Parameter-Dependent Competitive Analysis for Online Capacitated Coverage Maximization through Boostings and Attenuations

Pan Xu [1]

Abstract

In this paper, we consider a model called *Online Capacitated Coverage Maximization*, characterized by two features: (1) the dynamic arrival of online agents following a known identical and independent distribution, and (2) each offline agent is associated with a specific coverage valuation over the groundset of online agents. Additionally, both offline and online agents are assigned integer capacities, reflecting finite budgets and operational constraints. We introduce and analyze two matching policies. The first, a non-adaptive policy, utilizes offline statistics derived from solving a benchmark linear program. The second is an enhanced version equipped with real-time boostings and attenuations. We conduct a comprehensive competitive analysis and characterize the competitive ratio for both policies as functions of two crucial parameters: a lower bound on the matching capacity among offline agents and an upper bound on the number of online agents covering any specific feature for offline agents.

1. Introduction

Matching markets involve heterogeneous agents (typically from two parties) who are paired for mutual benefits. During the last decade, matching markets have emerged and proliferated through the Internet. They have evolved into a new style, called *Online Matching Markets* (OMMs), with examples ranging from crowdsourcing to Internet advertising. There are two features distinguishing OMMs from traditional matching markets. The first feature is that users from at least one party join the system dynamically, which are referred to as *online agents* (**F1**). Examples include keywords in online advertising and workers in Amazon Mechanical Turk (AMT). As opposed to online agents, the other party is called *offline agents*, such as sponsors in online advertising and tasks in AMT. The information of offline agents is known as *a priori*, and they are assumed static. The second feature is the real-time decision-making requirement (**F2**). It is highly desirable to match each online agent with one (or multiple) offline agent(s) upon its arrival due to online agents' low "patience". Generally, each match of offline and online agents contributes a certain amount of revenue to the system. Consequently, a well-studied topic in OMMs is to maximize the total revenue, which is modeled as a linear function over all matches; see, *e.g.,* (Ashlagi et al., 2019; Bei & Zhang, 2018; Dickerson et al., 2024; 2021).

In this paper, we consider a general setting introduced by (Xu, 2023b) when the linear-maximization objective is replaced by a *(weighted) coverage maximization*. One motivating example is called *online multi-skilled task-worker assignment problem* in crowdsourcing markets, where each task and worker is associated with a specific set of skills (Barnabò et al., 2019; Cheng et al., 2016; Anagnostopoulos et al., 2012). A natural goal there is to assign each task a set of workers such that the task has as many skills covered as possible. In a general scenario, each task-skill pair is associated with a specific positive weight reflecting the importance or priority of the skill for the task. In that case, the resulting objective is formulated as a maximization of a weighted coverage function. Similar issues exist in other gig platforms such as Upwork, where we need to enlist a diverse team of workers for each project (task) such that as many expertises are covered as possible that are requested by the project. Observe that the two aforementioned features (**F1** and **F2**) distinguish our problem from the classical (offline) coverage maximization and pose significant challenges in the matching-policy design.

1.1. Main Model

For the ease of presentation, we include a brief description of a slightly simplified version of the model proposed in (Xu, 2023b), where only the offline side is capacitated. In Appendix H, we will demonstrate how to generalize the current techniques and results to the two-sided capacitated case, which is the exact setting in (Xu, 2023b). Throughout

[1]Department of Computer Science, New Jersey Institute of Technology, Newark, USA. Correspondence to: Pan Xu <pxu@njit.edu>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

this paper, we refer to our model as *Online Capacitated Coverage Maximization* (**OCCM**) using the terminology of a multi-skilled task-worker matching problem in a typical crowdsourcing market, where "task" and "worker" refer to generic offline and online agents, respectively.

Suppose there is a bipartite graph $G = (I, J, E)$, where I and J denote the set of types of offline tasks and online workers, respectively. An edge $e = (i, j)$ indicates that worker (of type) j shows interest in the task (of type) i. We have a ground set $\mathcal{K}$ of K skills, and each task i and worker j are labeled with a binary vector $\boldsymbol{\chi}_i, \boldsymbol{\chi}_j \in \{0,1\}^K$, where $\chi_{ik} = 1$ and $\chi_{jk} = 1$ indicate that skill k is requested by task i and possessed by worker j, respectively. For each task-skill pair $\lambda = (i, k)$, it is associated with a weight $w_\lambda \geq 0$, which captures the value or priority of skill k to task i. We can simply assume $w_\lambda = 0$ for all those pairs $\lambda = (i, k)$ with $\chi_{ik} = 0$. Each task i has a matching capacity $b_i \in \mathbb{Z}^+$, *i.e.,* we can allocate at most b_i workers to it.[1] Upon the arrival of an online worker j, an immediate and irrevocable decision is required: either reject it or assign it to a task i with $(i, j) \in E$ that has remaining capacity. Our overall goal is to maximize the total weight of covered skills over all tasks. Below is a detailed description of the arrival setting of online workers.

KIID Arrivals of Online Workers. We consider a finite time horizon T. During each time (or round) $t \in [T] \doteq \{1, 2, \ldots, T\}$, one single worker (of type) $\hat{j}$ will be sampled (or $\hat{j}$ arrives) with replacement such that $\Pr[\hat{j} = j] = r_j/T$ for all $j \in J$ with $\sum_{j \in J} r_j/T = 1$. Here r_j is called the arrival rate of worker j, which is equal to the expected number of total arrivals of j during the T rounds. Note that the arrival distribution $\{r_j/T\}$ is assumed independent and invariant throughout the online phase. It is commonly referred to as the *known identical independent distributions* (KIID). This is mainly inspired by the fact that we can often learn the arrival distribution from historical logs (Yao et al., 2018; Li et al., 2018; Wang et al., 2018). KIID is widely used in practical applications of OMMs, including rideshare and crowdsourcing markets (Zhao et al., 2019; Dickerson et al., 2024; Fata et al., 2019). In this paper, we consider the integral arrival setting where workers' arrival rates are all integers. WLOG we assume that $r_j = 1$ for each $j \in J$ such that $T = |J|$ by creating r_j copies for each worker (type) j.

Consider a matching policy ALG (possibly randomized) and an allocation $\mathbf{X} = (X_{ij}) \in \{0,1\}^{|E|}$ output by ALG, where $X_{ij} = 1$ with $e = (i, j) \in E$ means e is matched, *i.e.,* j is assigned to i.[2] For each $i \in I$, let $\mathcal{N}_i = \{j \in J, (i, j) \in E\}$ be the set of neighbors of i; similarly for $\mathcal{N}_j$ with $j \in J$. We say $\mathbf{X}$ is *feasible* or *valid* iff $\sum_{j \in \mathcal{N}_i} X_{ij} \leq b_i$ for all $i \in I$. We define the utility of task i under $\mathbf{X}$ as $\mathsf{w}_i(\mathbf{x}) = \sum_{k \in \mathcal{K}} w_{ik} \cdot \min(1, \sum_{j \in \mathcal{N}_i} X_{ij} \cdot \chi_{jk})$, *i.e.,* the total sum of weights on skills covered under $\mathbf{X}$, and we set the resulting total utility under $\mathbf{X}$ as $\mathsf{w}(\mathbf{X}) = \sum_{i \in I} \mathsf{w}_i(\mathbf{X})$, *i.e.,* the sum of utilities over all tasks. An input instance of **OCCM** can be characterized as $\mathcal{I} = \{G = (I, J, E), \{b_i, w_{ik}, \boldsymbol{\chi}_i, \boldsymbol{\chi}_j | i \in I, k \in \mathcal{K}, j \in J\}, T\}$, and we assume all information there is accessible to the algorithm. Our goal is to design an allocation policy ALG such that it always outputs a feasible allocation $\mathbf{X}$ with $\mathsf{E}[\mathsf{w}(\mathbf{X})]$ being as large as possible. Here the expectation is taken over the randomness in the workers' dynamic arrivals and that potentially used in ALG. *Throughout this paper, we assume $T \gg \max_{i \in I} b_i \geq 1$,[3] and part of our results are obtained by taking $T \to \infty$, a common practice in studying competitiveness for online-matching algorithms under KIID* (Huang & Shu, 2021; Brubach et al., 2020; Jaillet & Lu, 2013; Manshadi et al., 2012; Haeupler et al., 2011; Feldman et al., 2009).

1.2. Preliminaries

Competitive Ratio (CR). CR is a common metric to evaluate the performance of online algorithms. Consider an instance $\mathcal{I}$ of **OCCM** as studied here, for example. Let $\mathsf{ALG}(\mathcal{I}) = \mathsf{E}_{S \sim \mathcal{I}}[\mathsf{ALG}(S)]$ denote the performance of ALG on the instance $\mathcal{I}$, where the expectation is taken over the randomness in the arrival sequence S of online workers and possible randomness in ALG. Let $\mathrm{OPT}(\mathcal{I}) = \mathsf{E}_{S \sim \mathcal{I}}[\mathrm{OPT}(S)]$ denote the performance of an offline optimal (*a.k.a.* a clairvoyant optimal), denoted by OPT, where $\mathrm{OPT}(S)$ refers to the value obtained by OPT, which has the privilege of accessing the full arrival sequence S before any actions. We say ALG achieves a CR of at least $\rho \in [0, 1]$ if $\mathsf{ALG}(\mathcal{I}) \geq \rho \cdot \mathrm{OPT}(\mathcal{I})$ for any input instance $\mathcal{I}$.

Benchmark Linear Program (LP). Throughout this paper, we denote as OPT a clairvoyant optimal policy (and its corresponding performance). Let $\Lambda \doteq I \times \mathcal{K}$ be the collection of all task-skill pairs. For each edge $e = (i, j)$ and $\lambda = (i, k) \in \Lambda$, let x_e and y_λ be the probabilities that e is matched (*i.e.,* j is assigned to i) and that λ gets matched or covered (*i.e.,* skill k is covered for task i) in OPT, respectively.[4] Recall that for each node $\ell \in I \cup J$, $\mathcal{N}_\ell$ is the set of

[1]The matching capacity is motivated by the limited budget allocated to each task in practice.

[2]Note that for each task, the weight on each covered skill is counted only once. That is why we assume WLOG that any policy will match any edge at most once.

[3]This is motivated by the fact that each task has a limited number of copies or is allocated a small budget, such that we can afford to recruit only a small portion of the arriving workers.

[4]Here we can assume WLOG that each edge gets matched at most once in OPT since we focus on coverage maximization. This is one of the key differences of our model from the online b-matching problem, where matching an edge multiple times yields the same number of copies of rewards.

Table 1. A glossary of notations used throughout this paper.

