as h increments from 1 to 10 and maintains almost no change when $h \geq 10$. The normalized profit of RBB-CR in Figs. 3(b) varies similarly with increasing h. It implies that the underlying stochastic system $\boldsymbol{N}^{\text{RBB-CR}}(t)$ has already reached the neighborhood of a stable point–it has already been close to its asymptotic behavior–when h is relatively small. In Figs. 3 and 4, RBB-CR maintains its clear advantages against all the tested baseline policies with respect to both the long-run average profit and the rejection rate.

VI. Conclusions

Effective task assignment is essential for the long-term sustainability of crowdshipping systems. While the extant body of the literature demonstrates several works aiming to maximize the profit of crowdshipping systems [24] [31] [32], they are mainly focused on a point in time. In other words, real-time assignment for profit maximization remains an area for further research. In this paper, we proposed a restless-bandit-based capacity relaxation technique to approximate the marginal profit of the crowdshipping system. We further developed a task assignment strategy, referred to as the RBB-CR policy, that prioritizes the assignment of work to crowdshippers that yield to highest approximated marginal profits. This technique allows for maximizing profits of large-scale crowdshipping problems in real time. Furthermore, we note that most crowdshipping systems consider a single type of delivery. The model proposed in this work accommodates multiple types of deliveries, which brings additional complexity to the model and computation.

To evaluate the performance of the proposed RBB-CR approach, we tested performance indicators of average revenue, cost, profit, and rejection rate, against two baseline greedy policies that prioritize the highest reimbursement and shortest travel distance. Our experimental results demonstrate that the RBB-CR policy outperforms two baseline policies, Highest Price and Shortest Distance, in some cases resulting in 25% higher average profit for the crowdshipping platform. In large-scale testing, similarly, the RBB-CR policy outperforms other baseline policies in terms of long-run average profit and the rejection rate. This performance is attributed to the unique mechanism of RBB-CR to approximate the marginal profit of assigning incoming delivery tasks to available crowdshippers.

Our work also comes with limitations that provide directions for future research. In this research, once the platform assigns a task to crowdshippers, they are not allowed to decline it. Such a feature is not commonly practized in modern crowd logistics systems. This limitation could significantly hinder the willingness of crowdshippers to participate in the system. Therefore, we suggest future research to incorporate task rejection into the model and policy design and further examine how such features could impact profitability and service quality. This consideration could also examine whether higher compensation rates could offset the non-rejection restriction.

Appendix A
Simulation Parameters

For the simulations presented in this paper, as described in Section V-A, we take instances of the random variables for the system parameters. In particular,

- $v_1 = 1$, $(x_1, y_1) = (27.00, 47.67)$, $\bar{E}_1 = 5.48$, and $\bar{c}_1 = 0.47$;
- $v_2 = 2.5$, $(x_2, y_2) = (12.97, 38.87)$, $\bar{E}_2 = 5.83$, and $\bar{c}_2 = 8.82$;
- $v_3 = 2.5$, $(x_3, y_3) = (30.17, 41.09)$, $\bar{E}_3 = 9.64$, and $\bar{c}_3 = 8.82$;
- $v_4 = 1$, $(x_4, y_4) = (46.79, 34.25)$, $\bar{E}_4 = 9.25$, and $\bar{c}_4 = 0.47$;
- $v_5 = 1$, $(x_5, y_5) = (16.93, 19.47)$, $\bar{E}_5 = 7.13$, and $\bar{c}_5 = 0.47$;
- $v_6 = 2.5$, $(x_6, y_6) = (7.38, 42.54)$, $\bar{E}_6 = 8.23$, and $\bar{c}_6 = 8.82$;
- $v_7 = 2.5$, $(x_7, y_7) = (9.78, 30.39)$, $\bar{E}_7 = 9.53$, and $\bar{c}_7 = 8.82$;
- $v_8 = 1$, $(x_8, y_8) = (42.71, 1.42)$, $\bar{E}_8 = 5.15$, and $\bar{c}_8 = 0.47$;
- $v_9 = 1$, $(x_9, y_9) = (9.41, 17.13)$, $\bar{E}_9 = 9.33$, and $\bar{c}_9 = 0.47$;

 ISBN: 978-1-68558-102-2

- and $v_{10} = 2.5$, $(x_{10}, y_{10}) = (24.92, 31.34)$, $\bar{E}_{10} = 9.63$, and $\bar{e}_{10} = 8.82$.