$[n]$	Set of $\{1, 2, \ldots, n\}$ for an integer n.	I (J)	Set of task (worker) types.
$\mathcal{N}_i$ $(\mathcal{N}_j)$	Set of neighbors of $i \in I$ $(j \in J)$.	$\mathcal{K}$	Set of skills with $\lvert\mathcal{K}\rvert = K$.
$\lambda = (i, k)$	Pair of task i and skill k.	$w_{\lambda=(i,k)}$	Weight of skill k to task i.
$\mathcal{N}_{\lambda=(i,k)}$	Set of i's neighbors covering k.	Δ, τ	$\Delta = \max_\lambda \lvert\mathcal{N}_\lambda\rvert$, $\tau = 1 - \mathsf{e}^{-\Delta}$.
b	Uniform capacity on all tasks.	T	Total number of online rounds.
ATT-I	First type of attenuations.	ATT-II	Second type of attenuations.
RTB	Real-Time boosting.	CR	Competitive Ratio.
SM-A	Non-adaptive policy in Alg. 1.	SM-B	Adaptive policy in Alg. 2.
e (Italic)	Edge $e \in E$.	e (Non-italic)	Natural base with $\mathsf{e} \sim 2.718$.
AG	$\mathsf{AG} = \eta - \kappa$, Adaptivity Gap.	$\mathsf{Ber}(\cdot), \mathsf{Pois}(\cdot)$	Bernoulli and Poisson random variables.

neighbors of ℓ. For each $\lambda = (i, k)$, let $\mathcal{N}_\lambda \subseteq \mathcal{N}_i$ be the set of i's neighbors covering skill k and w_λ be the weight of skill k with respect to task i. Set $\Delta := \max_{\lambda \in \Lambda} \lvert\mathcal{N}_\lambda\rvert$ and $\tau := 1 - \mathsf{e}^{-\Delta}$, where Δ is a parameter capturing the maximum number of worker types that can cover any specific skill among all those showing interest in the task.[5] Consider the below LP.

$$\max \sum_{\lambda \in \Lambda} w_\lambda \cdot y_\lambda \tag{1}$$

$$y_\lambda \le 1 - \mathsf{e}^{-\Delta} = \tau \qquad \forall \lambda \in \Lambda \tag{2}$$

$$y_\lambda \le x_\lambda := \sum_{j \in \mathcal{N}_\lambda} x_{ij} \qquad \forall \lambda = (i, k) \in \Lambda \tag{3}$$

$$x_i := \sum_{j \in \mathcal{N}_i} x_{ij} \le b_i \qquad \forall i \in I \tag{4}$$

$$x_j := \sum_{i \in \mathcal{N}_j} x_{ij} \le 1 \qquad \forall j \in J \tag{5}$$

$$0 \le x_e \le 1 - 1/\mathsf{e}, y_\lambda \le 1 \quad \forall e \in E, \lambda \in \Lambda. \tag{6}$$

We refer to the above LP as LP (1) throughout this paper.

Lemma 1. *The optimal value of* LP (1) *is a valid upper bound on the performance of a clairvoyant optimal (*OPT*).*

Proof. By definition of $\{x_e, y_\lambda\}$, we can verify that Objective (1) encodes the exact expected performance of OPT. To prove the above lemma, it suffices to show the validity of all constraints in LP (1) with respect to any policy captured by $\{x_e, y_\lambda\}$. Constraint (2) is valid since for each task-skill pair $\lambda = (i, k)$, λ will never get covered when none of the workers in $\mathcal{N}_\lambda$ has ever arrived once, which occurs with probability equal to $\mathsf{e}^{-\lvert\mathcal{N}_\lambda\rvert} \ge \mathsf{e}^{-\lvert\Delta\rvert}$. This implies that λ is covered with probability no more than $1 - \mathsf{e}^{-\lvert\Delta\rvert}$. Constraint (3) is reasonable since $\lambda = (i, k)$ gets covered iff at least one edge $e = (i, j)$ with $j \in \mathcal{N}_\lambda$ gets matched, which happens with probability at more $\sum_{j \in \mathcal{N}_\lambda} x_{ij}$. Constraint (4) is due to the matching capacity of b_i on task i, while Constraint (5) follows from that the expected number of arrivals for each worker (of type) j is one. As for Constraint (6), observe that $e = (i, j)$ gets matched at most once in OPT, and it happens only when j arrives at least once that occurs with probability equal to $1 - 1/\mathsf{e}$. ☐

Connections to Existing Models. We consider the exact model (**OCCM**) introduced by (Xu, 2023b), which is featured by (i) each offline agent is associated with a coverage valuation and (ii) both offline and online agents can each have an arbitrary integer matching capacity. Kapralov et al. (2013) considered Online Submodular Welfare Maximization (**OSWM**) under the KIID setting, where each offline agent is associated with a general monotone submodular valuation and each has an unbounded capacity, while every online agent has a unit capacity. Notably, the setting of possibly finite capacity among offline agents makes the model **OCCM** essentially different from **OSWM**. As pointed out in (Xu, 2023b), the natural Greedy (GRY) turns out to be zero-competitive for **OCCM**, while GRY is shown to achieve an optimal competitive ratio of $1 - 1/\mathsf{e}$ for **OSWM** (Kapralov et al., 2013). In Table 2, we list a few other works that have considered online submodular/coverage maximization under KIID arrival settings.

1.3. Overview of Algorithms and Techniques

Throughout this paper, when we say "at time t," we mean "at the very beginning of t before any online actions." We propose and carefully analyze two natural policies. The first one is non-adaptive (SM-A), which is simply guided by offline statistics through solving the benchmark LP (1); while the other is a fortified version armed with real-time boostings (SM-B). For each policy, we give a comprehensive competitive analysis and characterize the final CR as a function of two key parameters; see details in Section 1.7.

A Non-Adaptive Policy SM-A (**Algorithm 1**). SM-A samples a task i upon each arrival of a worker j following a

[5] Observe that $\Delta \le \max_{i \in I} \lvert\mathcal{N}_i\rvert \le \lvert J\rvert$ since $\mathcal{N}_\lambda \subseteq \mathcal{N}_i$ with $\lambda = (i, k)$. In many real-world applications, Δ typically takes a small value that can be far less than $\max_i \lvert\mathcal{N}_i\rvert$, especially when a large number of skills involved (Ahmed et al., 2020).

Table 2. Comparison of settings in recent works related to online coverage/submodular maximization under the KIID (Known Independent and Identical Distributions) arrival setting. In the **second column**, "OBJ" represents the objective type, while "SUB" and "COV" denote *monotone submodular* and *coverage maximization*, respectively. In the **third and fourth columns**, "b" represents a uniform matching capacity among offline agents, and "b'" represents an upper bound on the matching capacity among online agents. The entry "$\mathbb{N} \cup \infty$" means any fixed positive integer or infinity. In the **fifth column**, we classify algorithms into two classes based on whether the sampling distributions in the online phase are adapted to the arrival time of each online agent. All results in this paper are highlighted in blue.

Models	OBJ	b	b'	Algorithms	CR	Upper Bounds
(Kapralov et al., 2013)	SUB	∞	1	Greedy (Adaptive)	0.632	0.632
(Dickerson et al., 2019b)	SUB[a]	1	1	Non-Adaptive	0.399	–
(Xu, 2023b)	COV	$\mathbb{N} \cup \{\infty\}$	$\mathbb{N}$	Non-Adaptive	0.580	–
	SUB	$O(1)$	$\mathbb{N}$		0.436	
(This paper)	COV	$\mathbb{N} \cup \{\infty\}$	$\mathbb{N}$	Adaptive	$\eta(\tau, b) \geq 0.602$ [b]	
					$\eta(1, \infty) = 0.632$	0.632
					$\eta(\tau, 1) = 0.692$	0.729
					$\eta(1 - 1/\mathrm{e}, \infty) = 0.741$	0.896

[a] They considered a slightly different version of the objective, which is to maximize one single monotone submodular function defined over the groundset of all edges.

[b] We offer a comprehensive competitive analysis and express the final competitive ratio (CR) as a function $\eta(\tau, b)$, as defined in (8), where $\tau = 1 - \mathrm{e}^{-\Delta}$ with Δ being the maximum number of online agents covering any specific features/skills, and b representing uniform capacity among offline agents. The inequality $\eta(\tau, b) \geq 0.602$ holds for all possible combinations of (τ, b), which suggests a strict improvement over that of 0.580 for online coverage maximization due to (Xu, 2023b). For the setting of $(\Delta, \tau) = (\infty, 1)$ and $b = \infty$, our result matches the upper bound of $1 - 1/\mathrm{e} \sim 0.632$ (Kapralov et al., 2013), which suggests its optimality on that case. We also manage to identify upper bounds when $b = 1$ or $(\Delta, \tau) = (1, 1 - 1/\mathrm{e})$ based on the benchmark LP; see more details in Appendix F.

static distribution $\mathcal{D}_j := \{x^*_{ij} | i \in \mathcal{N}_j\}$, where $\{x^*_{ij}\}$ is an optimal solution to benchmark LP (1). Note that $\mathcal{D}_j$ is valid since $\sum_{i \in \mathcal{N}_j} x^*_{ij} \leq 1$ due to Constraint (5) of LP (1). Note that SM-A shares the essence with those proposed in (Esmaeili et al., 2023; Xu, 2023b; Dickerson et al., 2019b): They are all non-adaptive, meaning that the sampling distributions are solely determined by the arriving online agents themselves and remain unaffected by their arrival time.

An Adaptive Policy SM-B **(Algorithm 2)**. SM-B consists of two types of attenuations and one type of boostings. We carefully craft two auxiliary sequences, namely, $\{\phi_t, \psi_t | t \in [T]\}$ defined in (11), to guide the two types of attenuations. The first type of attenuation (ATT-I) applies to random events that tasks stay safe. A task is called *safe* at time t iff it has at least one remaining capacity then. The goal of ATT-I is to ensure that each task i is safe at t, denoted by ($\mathsf{SF}_{it} = 1$), with probability *equal* to $\mathsf{E}[\mathsf{SF}_{it}] = \phi_t$. The idea of Real-Time Boostings (RTB) is that upon every arrival of online worker j at time t, we sample one from the set of j's *safe* neighbors at t only, denoted by $\mathcal{N}_{j,t}$. Specifically, the boosted sampling distribution upon j's arrival is captured as $\widetilde{\mathcal{D}}_{j,t} = \{x^*_{ij} / \sum_{\tilde{i} \in \mathcal{N}_{jt}} x^*_{\tilde{i}j} | i \in \mathcal{N}_{jt}\}$. The second type of attenuation (ATT-II) applies to random events of tasks getting matched at t after RTB such that each safe task $i \in \mathcal{N}_{j,t}$ gets sampled *and* matched with j at time t with probability *equal* to $\psi_t \cdot x^*_{ij}/T$. The improvement of the performance of SM-B over SM-A comes from the key idea of RTB. Note that the size of the random set $\mathcal{N}_{j,t}$ gets reduced as time t since more and more of j's neighbors have their capacity exhausted and then become unsafe, which leads to $\widetilde{\mathcal{D}}_{j,t}$ gets boosted over time. The two types of attenuations are introduced mainly to make the algorithm SM-B with boostings function well.

In Appendix A, we utilize the example presented in (Xu, 2023b) to illustrate the distinctions between Greedy (GRY), which has been established as optimal for Online Submodular Welfare Maximization (Kapralov et al., 2013), and the two policies SM-A and SM-B, as studied here.

1.4. Motivation for the Second Algorithm SM-B

Observe that our model captures, as a strictly special case, edge-weighted online matching when every task and worker has a unit matching capacity. In the following, we use this simple model to illustrate the motivation behind SM-B. In algorithm design for (edge-weighted) online stochastic optimization, balancing the performance among all edges is a vital and challenging issue. In this case, we choose to lower bound the competitiveness of an algorithm (ALG), as $\min_e \sigma_e := \mathsf{E}[Y_e]/y_e^*$, where $\mathsf{E}[Y_e]$ represents the probability that edge e is matched in ALG and y_e^* is the optimal solution in an LP benchmark denoting the probability that e is matched in an OPT.

In most cases, when evaluating a matching policy, we often encounter a bottleneck scenario where the lowest possible ratio is achieved, denoted as $e^* = \mathsf{argmin}_e \sigma_e$, arrives when e^* neighbors a few edges with relatively large values of σ_e. This phenomenon arises because neighboring edges are always competing with each other. Consequently, when an adversary orchestrates a worst-case scenario for e^*, it typically arranges numerous strong neighbors to "bully" edge e^* to an extreme degree. This underscores the importance of balancing the performance of all edges. Intuitively, by suppressing the performance of neighboring edges of e^*, we can enhance that of e^*, resulting in improved competitiveness. This is part of our motivation behind the carefully-designed attenuations.

In our context, "edges" should be interpreted as task-skill pairs, and thus, the balancing act involves coordinating events of all task-skill pairs getting covered. Note that the event that a given task-skill pair $\lambda = (i, k)$ gets covered is jointly determined by two factors. The first is whether task i is safe when a worker j with skill k arrives. The second is whether we sample the neighboring task i for worker j upon its arrival, given that task i is safe. The two types of attenuation are proposed to target these two events, respectively, ensuring that the overall matching probability of every task-skill pair getting covered at any time is equal to a pre-arranged value. From the perspective of worst-case analysis, we expect that the two types of attenuations will effectively suppress those strong "neighbors" of the bottleneck task-skill pair, which in turn boosts the worst performance and results in higher competitiveness.

1.5. Comparison against the Work by (Ma et al., 2023)

We detail the differences between ours and the work (Ma et al., 2023) in the following aspects.

Model. Ma et al. (2023) considered an online matching model under KIID with the objective of maximizing fairness among offline agents. Specifically, they formulated the objective in the benchmark LP as $\max \min_{i \in I}(x_i := \sum_{j \in N_i} x_{ij})$, which can be interpreted as maximizing the minimum matching probability among all offline agents. This objective can be directly reduced to a linear objective of $\max \rho$ together with constraints $\rho \leq x_i$ for all $i \in I$. Additionally, Ma et al. (2023) assumed unit matching capacity for every offline and online agent.

Approach and Analysis. Several features in the model of (Ma et al., 2023) significantly simplify the matching algorithm design and analysis compared to ours. First, the linear objective of maximizing fairness among all offline agents enables them to conduct a competitive analysis for each offline vertex. This means they can lower bound the competitiveness of a policy (ALG) as $\min_i \mathsf{E}[X_i]/x_i^*$, where $X_i = 1$ indicates that i is matched in ALG and x_i^* denotes an optimal LP solution on i. Furthermore, the benchmark LP in (Ma et al., 2023) enjoys an exclusive property that allows them to assume, without loss of generality, that $x_i^* = \rho^*$ for every $i \in I$. This further simplifies the analysis. For this reason, they introduce a single layer of attenuation, which targets offline agents only and ensures that every agent is safe equal to a preset value. In contrast, we conduct an analysis for each task-skill pair, which requires us to introduce two layers of attenuations targeting two kinds of events: whether a task is safe at any time and whether a task gets sampled upon a worker's arrival, respectively. Moreover, we have to carefully correlate the two layers of attenuations so that they can work properly under real-time boostings. Second, Ma et al. (2023) assumed unit capacity for all offline and online agents. Therefore, the single-layer attenuation sequence there involves only one factor of time. In contrast, the two-layer attenuation sequences here involve both time and capacity. Meanwhile, in order to incorporate the two parameters into the final competitiveness result, we have to resort to more advanced math tools such as ordinary differential equations, which are absent in (Ma et al., 2023) since they just need to give a parameter-free analysis.

1.6. Comparison against the Work of (Xu, 2023b)

We consider the exact model (**OCCM**) introduced by Xu (2023b). That being said, Xu (2023b) presented only one non-adaptive algorithm, which shares the essence with $\mathsf{SM\text{-}A}$ in Algorithm 1. They conducted a parameter-free competitive analysis and showed that $\mathsf{SM\text{-}A}$ achieves a competitiveness of 0.580. This result is recovered by our analysis; see details in the paragraph titled "**Remarks on Theorem 1**" below the statement of Theorem 1.

Note that for the same algorithm $\mathsf{SM\text{-}A}$, Xu (2023b) analyzed its competitiveness directly by focusing on the worst-scenario setting ($\tau = 1$ and $b = 3$) since they did not need to conduct a parameter-dependent analysis as we do here. In contrast, we analyze $\mathsf{SM\text{-}A}$ by casting it as a strictly special case of another adaptive algorithm ($\mathsf{SM\text{-}B}$); see details in the proof of Theorem 1 in Appendix C. In other words, we do not analyze $\mathsf{SM\text{-}A}$ directly; instead, we obtain the competitiveness function $\kappa(\tau, b)$ of $\mathsf{SM\text{-}A}$ as a byproduct while evaluating the exact competitiveness function $\eta(\tau, b)$ for $\mathsf{SM\text{-}B}$. The competitive analysis of $\mathsf{SM\text{-}B}$ is much more complicated than that analysis of $\mathsf{SM\text{-}A}$ in (Xu, 2023b): It is caused not only by the adaptivity and the two layers of attenuations in $\mathsf{SM\text{-}B}$ (note that $\mathsf{SM\text{-}A}$ is non-adaptive with no attenuations and it has a lower competitiveness, i.e., $\kappa \leq \eta$ in Theorem 3), but also by the need for a parameter-dependent analysis. Consequently, we introduce more advanced techniques, such as second-order ordinary differential equations, which are completely absent in (Xu, 2023b).

1.7. Main Contributions

Recall that $\Delta = \max_{\lambda \in \Lambda} |\mathcal{N}_\lambda|$ and $\tau = 1 - \mathsf{e}^{-\Delta}$. For the ease of presentation, we assume every task takes a uniform capacity $b_i = b$ with $b \ll T$.[6] Below are our main theoretical results.

Theorem 1. *[Section 2 and Appendix C] The non-adaptive algorithm* SM-A *(Algorithm 1) achieves a competitive ratio (CR) of* $\kappa(\tau, b)$, *where*

$$\kappa(\tau, b) = \int_0^1 dz \Pr\Big[\text{Pois}\big((b-\tau)z\big) \le b-1\Big]\mathsf{e}^{-z\cdot\tau}. \quad (7)$$

Remarks on Theorem 1. Note that the non-adaptive algorithm SM-A shares the essence with that in (Xu, 2023b). Numerical values in Table 4 in Appendix E suggest that $\kappa(\tau, b) \ge \kappa(1, 3) \sim 0.580$ for any $\tau \in [0, 1]$ and $b \ge 1$, where the worst setting arrives at $(\Delta, \tau) = (\infty, 1)$ and $b = 3$. This recovers the competitiveness result claimed in (Xu, 2023b). In Appendix C, we present a simple proof of Theorem 1 by casting κ as a special case of η, which captures the CR of the adaptive algorithm SM-B (as shown in Theorem 2 below). Thus, our analysis offers an alternative competitive analysis to that in (Xu, 2023b).

Theorem 2. *[Section 3 and Appendix B] The adaptive algorithm* SM-B *(Algorithm 2) achieves a competitive ratio (CR) of* $\eta(\tau, b)$, *where*

$$\eta(\tau, b) = \int_0^1 dz \cdot \Big(h'(z)\cdot \mathsf{e}^{-\tau\cdot h(z)} \cdot \cdot \Pr\Big[\text{Pois}\big((b-\tau)\cdot h(z)\big) \le b-1\Big]\Big), \quad (8)$$

and where $h(z)$ *is the function satisfying*

$$h'' = (h')^3 h^{b-1}\mathsf{e}^{-bh-1}\frac{b^b}{(b-1)!}, h(0) = 0, h'(0) = 1. \quad (9)$$

For each given integer $b \ge 1$, we can get an analytical form of $h(z)$; see the example when $b = 2$ in Equality (16) in Section 3.1. More examples of h are offered in Appendix E.

[6]When tasks take distinct capacities, our results continue to hold after resetting $b := \min_i b_i$ and assuming b takes a reasonably large value. Note that the CR functions of κ and η in (7) and (8) are not universally increasing over all positive integers of b, but they are indeed when b is slightly large, *e.g.,* $b \ge 10$. In practical settings, b typically takes a small finite value since each task is allocated a small budget that can afford to recruit a certain number of workers.

Theorem 3. *[Section D] (1) For any given* $b \ge 1$, *both* $\kappa(\tau, b)$ *and* $\eta(\tau, b)$ *are non-increasing over* $\tau \in [0, 1]$; *(2) For any given* $b \ge 1$ *and* $\tau \in [0, 1]$, $\eta(\tau, b) \ge 0.602$ *and* $\eta(\tau, b) \ge \kappa(\tau, b) = (1 - \mathsf{e}^{-\tau})/\tau - \Theta(b^{-1/2})$, *where* $\Theta(b^{-1/2}) = c \cdot b^{-1/2}(1 + o(1))$ *with* $c \in [\sqrt{1/(2\pi\mathsf{e}^2)}, \sqrt{2/\pi}]$ *being a constant and* $o(1)$ *vanishing as* $b \to \infty$.

Remarks on Theorem 3. (1) The second claim in Theorem 3 implies that both SM-A and SM-B achieve a CR of at least $(1 - \mathsf{e}^{\tau})/\tau$ when $b \to \infty$, which further approaches $1 - 1/\mathsf{e}$ when $(\Delta, \tau) \to (\infty, 1)$. Meanwhile, Kapralov et al. (2013) offered an instance of Online Capacitated Coverage Maximization (**OCCM**) under KIID showing that no algorithm can beat $1 - 1/\mathsf{e}$ when both b and Δ go to unbounded. *This suggests the optimality of both* SM-A *and* SM-B *when* b *and* Δ *are both unrestricted.* (2) The inequality $\eta(\tau, b) \ge 0.602$ holds for all possible combinations of (τ, b), which suggests a strict improvement on that of 0.580 for **OCCM** due to (Xu, 2023b).