The urgent delivery request (type-1 requests) have $\lambda_1 = 1.49$, $p_1 = 276.34$, $w_{1,m} = 4.25$ for all $m \in \mathscr{M}_1$, $(\bar{x}_1, \bar{y}_1) = (7.38, 42.54)$, and $\mathscr{M}_1 = \{2, 3, 6, 7, 10\}$; and, for the regular requests (type-2 requests), we set $\lambda_2 = 1.17$, $p_2 = 101.36$, $w_{2,m} = 1.56$ for all $m \in \mathscr{M}_2$, $(\bar{x}_1, \bar{y}_1) = (30.17, 41.09)$, and $\mathscr{M}_2 = \{1, 2, 3, 4, 6, 7, 10\}$. Recall that, as described in Section V-A, all the above listed numbers are instances of pseudo-random numbers.

REFERENCES

[1] A. Risberg. and H. Jafari, "Last mile practices in e-commerce: framework development and empirical analysis of swedish firms," *International Journal of Retail & Distribution Managemen*, vol. 50, no. 8/9, pp. 942–961, 2022.

[2] A. Pahwa and M. Jaller, "Assessing last-mile distribution resilience under demand disruptions," *Transportation Research Part E: Logistics & Transportation Review*, vol. 172, p. 103066, 2023.

[3] S. Pan, L. Zhang, R. G. Thompson, and H. Ghaderi, "parcel network flow approach for joint delivery networks using parcel lockers," *International Journal of Production Research*, vol. 59, no. 7, pp. 2090–2115, 2021.

[4] S. Wicaksono, X. Lin, and L. Tavasszy, "Market potential of bicycle crowdshipping: A two-sided acceptance analysis," *Research in Transportation Business & Management*, vol. 45, no. Part A, p. 100660, 2022.

[5] H. Buldeo Rai, S. Verlinde, J. Merckx, and C. Macharis, "Crowd logistics: an opportunity for more sustainable urban freight transport?" *European Transport Research Review*, vol. 9, no. 3, pp. 1–13, 2017.

[6] A. Alnaggar, F. Gzara, and J. Bookbinder, "Crowdsourced delivery: A review of platforms and academic literature," *Omega*, vol. 98, p. 102139, 2021.

[7] A. Fessler, M. Thorhauge, S. Mabit, and S. Haustein, ". a public transport-based crowdshipping concept as a sustainable last-mile solution: Assessing user preferences with a stated choice experiment," *Transportation Research Part A: Policy and Practice*, vol. 158, pp. 210–223, 2020.

[8] H. Ghaderi, L. Zhang, P. Tsai, and J. Woo, "Crowdsourced last-mile delivery with parcel lockers," *International Journal of Production Economics*, vol. 251, p. 108549, 2022.

[9] A. Punel and A. Stathopoulos, "Modeling the acceptability of crowdsourced goods deliveries: Role of context and experience effects," *Transportation Research Part E: Logistics and Transportation Review*, vol. 105, pp. 518–38, 2017.

[10] V. Castillo, D. Mollenkopf, J. Bell, and T. Esper, "Designing technology for on-demand delivery: The effect of customer tipping on crowdsourced driver behavior and last mile performance," *Journal of Operations Management*, vol. 68, pp. 424–453, 2022.

[11] S. S. Mohri, H. Ghaderi, N. Nassir, and R. Thompson, "Crowdshipping for sustainable urban logistics: A systematic review of the literature," *Transportation Research Part E: Logistics and Transportation Review*, p. 178, 2023.

[12] M. Behrend, F. Meisel, K. Fagerholt, and H. Andersson, "A multi-period analysis of the integrated item-sharing and crowdshipping problem," *European Journal of Operational Research*, vol. 292, no. 2, pp. 483–499, 2021.

[13] H. Shen and J. Lin, "Investigation of crowdshipping delivery trip production with real-world data," *Transportation Research Part E: Logistics and Transportation Review*, vol. 143, p. 102106, 2020.

[14] L. Kung and G. Zhong, "The optimal pricing strategy for two-sided platform delivery in the sharing economy," *Transportation Research Part E: Logistics and Transportation Review*, vol. 101, pp. 1–12, 2017.