We acknowledge the challenges in deriving a universal hardness result (upper bound) for Online Capacitated Coverage Maximization (**OCCM**) as a function of (τ, b) for any $\tau \in [0, 1]$ and $b \in \mathbb{Z}^+$. This is in contrast to what we have accomplished in lower bounding the CR for SM-A and SM-B. Nevertheless, we initiate the hardness analysis with a special case when $(\Delta, \tau) = (1, 1 - 1/\mathsf{e})$.

Theorem 4. *[Appendix F] For* ***OCCM*** *with* $(\Delta, \tau) = (1, 1 - 1/\mathsf{e})$, *no algorithm can achieve a competitive ratio better than* $\bar{\eta}(b)$ *using* LP (1) *as a benchmark, where*

$$\bar{\eta}(b) := \mathsf{E}\Big[\min\Big(1, \frac{1}{2b}\sum_{j=1}^{2b}\min(2, \mathsf{Pois}(1))\Big)\Big]. \quad (10)$$

Remarks on Theorem 4. (1) We can verify that $\bar{\eta}(1) = 1 - 2/\mathsf{e}^2 \sim 0.729$ and $\bar{\eta}(\infty) = \lim_{b\to\infty} \bar{\eta}(b) = 2 - 3/\mathsf{e} \sim 0.896$.(2) The above hardness result suggests that **OCCM** (with $\Delta = 1$) differs significantly from classical online b-matching problems. There are a few studies showing that the same non-adaptive policy SM-A achieves a CR of $1 - o(1)$ for online b-matching under KIID using an even weaker benchmark than LP (1), where $o(1)$ is a vanishing term when $b \to \infty$ (Brubach et al., 2016; Xu, 2023a). This contrasts with the fact that no algorithm can achieve a CR better than $\bar{\eta}(\infty) \sim 0.896$ for **OCCM**, even as $b \to \infty$. We provide further clarification on the differences between **OCCM** (with $\Delta = 1$) and online b-matching problems in Appendix G.

Numerical Evaluations of the Adaptivity Gap. Inspired by works investigating adaptivity gaps for (offline) stochastic optimization problems (Bradac et al., 2019; Dean et al., 2008), we study the issue in the context of online optimization. Let $\mathsf{AG} = \eta - \kappa$, which is referred to as *Adaptivity Gap* that captures the improvement in competitive ratio (CR) brought by real-time boostings. We numerically evaluate κ, η, and AG, when parameters (τ, b) take different values; see details in Table 3. Due to the space limit, we list only a few examples and defer a full version to the Appendix. Throughout the main body of the paper, we maintain the accuracy to the third decimal place for fractional values.

Table 3. Values of (κ, AG) under different settings of (τ, b) with $\tau = 1 - \mathrm{e}^{-\Delta}$ and $\mathsf{AG} = \eta - \kappa$. For example, when $(\Delta, \tau) = (1, 0.632)$ and $b = 1$, $(\kappa, \mathsf{AG}) = (0.632, 0.060)$ with $\eta = \kappa + \mathsf{AG} = 0.692$. Kapralov et al. (2013) considered the setting of $(\Delta, \tau) = (\infty, 1), b = \infty$ only and showed that Greedy (GRY) achieves an optimal CR of $1 - 1/\mathrm{e} \sim 0.632$ for general monotone submodular valuation. The same optimal CR is recovered by SM-A and SM-B for coverage valuation; see the result in the lower-right corner marked in blue.

b \ τ	$0.632(\Delta = 1)$	$0.950(\Delta = 3)$	$1(\Delta = \infty)$
1	$(0.632, 0.060)$	$(0.632, 0.060)$	$(0.632, 0.060)$
3	$(0.647, 0.030)$	(0.589,0.025)	(0.580,0.024)
5	$(0.663, 0.024)$	(0.595,0.018)	(0.585,0.017)
∞	$(0.741, 0)$	(0.646,0)	(0.632,0)

Remarks on Results in Table 3. (1) For each row with a given b value, we see that (i) both $\kappa(\tau, b)$ and $\eta(\tau, b)$ are non-increasing over $\tau \in [0, 1]$, which is consistent with Claim (1) of Theorem 3, and (ii) all $\mathsf{AG} = \eta - \kappa$ take non-negative values, which is in line with Claim (2) of Theorem 3. (2) For each column with a given $\tau \in [0, 1]$, AG keeps decreasing from $\mathsf{AG} = 0.0603$ at $b = 1$ to 0 at $b = \infty$. This suggests that the advantage brought by real-time boostings to SM-B diminishes when b increases. (3) The setting of $(\Delta, \tau) = (\infty, 1)$ and $b = \infty$ is exactly what considered by (Kapralov et al., 2013), where they have shown GRY obtains an *optimal* CR of $1 - 1/\mathrm{e}$. The same result is recovered by both SM-A and SM-B (see the result marked in blue). Our results suggest that the optimality of the golden barrier of $1 - 1/\mathrm{e}$ applies only when $(\Delta, \tau) = (\infty, 1)$ and $b = \infty$. In other words, we can well overcome the barrier when either $\Delta < \infty$ (a finite value with $\tau = 1 - \mathrm{e}^{-\Delta} < 1$) or $b < \infty$; see, *e.g.*, SM-B attains a CR of $\eta(\tau, b = \infty) = (1 - \mathrm{e}^{-\tau})/\tau > 1 - 1/\mathrm{e}$ when $\tau = 1 - \mathrm{e}^{-\Delta} < 1$ with $\Delta < \infty$. (4) As noted before, both parameters b and Δ generally take small finite values in real-world settings since each task is allocated a small budget that can afford to recruit a limited number of workers, while each given skill is typically covered by a small number of workers in a diverse workforce. This suggests the superiority of SM-A and SM-B over GRY in practice, where the former two can well beat the golden barrier of $1 - 1/\mathrm{e}$, while the latter has no theoretical guarantee in CR. The example in Figure 1 in Appendix A with $b = 1$ and $(\Delta, \tau) = (1, 1 - 1/\mathrm{e})$ highlights the gap: GRY is zero-competitive, while SM-A and SM-B achieve CRs of at least 0.632 and $0.632 + 0.060 = 0.692$, respectively, as shown in the upper-left corner of Table 3.

Other Related Work. We acknowledge that the idea of real-time boosting (RTB) has already been proposed and implemented as heuristics for various real-world online-matching markets such as ride-hailing and crowdsourcing markets (Dickerson et al., 2019a; 2024). However, very few of them have ever analyzed the performance theoretically. Recently, the idea of RTB has been formally investigated in the context of fairness maximization (Ma et al., 2023) in online matching with all the offline and online agents having a unit matching capacity. The analysis there is very different from here since they focus on the framework of online matching with linear objectives only (instead of coverage functions) and there are no sequence-guided attenuations as proposed here.

Weighted coverage studied here is one classical representative of a more general class of functions, called *monotone submodular*. As for the online setting (where at least part of agents arrive dynamically), Dickerson et al. (2019b) considered a variant of the online matching model under KIID, whose objective is to maximize a single monotone submodular function over the set of all matched edges. They gave a 0.399-competitive randomized matching policy. There are several works of online submodular maximization on other arrival settings; see, *e.g.,* (Esfandiari et al., 2016) (maximization of two monotone submodular functions under adversarial), (Korula et al., 2018) (random order), and (Chan et al., 2017; Rawitz & Rosén, 2021; Ausiello et al., 2012) (adversarial under preemption). As for the offline setting, submodular maximization and its variants have been extensively studied; see a few recent examples (Breuer et al., 2020; Fahrbach et al., 2019; Badanidiyuru et al., 2020; Karimi et al., 2017; Wei et al., 2014; Hassani et al., 2017).

2. A Non-Adaptive Sampling Policy: SM-A

For self-completeness, we state the non-adaptive policy (SM-A) formally in Algorithm 1. As noted before, SM-A appears as a basic module in several other works before.

3. An Adaptive Sampling Policy: SM-B

The adaptive policy SM-B, as formally described in Algorithm 2, takes two auxiliary sequences as input parameters, which are defined in Equation (11). Let $\mathsf{Ber}(\cdot)$ denote a Bernoulli random variable.

$$\phi_1 = \psi_1 = 1, \tag{11}$$

$$\phi_t = \Pr\Big[\sum_{\ell=1}^{t-1} \mathsf{Ber}\Big(\frac{b \cdot \psi_\ell}{T}\Big) \leq b - 1\Big], \quad \forall 2 \leq t \leq T;$$

Algorithm 1 A non-adaptive sampling policy (SM-A)

Offline Phase:
Solve LP (1) for an optimal solution $\{x^*_e | e \in E\}$.
Online Phase:
for $t = 1, 2, \ldots, T$ **do**
  Let an online worker (of type) j arrive at time t.
  Sample a task $i \in \mathcal{N}_j$ following a static distribution $\mathcal{D}_j := \{x^*_{ij} | i \in \mathcal{N}_j\}$, which is valid since $\sum_{i \in \mathcal{N}_j} x^*_{ij} \le 1$ due to Constraint (5) of LP (1).
  If i is safe at t (*i.e.,* i's capacity remains), then assign j to i; otherwise, reject j.
end for

$$\psi_t = \frac{1}{1 - 1/\mathsf{e} + \phi_t/\mathsf{e}}, \qquad \forall 2 \le t \le T.$$

In Equation (11), the term $\sum_{\ell=1}^{t-1} \mathsf{Ber}\big(\frac{b \cdot \psi_\ell}{T}\big)$ represents the sum of $t-1$ independent Bernoulli random variables, each with a mean of $b \cdot \frac{\psi_\ell}{T}$, for $1 \le \ell \le t-1$. Using the above definition, we can sequentially compute the values of $\{\phi_t, \psi_t | t \in [T]\}$ for any given integer $b \ge 1$. In Section 3.1, we propose an Ordinary Differential Equation (ODE)-based approach for the computation of $\{\phi_t, \psi_t | t \in [T]\}$.

Lemma 2. SM-B *parameterized with* $\{\phi_t, \psi_t\}$ *is valid.*

Proof. Recall that $\{\phi_t, \psi_t\}$ are used to guide the two types of attenuations (ATT-I and ATT-II) such that (1) each task is safe at t with probability equal to ϕ_t after ATT-I and (2) each safe task i gets matched by a neighbor j with probability equal to $\psi_t \cdot x^*_{ij}/T$ after real-time boosting (RTB) and ATT-II. For each task i, let α_{it} be the probability that task i is safe at t before ATT-I; and for each edge $e = (i, j) \in E$, let $\beta_{e,t} \cdot x^*_e$ be the probability that i is sampled in RTB before ATT-II, which excludes that j arrives at t and i is safe then. We show $\alpha_{it} \ge \phi_t$ and $\beta_{e,t} \ge \psi_t$ for every $i \in I, e \in E$ and $t \in [T]$ over the induction on $t \in \{1, 2, \ldots, T\}$.

Consider the base case $t = 1$. We see that $\alpha_{it} = 1 = \phi_t$ for each $i \in I$. Focus on a given edge $e = (i, j)$. Observe that $\mathcal{N}_{jt} = \mathcal{N}_j$ when $t = 1$ since all neighbors of i are safe then. Thus, e gets sampled with probability equal to

$$\beta_{e,t} \cdot x^*_e := \frac{x^*_e}{\sum_{\bar{i} \in \mathcal{N}_{jt}} x^*_{\bar{i}j}} = \frac{x^*_e}{\sum_{\bar{i} \in \mathcal{N}_j} x^*_{\bar{i}j}} \ge x^*_e,$$

where the last inequality follows from Constraint (5) of LP (1), *i.e.,* $\sum_{\bar{i} \in \mathcal{N}_j} x^*_{\bar{i}j} \le 1$. Thus, $\beta_{e,t} \ge 1 = \psi_t$ at $t = 1$.

Now assume $\alpha_{i\ell} \ge \phi_\ell$ and $\beta_{e,\ell} \ge \psi_\ell$ for every $i \in I, e \in E$ and $\ell \in [t]$ and we show the case $\ell = t + 1$. Our induction assumption means that SM-B can function well at least by (the beginning of time) $t + 1$. This allows us to assume that for every $i \in I, e \in E$ and $\ell \in [t]$, (1) each task i is safe at ℓ equal to ϕ_ℓ after ATT-I, denoted by $\mathsf{E}[\mathsf{SF}_{it}] = \phi_\ell$; and (2) each edge $e = (i, j)$ gets matched during ℓ with probability equal to $\mathsf{E}[M_{e,\ell}] = \psi_\ell \cdot x^*_e/T$ after RTB and ATT-II, assuming i is safe at ℓ. Consider a given task $i \in I$. From our analysis, assuming i is safe at ℓ, i gets matched once during $\ell \in [t]$ iff one neighbor $j \in \mathcal{N}_i$ arrives and matched i, which occurs with probability equal to $\sum_{j \in \mathcal{N}_i} \psi_\ell \cdot x^*_{ij}/T \le \psi_\ell \cdot b/T$ due to Constraint (4) of LP (1). Note that by definition, i stays safe at $t + 1$ (before ATT-I) if the total number of matching times before $t + 1$ is no more than $b - 1$. Thus,

$$\alpha_{i,t+1} \ge \Pr\Big[\sum_{\ell=1}^{t} \mathsf{Ber}\Big(\frac{b \cdot \psi_\ell}{T}\Big) \le b - 1\Big] = \phi_{t+1}.$$

Now assume that after ATT-I, every task $i \in I$ stays safe with probability equal to ϕ_{t+1}, denoted by $\mathsf{E}[\mathsf{SF}_{i,t+1}] = \phi_{t+1}$. Consider a given edge $e = (i, j)$. Observe that i gets sampled in RTB when j arrives at $t + 1$ with probability

$$\beta_{e,t+1} \cdot x^*_e := \mathsf{E}\left[\frac{x^*_e}{\sum_{\bar{i} \in \mathcal{N}_{j,t+1}} x^*_{\bar{i}j}} \;\Big|\; \mathsf{SF}_{i,t+1} = 1\right]$$

$$\ge \frac{x^*_e}{x^*_e + \sum_{\bar{i} \in \mathcal{N}_j, \bar{i} \ne i} x^*_{\bar{i}j} \cdot \phi_{t+1}} \tag{12}$$

$$\ge \frac{x^*_e}{x^*_e + (1 - x^*_e) \cdot \phi_{t+1}} \tag{13}$$

$$\ge \frac{x^*_e}{(1 - 1/\mathsf{e}) + \phi_{t+1}/\mathsf{e}} = x^*_e \cdot \psi_{t+1}, \tag{14}$$

where Inequality (12) is due to Jensen's inequality; Inequality (13) follows from Constraint (5) of LP (1) and Inequality (14) from Constraint (6) of LP (1) (*i.e.,* $x^*_e \le 1 - 1/\mathsf{e}$) and the fact $\phi_{t+1} \in [0, 1]$. Thus, we claim $\beta_{e,t+1} \ge \psi_{t+1}$ and complete the induction. ☐

Implementations of ATT-I **and** ATT-II. Recall that α_{it} is the probability that task i is safe at t before ATT-I, and $\beta_{e,t} \cdot x^*_e$ is the probability that i is sampled in RTB when j arrives at t. From Lemma 2, we see $\alpha_{it} \ge \phi_t$ and $\beta_{e,t} \ge \psi_t$ for every $i \in I, e \in E$ and $t \in [T]$. By Monte-Carlo simulating SM-B recursively to time t, we can get a sharp estimate of all values $\{\alpha_{it}, \beta_{e,t} | i \in I, e \in E\}$ for each given $t \in [T]$.[7] For ATT-I, by independently relabeling each safe task i as "safe" and "unsafe" with probability ϕ_t/α_{it} and $1 - \phi_t/\alpha_{it}$, respectively, we achieve the goal that each task i survives to be safe with probability equal to ϕ_t. Similarly, for ATT-II, we can match edge $e = (i, j)$ with probability $\psi_t/\beta_{e,t}$ (and reject i otherwise) after i gets sampled in RTB. In that way, we achieve our goal that each e is sampled and matched during t with probability equal to $\psi_t \cdot x^*_e$.

[7]We can obtain an estimate with multiplicative error of ϵ under confidence of $(1 - \delta)$ by choosing a sample size of $\Theta((1/\epsilon^2) \cdot \ln(1/\delta))$. By taking $\epsilon = 1/\mathrm{poly}(N)$, where N is the input size, we can ensure ϵ brings just lower-order terms in the final competitive ratios.

Algorithm 2 An adaptive policy SM-B parameterized with auxiliary sequences $\{\phi_t, \psi_t\}$, as defined in Equation (11).

Offline Phase:
Solve LP (1) for an optimal solution $\{x_e^* | e \in E\}$.
Online Phase:
for $t = 1, 2, \ldots, T$ **do**
  The first type of attenuations (ATT-I): Apply simulation-based attenuations such that each task is *safe* at (the beginning of) t with probability *equal* to ϕ_t. ▷ A task being safe means its capacity remains then.◁
  Let an online worker (of type) j arrive at t, and $\mathcal{N}_{j,t}$ be the set of *safe* neighbors at t.
  Adaptive Boosting (RTB): Sampling a safe neighbor i following a boosted distribution $\widetilde{\mathcal{D}}_{j,t} = \{x_{ij}^* / \sum_{\bar{i} \in \mathcal{N}_{jt}} x_{\bar{i}j}^* | i \in \mathcal{N}_{jt}\}$.
  The second type of attenuations (ATT-II): Match i and j with a certain attenuation factor such that each safe neighbor $i \in \mathcal{N}_{j,t}$ gets sampled and matched with j during t with probability *equal* to $\psi_t \cdot x_{ij}^*$.
  ▷ *Details of* ATT-I *and* ATT-II *are offered in the paragraph titled "**Implementations of** ATT-I **and** ATT-II." The probability of ($\psi_t \cdot x_{ij}^*$) in* ATT-II *captures the event that i gets sampled and matched with j during time t, which conditions on i is safe and j arrives at time t.*◁
end for

3.1. An Ordinary Differential Equation (ODE)-Based Approach to Computing $\{\phi_t, \psi_t\}$ when $T \to \infty$

For each $t \in [T]$, let $\Phi(t/T) := \phi_t$ and $\Psi(t/T) := \psi_t$, where $\Phi(z)$ and $\Psi(z)$ are the two continuous functions over $z \in [0, 1]$ with $\Phi(0) = \Psi(0) = 1$ and $\Psi = 1/(1 - 1/\mathsf{e} + \Phi/\mathsf{e})$. Let $h(z) = \int_0^z d\zeta \cdot \Psi(\zeta)$ for $z \in [0, 1]$. In the following, we aim to derive analytical forms of Φ and Ψ when $T \to \infty$. Consider a given integer $b \geq 1$. Let $X_t := \sum_{\ell=1}^{t-1} \mathsf{Ber}\big(\frac{b\psi_\ell}{T}\big)$. By definition of $\{\phi_t, \psi_t\}$ in (11),

$$\phi_{t+1} = \Pr[X_t \leq b-2] + \Pr[X_t = b-1](1 - b \cdot \psi_t / T),$$
$$\phi_t = \Pr[X_t \leq b-2] + \Pr[X_t = b-1].$$

Thus,

$$\begin{aligned}
&\phi_{t+1} - \phi_t = (-b\psi_t/T) \cdot \Pr[X_t = b-1] \\
\Leftrightarrow &(\phi_{t+1} - \phi_t) \cdot T = (-b\psi_t) \Pr[X_t = b-1] \\
\Leftrightarrow &(\phi_{t+1} - \phi_t) \cdot T = (-b\psi_t) \Pr\Big[\sum_{\ell<t} \mathsf{Ber}\big(b\psi_\ell/T\big) = b-1\Big] \\
\Leftrightarrow &(\phi_{t+1} - \phi_t) \cdot T = \qquad (15)\\
&\quad (-b\psi_t)\Big(\Pr\Big[\mathsf{Pois}\Big(\sum_{\ell<t} b\psi_\ell/T\Big) = b-1\Big] + O(b/T)\Big) \\
\Leftrightarrow &\Phi'(z) = (-b\Psi(z)) \cdot \Pr\Big[\mathsf{Pois}\Big(b \cdot h(z)\Big) = b-1\Big] \\
&\quad \Big(\text{where } z = t/T, h(z) = \textstyle\int_0^z d\zeta \cdot \Psi(\zeta), T \to \infty\Big) \\
\Leftrightarrow &(-\mathsf{e}) \cdot \Psi^{-2} \cdot \Psi' = (-b\Psi) \cdot \Pr\Big[\mathsf{Pois}\big(b \cdot h\big) = b-1\Big] \\
&\quad \big(\text{since } \Psi = 1/(1 - 1/\mathsf{e} + \Phi/\mathsf{e})\big) \\
\Leftrightarrow &h'' = (h')^3 \cdot h^{b-1} \cdot \mathsf{e}^{-b \cdot h - 1} b^b / (b-1)!.
\end{aligned}$$

In Equality (15), we use the Poisson approximation of a series of Bernoulli trials due to (Serfling, 1978). Observe that $h(0) = 0$ and $h'(0) = 1$ since $\Psi(0) = 1$. Thus, for each given integer $b \geq 1$, we can solve $h(z)$ from the differential equation as shown in (9), which is re-stated as below,

$$h'' = (h')^3 \cdot h^{b-1} \cdot \mathsf{e}^{-b \cdot h - 1} \cdot b^b / (b-1)!, \;\; h(0) = 0, h'(0) = 1.$$

Since $h(z) = \int_0^z d\zeta \cdot \Psi(\zeta)$, we can get $\Psi(z) = h'(z)$ and Φ through Equality $\Psi = 1/(1 - 1/\mathsf{e} + \Phi/\mathsf{e})$. Below is an example of $h(z)$ when $b = 2$.

$$h(z) = \mathsf{IV}\Big[\frac{(\mathsf{e}-1) \cdot \tilde{z}}{\mathsf{e}} - \frac{1 + \tilde{z}}{\mathsf{e}^{2\tilde{z}+1}}\Big](z - 1/\mathsf{e}), \qquad (16)$$

where $\mathsf{IV}[f]$ means the inverse function of f. Thus, the above expression means that when $b = 2$, $h(z) = \tilde{z}$ with

$$f(\tilde{z}) := \frac{(\mathsf{e}-1) \cdot \tilde{z}}{\mathsf{e}} - \frac{1 + \tilde{z}}{\mathsf{e}^{2\tilde{z}+1}} = z - 1/\mathsf{e}.$$

See more examples of h on other b values in Appendix E.