[15] T. Le, S. Ukkusuri, J. Xue, and T. Van Woensel, "Designing pricing and compensation schemes by integrating matching and routing models for crowd-shipping systems," *Transportation Research Part E: Logistics and Transportation Review*, vol. 149, p. 102209, 2017.

[16] A. Ermagun and A. Stathopoulos, "To bid or not to bid: An empirical study of the supply determinants of crowd-shipping," *Transportation Research Part A: Policy and Practice*, vol. 116, pp. 468–483, 2018.

[17] N. Boysen, S. Emde, and S. Schwerdfeger, "Crowdshipping by employees of distribution centers: Optimization approaches for matching supply and demand," *European Journal of Operational Research*, vol. 296, no. 2, pp. 539–556, 2022.

[18] Y. Lan, F. Liu, W. Ng, M. Gui, and C. Lai, "Multi-objective two-echelon city dispatching problem with mobile satellites and crowd-shipping," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 9, pp. 15 340–15 353, 2022.

[19] J. Fu, B. Moran, and P. G. Taylor, "A restless bandit model for resource allocation, competition and reservation," *Operations Research*, vol. 70, no. 1, pp. 416–431, 2022.

[20] M. Simoni and M. Winkenbach, "Crowdsourced on-demand food delivery: An order batching and assignment algorithm," *Transportation Research Part C: Emerging Technologies*, vol. 149, p. 104055, 2023.

[21] H. Akeb, B. Moncef, and B. Durand, "Building a collaborative solution in dense urban city settings to enhance parcel delivery: An effective crowd model in paris," *Transportation Research Part E: Logistics and Transportation Review*, vol. 119, pp. 223–233, 2018.

[22] J. Fu, L. Zhang, K. Huang, and Z. Liu, "A large-scale multi-rider matching problem with reneging passengers: Single source case," in *the 24th International Congress on Modelling and Simulation (MODSIM2021)*, Sydney, Australia, Dec. 2021, pp. 792–798.

[23] F. Torres, M. Gendreau, and W. Rei, "Vehicle routing with stochastic supply of crowd vehicles and time windows," *Transportation Science*, vol. 56, no. 3, pp. 631–653, 2022.

[24] H. Ghaderi, P. Tsai, L. Zhang, and A. Moayedikia, "An integrated crowdshipping framework for green last mile delivery," *Sustainable Cities and Society*, vol. 78, p. 103552, 2022.

[25] K. Mousavi, M. Bodur, and M. J. Roorda, "Stochastic last-mile delivery with crowd-shipping and mobile depots," *Transportation Science*, vol. 56, no. 3, pp. 612–630, 2022.

[26] K. Gdowska, A. Viana, and J. P. Pedroso, "Stochastic last-mile delivery with crowdshipping," *Transportation research procedia*, vol. 30, pp. 90–100, 2018.

[27] N. P. Farazi, B. Zou, and T. Tulabandhula, "Dynamic on-demand crowdshipping using constrained and heuristics-embedded double dueling deep q-network," *Transportation Research Part E: Logistics and Transportation Review*, vol. 166, p. 102890, 2022.

[28] J. Dötterl, R. Bruns, J. Dunkel, and S. Ossowski, "Evaluating crowdshipping systems with agent-based simulation," in *Multi-Agent Systems and Agreement Technologies*, N. Bassiliades, G. Chalkiadakis, and D. de Jonge, Eds. Cham: Springer International Publishing, 2020, pp. 396–411.

[29] G. Lee, W. Saad, and M. Bennis, "An online secretary framework for fog network formation with minimal latency," in *2017 IEEE International Conference on Communications (ICC)*, May 2017, pp. 1–6.

[30] The University of Melbourne, "Spartan," May 2023. [Online]. Available: https://dashboard.hpc.unimelb.edu.au/

[31] L. Dahle, H. Andersson, M. Christiansen, and M. Speranza, "The pickup and delivery problem with time windows and occasional drivers," *Computers and Operations Research*, vol. 109, pp. 122–133, 2019.

[32] C. Triki, "Using combinatorial auctions for the procurement of occasional drivers in the freight transportation: A case-study," *Journal of Cleaner Production*, vol. 304, p. 127057, 2021.