4. Conclusions and Future Work

In this paper, we proposed a model of online coverage maximization (**OCCM**), which is featured by that each offline agent is associated with a coverage valuation. We carefully analyzed two LP-based sampling policies; one is non-adaptive, while the other is adaptive armed with boostings. For each policy, we characterized the final CR as a function of two parameters, a uniform matching capacity among tasks and an upper bound on the number of online agents covering any feature for any given offline agent.

Our work opens a few new directions. The first is generalizing the current techniques to the case when each offline agent is associated with a generic monotone submodular function. Since coverage valuation is perhaps the most fundamental representation of a monotone submodular, we believe our techniques could offer useful insights into that setting. The second goal is to identify some real datasets and numerically evaluate the performance of the two policies against Greedy among others.

Acknowledgements

Pan Xu was partially supported by NSF CRII Award IIS-1948157. The author would like to thank the anonymous reviewers for their valuable comments.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none of which we feel must be specifically highlighted here.

References

Ahmed, F., Dickerson, J., and Fuge, M. Forming diverse teams from sequentially arriving people. *Journal of Mechanical Design*, 142(11):111401, 2020.

Anagnostopoulos, A., Becchetti, L., Castillo, C., Gionis, A., and Leonardi, S. Online team formation in social networks. In *Proceedings of the 21st international conference on World Wide Web*, pp. 839–848, 2012.

Ashlagi, I., Burq, M., Dutta, C., Jaillet, P., Sholley, C., and Saberi, A. Edge weighted online windowed matching. In *Proceedings of the Nineteenth ACM Conference on Economics and Computation*, 2019.

Ausiello, G., Boria, N., Giannakos, A., Lucarelli, G., and Paschos, V. T. Online maximum k-coverage. *Discrete Applied Mathematics*, 160(13-14):1901–1913, 2012.

Badanidiyuru, A., Karbasi, A., Kazemi, E., and Vondrák, J. Submodular maximization through barrier functions. *Advances in Neural Information Processing Systems*, 33: 524–534, 2020.

Barnabò, G., Fazzone, A., Leonardi, S., and Schwiegelshohn, C. Algorithms for fair team formation in online labour marketplaces. In *Companion Proceedings of The 2019 World Wide Web Conference*, pp. 484–490, 2019.

Bei, X. and Zhang, S. Algorithms for trip-vehicle assignment in ride-sharing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 3–9, 2018.

Bradac, D., Singla, S., and Zuzic, G. (near) optimal adaptivity gaps for stochastic multi-value probing. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2019)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2019.

Breuer, A., Balkanski, E., and Singer, Y. The fast algorithm for submodular maximization. In *International Conference on Machine Learning*, pp. 1134–1143. PMLR, 2020.

Brubach, B., Sankararaman, K. A., Srinivasan, A., and Xu, P. New algorithms, better bounds, and a novel model for online stochastic matching. In *24th Annual European Symposium on Algorithms (ESA 2016)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2016.

Brubach, B., Sankararaman, K. A., Srinivasan, A., and Xu, P. Online stochastic matching: New algorithms and bounds. *Algorithmica*, pp. 1–47, 2020.

Canonne, C. A short note on poisson tail bounds. `http://www.cs.columbia.edu/~ccanonne/files/misc/2017-poissonconcentration.pdf`, 2020. Accessed: 2020-02-01.

Chan, T. H., Huang, Z., Jiang, S. H.-C., Kang, N., and Tang, Z. G. Online submodular maximization with free disposal: Randomization beats for partition matroids. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 1204–1223. SIAM, 2017.

Cheng, P., Lian, X., Chen, L., Han, J., and Zhao, J. Task assignment on multi-skill oriented spatial crowdsourcing. *IEEE Transactions on Knowledge and Data Engineering*, 28(8):2201–2215, 2016.

Dean, B. C., Goemans, M. X., and Vondrák, J. Approximating the stochastic knapsack problem: The benefit of adaptivity. *Mathematics of Operations Research*, 33(4): 945–964, 2008.

Dickerson, J. P., Sankararaman, K. A., Sarpatwar, K. K., Srinivasan, A., Wu, K.-L., and Xu, P. Online resource allocation with matching constraints. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*, pp. 1681–1689. International Foundation for Autonomous Agents and Multiagent Systems, 2019a.

Dickerson, J. P., Sankararaman, K. A., Srinivasan, A., and Xu, P. Balancing relevance and diversity in online bipartite matching via submodularity. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 1877–1884, 2019b.

Dickerson, J. P., Sankararaman, K. A., Srinivasan, A., and Xu, P. Allocation problems in ride-sharing platforms: Online matching with offline reusable resources. *ACM Transactions on Economics and Computation (TEAC)*, 9 (3):1–17, 2021.

Dickerson, J. P., Sankararaman, K. A., Srinivasan, A., Xu, P., and Xu, Y. Matching tasks and workers under known arrival distributions: Online task assignment with two-sided arrivals. *ACM Transactions on Economics and Computation*, 2024.

Esfandiari, H., Korula, N., and Mirrokni, V. Bi-objective online matching and submodular allocations. In *Advances in Neural Information Processing Systems*, pp. 2739–2747, 2016.

Esmaeili, S., Duppala, S., Cheng, D., Nanda, V., Srinivasan, A., and Dickerson, J. P. Rawlsian fairness in online bipartite matching: Two-sided, group, and individual. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 5624–5632, 2023.

Fahrbach, M., Mirrokni, V., and Zadimoghaddam, M. Non-monotone submodular maximization with nearly optimal adaptivity and query complexity. In *International Conference on Machine Learning*, pp. 1833–1842. PMLR, 2019.

Fata, E., Ma, W., and Simchi-Levi, D. Multi-stage and multi-customer assortment optimization with inventory constraints. *Available at SSRN 3443109*, 2019.

Feldman, J., Mehta, A., Mirrokni, V., and Muthukrishnan, S. Online stochastic matching: Beating 1-1/e. In *Foundations of Computer Science, 2009. FOCS'09. 50th Annual IEEE Symposium on*, pp. 117–126. IEEE, 2009.

Gandhi, R., Khuller, S., Parthasarathy, S., and Srinivasan, A. Dependent rounding and its applications to approximation algorithms. *Journal of the ACM (JACM)*, 53(3):324–360, 2006.

Haeupler, B., Mirrokni, V. S., and Zadimoghaddam, M. Online stochastic weighted matching: Improved approximation algorithms. In *Internet and Network Economics*, volume 7090 of *Lecture Notes in Computer Science*, pp. 170–181. Springer Berlin Heidelberg, 2011. ISBN 978-3-642-25509-0.

Hassani, H., Soltanolkotabi, M., and Karbasi, A. Gradient methods for submodular maximization. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pp. 5843–5853, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

Huang, Z. and Shu, X. Online stochastic matching, poisson arrivals, and the natural linear program. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pp. 682–693, 2021.

Jaillet, P. and Lu, X. Online stochastic matching: New algorithms with better bounds. *Mathematics of Operations Research*, 39(3):624–646, 2013.

Kapralov, M., Post, I., and Vondrák, J. Online submodular welfare maximization: Greedy is optimal. In *Proceedings of the twenty-fourth annual ACM-SIAM symposium on Discrete algorithms*, pp. 1216–1225. SIAM, 2013.

Karimi, M., Lucic, M., Hassani, H., and Krause, A. Stochastic submodular maximization: The case of coverage functions. In *Advances in Neural Information Processing Systems*, pp. 6853–6863, 2017.

Korula, N., Mirrokni, V., and Zadimoghaddam, M. Online submodular welfare maximization: Greedy beats 1/2 in random order. *SIAM Journal on Computing*, 47(3):1056–1086, 2018.

Li, Y., Fu, K., Wang, Z., Shahabi, C., Ye, J., and Liu, Y. Multi-task representation learning for travel time estimation. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 1695–1704, 2018.

Ma, W., Xu, P., and Xu, Y. Fairness maximization among offline agents in online-matching markets. *ACM Transactions on Economics and Computation*, 10(4):1–27, 2023.

Manshadi, V. H., Gharan, S. O., and Saberi, A. Online stochastic matching: Online actions based on offline statistics. *Mathematics of Operations Research*, 37(4): 559–573, 2012.

Mehta, A. Online matching and ad allocation. *Foundations and Trends in Theoretical Computer Science*, 8(4):265–368, 2013.

Rawitz, D. and Rosén, A. Online budgeted maximum coverage. *Algorithmica*, 83:2989–3014, 2021.

Serfling, R. J. Some elementary results on poisson approximation in a sequence of bernoulli trials. *SIAM Review*, 20(3):567–579, 1978.

Slud, E. V. Distribution inequalities for the binomial law. *The Annals of Probability*, pp. 404–412, 1977.

Wang, Z., Fu, K., and Ye, J. Learning to estimate the travel time. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 858–866, 2018.

Wei, K., Iyer, R., and Bilmes, J. Fast multi-stage submodular maximization. In Xing, E. P. and Jebara, T. (eds.), *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pp. 1494–1502, Bejing, China, 22–24 Jun 2014. PMLR.

Xu, P. Exploring the tradeoff between competitive ratio and variance in online-matching markets. In *The 18th Conference on Web and Internet Economics*, 2023a.

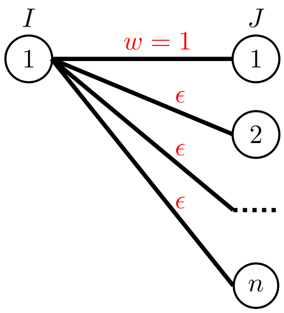
Xu, P. Two-sided capacitated submodular maximization in gig platforms. In *International Conference on Web and Internet Economics*, pp. 600–617. Springer, 2023b.

Yao, H., Wu, F., Ke, J., Tang, X., Jia, Y., Lu, S., Gong, P., Ye, J., and Li, Z. Deep multi-view spatial-temporal network for taxi demand prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, AAAI'18, pp. 2588–2595, 2018.

Zhao, B., Xu, P., Shi, Y., Tong, Y., Zhou, Z., and Zeng, Y. Preference-aware task assignment in on-demand taxi dispatching: An online stable matching approach. In *Proceedings of the AAAI Conference on Artificial Intelligence*, AAAI'19, 2019.

A. Illustrating the Differences Between GRY, SM-A, and SM-B: An Example due to (Xu, 2023b)

Xu (2023b) presented an example (see Figure 1) to show that Greedy (GRY) achieves a competitive ratio of zero for **OCCM**. We use the same example to highlight the differences between GRY and the two policies SM-A (Algorithm 1) and SM-B (Algorithm 2), as studied here.



$$I = \{1\}, J = [n];$$
$$T = |J| = K = n;$$
$$b_i = b_j = 1, \forall i, j;$$
$$w_{i=1,k=1} = 1;$$
$$w_{1,k} = \epsilon, 2 \leq k \leq n;$$
$$\mathsf{GRY} = \epsilon \cdot (1 - 1/n) + 1/n;$$
$$\mathrm{OPT} = 1 - 1/\mathsf{e} + \epsilon/\mathsf{e}.$$

$$y^*_{i=1,k=1} = 1 - 1/\mathsf{e}, y^*_{12} = 1/\mathsf{e};$$
$$y^*_{1,k} = 0, \forall 3 \leq k \leq n;$$
$$x^*_{ij} = y^*_{i,k=j}, \forall i, j;$$
$$\mathsf{SM\text{-}A} = (1 - 1/\mathsf{e})^2 + \frac{\epsilon(1 - 1/\mathsf{e})}{\mathsf{e}}.$$
$$\mathsf{SM\text{-}A}/\,\mathrm{OPT} \sim 0.6321.$$
$$\mathsf{SM\text{-}B}/\,\mathrm{OPT} \geq 0.6924.$$

Figure 1. An example highlighting the difference between Greedy (GRY) and the two policies SM-A (Algorithm 1) and SM-A (Algorithm 2) proposed here. There are $n = |J|$ worker types and $K = n$ skills such that each worker (of type) j possesses one single skill $k = j \in [n]$, and there is one single task $i = 1$ with a unit matching capacity such that $w_{ik} = 1$ for $k = 1$ and $w_{ik} = \epsilon > 0$ for $2 \leq k \leq n$. We can verify that (1) GRY yields an expected utility of no more than $\epsilon + 1/n$ since it matches whatever arrives at $t = 1$ and then stops. In contrast, an offline optimal (OPT) achieves a total of $1 - 1/\mathsf{e} + \epsilon/\mathsf{e}$ in expectation, which matches $i = 1$ with $j = 1$ if it arrives at least once (with probability $1 - 1/\mathsf{e}$) and with any other arriving $j \neq 1$ otherwise. Thus, we claim that GRY achieves a CR no more than $(\epsilon + 1/n)/(1 - 1/\mathsf{e})$, which approaches zero when $n \to \infty, \epsilon \to 0$. (2) An optimal solution to benchmark LP (1) can be as follows: $x^*_{i=1,j=1} = y^*_{i=1,k=1} = 1 - 1/\mathsf{e}$, $x^*_{i=1,j=2} = y^*_{i=1,k=2} = 1/\mathsf{e}$, and $x^*_{i=1,j} = y^*_{i=1,k} = 0$ for all $3 \leq j, k \leq n$ with an optimal value $\mathsf{Val}(\mathrm{LP}\text{-}(1)) = 1 - 1/\mathsf{e} + \epsilon/\mathsf{e}$. (3) SM-A will match $j = 1$ and $j = 2$ with respective probabilities $1 - 1/\mathsf{e}$ and $1/\mathsf{e}$ non-adaptively when each arrives (with probability $1/n$) during each time, yielding an expected utility of $(1 - 1/\mathsf{e})(1 - 1/\mathsf{e} + \epsilon/\mathsf{e})$. This indicates that SM-A achieves a CR of $1 - 1/\mathsf{e}$ on the instance. (4) SM-B will match $j = 1$ and $j = 2$ with respective probabilities $(1 - 1/\mathsf{e}) \cdot \psi_t$ and $(1/\mathsf{e}) \cdot \psi_t$ when each arrives (with probability $1/n$) at time $t \in [T]$ given i is not matched then, where $\{\psi_t | t \in [T]\}$ is an increasing sequence with $\psi_1 = 1$ and $\psi_T \geq 1/\big(1 - 1/\mathsf{e} + (1 - 1/\mathsf{e})/\mathsf{e}\big)$ as specified in (11). This yields an improved CR of at least $0.6321 + 0.0603 = 0.6924$ for SM-B.

B. Proof of Theorem 2

B.1. An Auxiliary Balls-and-Bins Model for Competitive Analysis

Consider a given task-skill pair $\lambda = (i, k) \in \Lambda$ and recall that $\mathcal{N}_\lambda$ is the set of neighbors of i covering skill k. *Thus, skill k is covered for i iff one neighbor $j \in \mathcal{N}_\lambda$ is matched before the capacity b of task i gets exhausted.* Note that during each round t, SM-B matches a neighbor $j \in \mathcal{N}_\lambda$ with probability $\sum_{j \in \mathcal{N}_\lambda} \psi_t \cdot x^*_{ij}/T = \psi_t \cdot x^*_\lambda/T := \psi_t \cdot p/T$ (after RTB and ATT-II), where $p = x^*_\lambda$. Meanwhile, SM-B matches a neighbor $j \in (\mathcal{N}_i - \mathcal{N}_\lambda)$ with probability $\sum_{j \in (\mathcal{N}_i - \mathcal{N}_\lambda)} \psi_t \cdot x^*_{ij}/T := \psi_t \cdot q/T$ with $q = \sum_{j \in (\mathcal{N}_i - \mathcal{N}_\lambda)} x^*_{ij}$. By the nature of SM-B, it keeps on sampling neighbors from $\mathcal{N}_\lambda$ and $(\mathcal{N}_i - \mathcal{N}_\lambda)$ with respective probabilities $\psi_t \cdot p/T$ and $\psi_t \cdot q/T$ during each round until either b neighbors are matched or we reach the last round T. Observe that (1) $p + q \leq b$ from Constraint (4) and (2) $p = x^*_\lambda \geq y^*_\lambda$ by Constraint (3) of LP-(1). For $\lambda = (i, k)$, let $Y_\lambda = 1$ indicate at least one neighbor from $\mathcal{N}_\lambda$ gets matched in the end, which suggests skill k is covered for task i. For fixed values of b and y^*_λ, we see $\mathsf{E}[Y_\lambda]/y^*_\lambda$ gets minimized when $p = y^*_\lambda$. Thus, assume WLOG that $p + q = b$ [8] and $p = y^*_\lambda \leq 1 - \mathsf{e}^{-\Delta} = \tau$, where the last inequality is due to Constraint (2).

Let us treat task i as a bin with a capacity b and neighbors from $\mathcal{N}_\lambda$ and $\mathcal{N}_i - \mathcal{N}_\lambda$ as two types of balls. Then we can rephrase the above sampling process of SM-B alternatively as a Balls-and-Bins model as follows.

A Balls-and-Bins Model (BBM). Suppose we have one single bin and two types of balls, namely type A and type B. We have T rounds and during each round $t \in [T]$, at most one single ball will get sampled (with replacement) such that it is from type A with probability $\psi_t \cdot p/T$ and from type B with $\psi_t \cdot q/T$, and with probability $1 - (p + q)/T$, no ball gets sampled. Here we assume $T \gg b \geq 1$, $0 \leq p, q \leq b$, $p + q = b$, and $p \leq \tau$. Each bin has a capacity of b such that the sampling process will stop either the bin has b balls (copies of each type will be counted) or we reach the last round $t = T$. Let $Y = 1$

[8] We can create virtual neighbors of i to make $p + q = b$, which have the same impact as ATT-I.

indicate that at least one ball of type A gets sampled by the termination of **BBM**. We aim to prove that $\mathsf{E}[Y]/p \geq \eta(\tau, b)$, where η is as defined in (8).

B.2. A Key Lemma and Its Proof

For each task-skill pair $\lambda = (i, k)$, let $Y_\lambda = 1$ indicate that skill k is covered for task i in SM-B.

Lemma 3. $\mathsf{E}[Y_\lambda]/y^*_\lambda \geq \eta(\tau, b)$ *for every* $\lambda \in \Lambda$*, where* $\eta(\tau, b)$ *is as defined in* (8) *and* $\{y^*_\lambda\}$ *is part of an optimal solution to* LP (1).

Proof. Let $Y(t) = 1$ indicate that one ball of type A gets sampled for the first time during $t \in [T]$. Thus, $Y = \sum_{t=1}^{T} Y(t)$. Observe that $Y(t) = 1$ iff (1) no ball of type A has ever been sampled before t, which occurs with probability $\prod_{\ell<t}(1 - \psi_t \cdot p/T)$; (2) one ball of type A gets sampled at t with probability $\psi_t \cdot p/T$; and (3) the bin has a capacity no more than $b-1$ at (the beginning of) t, which happens with probability $\Pr\left[\sum_{\ell=1}^{t-1} \mathsf{Ber}\left(\frac{(b-\tau)\cdot\psi_\ell}{T-\psi_\ell}\right) \leq b-1\right]$. The last probability conditions on no ball of type A gets sampled during each time $1 \leq \ell < t$, which suggests a ball of type B gets sampled during ℓ with a probability $(\psi_\ell \cdot q/T)/(1 - \psi_\ell \cdot p/T) = (\psi_\ell \cdot (b-\tau))/(T - \psi_\ell \cdot p)$.

$$
\begin{aligned}
\mathsf{E}[Y]/p &= \sum_{t=1}^{T} \mathsf{E}[Y(t)]/p = \sum_{t=1}^{T} \frac{\psi_t}{T} \prod_{\ell=1}^{t-1} \left\{ \left(1 - \frac{p\cdot\psi_\ell}{T}\right) \cdot \Pr\left[\sum_{\ell=1}^{t-1} \mathsf{Ber}\left(\frac{(b-p)\cdot\psi_\ell}{T - p\cdot\psi_\ell}\right) \leq b-1\right] \right\} \\
&= \sum_{t=1}^{T} \frac{\psi_t}{T} \prod_{\ell<t} \left\{ \left(1 - \frac{p\cdot\psi_\ell}{T}\right) \cdot \left(\Pr\left[\mathsf{Pois}\left(\sum_{\ell<t} \frac{(b-p)\cdot\psi_\ell}{T}\right) \leq b-1\right] - O(b/T) \right) \right\} \\
&= \int_0^1 dz \cdot \left\{ \Psi(z) \cdot \mathsf{e}^{-p\cdot h(z)} \cdot \mathsf{e}^{-(b-p)\cdot h(z)} \cdot \sum_{\ell=0}^{b-1} \frac{(b-p)^\ell \cdot h^\ell(z)}{\ell!} \right\} \\
&\quad \left(\text{where } z = t/T \text{ and } T \to \infty, \text{ and recall that } \Psi(t/T) = \psi_t \text{ and } h(z) = \textstyle\int_0^z d\zeta \cdot \Psi(\zeta)\right) \\
&= \int_0^1 dz \cdot \left\{ h'(z) \cdot \mathsf{e}^{-b\cdot h(z)} \cdot \sum_{\ell=0}^{b-1} \frac{(b-p)^\ell \cdot h^\ell(z)}{\ell!} \right\} \\
&\geq \int_0^1 dz \cdot \left\{ h'(z) \cdot \mathsf{e}^{-b\cdot h(z)} \cdot \sum_{\ell=0}^{b-1} \frac{(b-\tau)^\ell \cdot h^\ell(z)}{\ell!} \right\} \quad \left(\text{since } p \leq 1 - \mathsf{e}^{-\Delta} = \tau\right) \\
&= \int_0^1 dz \cdot \frac{h'(z)}{\mathsf{e}^{\tau\cdot h(z)}} \cdot \Pr\Big[\mathsf{Pois}((b-\tau)\cdot h(z)) \leq b-1\Big] = \eta(\tau, b).
\end{aligned}
$$

□

B.3. Proof of Theorem 2

Proof. By the linearity of expectation, we have

$$
\mathsf{E}[\text{SM-B}] = \sum_{\lambda\in\Lambda} w_\lambda \cdot \mathsf{E}[Y_\lambda] \geq \eta(\tau, b) \sum_{\lambda\in\Lambda} w_\lambda \cdot y^*_\lambda = \eta(\tau, b) \cdot \mathsf{Val}(\text{LP}\,(1)) \geq \eta(\tau, b) \cdot \text{OPT},
$$

where $\mathsf{Val}(\text{LP}\,(1))$ and OPT denote the optimal value of LP (1) and the performance of a clairvoyant optimal, respectively. The last inequality above follows from Lemma 1. Thus, we establish the competitiveness of $\eta(\tau, b)$ for SM-B. □

C. Proof of Theorem 1

Proof. We can cast SM-A as a special case of SM-B with no RTB, ATT-II, or ATT-I as follows. For SM-A, we see that (1) during each time t when a worker j arrives, each neighbor $i \in \mathcal{N}_j$ gets sampled and matched with probability equal to $\tilde{\psi}_t \cdot x^*_{ij}$ with $\tilde{\psi}_t = 1$, conditioning on j arrives and i is safe then; and (2) each i is safe at t with probability equal to $\tilde{\phi}_t = \Pr[\sum_{\ell=1}^{t-1} \mathsf{Ber}(b/T) \leq b-1]$ in the worst scenario when $x^*_i = b$. Thus, we can analyze SM-A in the same way as we did for SM-B but with $\{\tilde{\phi}_t, \tilde{\psi}_t\}$ redefined as below,

$$\begin{aligned}
&\tilde{\phi}_1 = 1, \\
&\tilde{\psi}_t = 1, \forall 1 \le t \le T \\
&\tilde{\phi}_t = \Pr\Big[\sum_{\ell=1}^{t-1} \mathsf{Ber}\Big(\frac{b\cdot\tilde{\psi}_\ell}{T}\Big) \le b-1\Big], \forall 1 \le t \le T.
\end{aligned} \tag{17}$$

Consider a given task-skill pair $\lambda = (i,k)$. Similar to SM-B, assume that we have one single bin with capacity b and two types of balls, which get sampled with probability $p/T = y^*_\lambda/T$ and $(b-p)/T$, respectively. We aim to show that $\mathsf{E}[Y]/p \ge \kappa(\tau, b)$, where $Y = 1$ indicates that a ball of type I gets sampled at least once before either the bin exhausts capacity or it reaches the last round T. Observe that in our case, $\widetilde{\Psi}(z) = 1$, where $\widetilde{\Psi}(t/h) = \tilde{\psi}_t = 1$ as defined in (17). Thus, $\tilde{h}(z) = \int_0^z d\zeta \cdot \widetilde{\Psi}(\zeta) = z$ for $z \in [0,1]$. Following the same analysis we did to $\mathsf{E}[Y]/p$ as shown in SM-B, we see that $\mathsf{E}[Y]/p \ge \eta(\tau, b) = \kappa(\tau, b)$, where $h(z)$ appearing in $\eta(\tau, b)$ is replaced by the updated version $\tilde{h}(z) = z$. ☐

Remarks. We emphasize that dual sequences $\{\tilde{\phi}_t, \tilde{\psi}_t\}$ in Equation (17) serve an essentially different purpose from $\{\phi_t, \psi_t\}$ in Equation (11), though the two look similar. For SM-B, we introduce the dual sequences in (11) to guide the two types of attenuations, which have a direct impact on how SM-B operates. In contrast, we create the dual sequences in (17) here just to facilitate the online analysis of SM-A, which have no effect on the matching policy itself.

D. Proof of Theorem 3

We split the main Theorem 3 into the following three lemmas.

Lemma 4. *For any given $b \ge 1$, both $\kappa(\tau, b)$ and $\eta(\tau, b)$ are non-increasing over $\tau \in [0,1]$.*

Lemma 5. *For any given $b \ge 1$ and $\tau \in [0,1]$, $\eta(\tau, b) \ge 0.602$ and $\eta(\tau, b) \ge \kappa(\tau, b)$.*

Lemma 6. *$\kappa(\tau, b) = (1 - \mathsf{e}^{-\tau})/\tau - \Theta(b^{-1/2})$, where $\Theta(b^{-1/2}) = c \cdot b^{-1/2}(1 + o(1))$ with $c \in [\sqrt{1/(2\pi \mathsf{e}^2)}, \sqrt{2/\pi}]$ being a constant and $o(1)$ vanishing as $b \to \infty$.*

D.1. Proof of Lemma 4

Proof. Note that for any $\tau \in [0,1]$ and $b \ge 1$,

$$\begin{aligned}
\kappa(\tau, b) &= \int_0^1 dz \cdot \mathsf{e}^{-\tau\cdot z} \cdot \Pr\Big[\mathsf{Pois}\big((b-\tau)\cdot z\big) \le b-1\Big] = \int_0^1 dz \cdot \mathsf{e}^{-\tau\cdot z} \cdot \sum_{\ell=0}^{b-1} \mathsf{e}^{-(b-\tau)\cdot z} \frac{(b-\tau)^\ell z^\ell}{\ell!} \\
&= \int_0^1 dz \cdot \sum_{\ell=0}^{b-1} \mathsf{e}^{-b\cdot z} \frac{(b-\tau)^\ell z^\ell}{\ell!}.
\end{aligned}$$

Observe that each term in the above summation is decreasing over $\tau \in [0,1]$ for any given $b \ge 1$. So is $\kappa(\tau, b)$. Following the same analysis, we can prove it for the function η. ☐

D.2. Proof of Lemma 5

As for the first claim $\eta(\tau, b) \ge 0.602$ for any $\tau \in [0,1]$ and $b \ge 1$: It suffices to show that $\eta(1, b) \ge 0.602$ for any $b \ge 1$ by Lemma 4. We can employ techniques similar to those presented in (Xu, 2023b). Specifically, we can manually compute and verify that $\eta(1, b) \ge 0.602$ for small b values (as shown in Table 4 in Appendix E) and then apply Chernoff bound to establish $\eta(1, b) \ge 0.602$ for large b values.

As for the second claim: Intuitively, for any given pair of parameters (τ, b) with $\tau \in [0,1]$ and $b \ge 1$, the policy with real-time boostings (SM-B) should outperform the counterpart (SM-A) without boostings, which suggests $\eta(\tau, b) \ge \kappa(\tau, b)$. We offer a rigorous proof as follows.

Proof. Consider a given $\tau \in [0,1]$. Let $g(z)$ be a generic function, which is continuous and has the first order of derivative over $z \in [0,1]$. Define

$$\Gamma(\tau, b, g) = \int_0^1 dz \cdot g'(z) \cdot \mathsf{e}^{-\tau \cdot g(z)} \cdot \Pr\Big[\mathsf{Pois}\big((b-\tau) \cdot g(z)\big) \le b-1\Big]$$
$$= \int_0^1 dz \cdot \left\{ g'(z) \cdot \mathsf{e}^{-b \cdot g(z)} \cdot \sum_{\ell=0}^{b-1} \frac{(b-\tau)^\ell \cdot g^\ell(z)}{\ell!} \right\}.$$

Note that when $g(z) = h(z) = \int_0^z d\zeta \cdot \Psi(\zeta)$, where $\Psi(t/T) = \psi_t \ge 1$ for any $\zeta \in [0,1]$ with $\{\psi_t\}$ as defined in (11), we have $\Gamma(\tau, b, h) = \eta(\tau, b)$. In this case, we have $h(z) \ge z$ for $z \in [0,1]$ and $h(0) = 0$ and $h(1) \ge 1$. Meanwhile, when $g(z) = \tilde{h}(z) = z$ for $z \in [0,1]$, we have $\Gamma(\tau, b, \tilde{h}) = \kappa(\tau, b)$. Define

$$\Gamma_\ell(\tau, b, g) = \int_0^1 dz \cdot g'(z) \cdot \mathsf{e}^{-b \cdot g(z)} \cdot \frac{(b-\tau)^\ell \cdot g^\ell(z)}{\ell!},$$

such that $\Gamma(\tau, b, g) = \sum_{\ell=0}^{b-1} \Gamma_\ell(\tau, b, g)$. In the following, we prove that $\Gamma_\ell(\tau, b, h) \ge \Gamma_\ell(\tau, b, \tilde{h})$ by induction over $\ell = 0, 1, \ldots, b-1$, which leads to the claim that $\eta(\tau, b) = \Gamma(\tau, b, h) \ge \Gamma(\tau, b, \tilde{h}) = \kappa(\tau, b)$.

Consider the base case $\ell = 0$. Thus,

$$\Gamma_0(\tau, b, h) = \int_0^1 dz \cdot h'(z) \cdot \mathsf{e}^{-b \cdot h(z)} = \frac{1}{b}\Big(1 - \mathsf{e}^{-b \cdot h(1)}\Big)$$
$$\ge \frac{1}{b}\Big(1 - \mathsf{e}^{-b \cdot \tilde{h}(1)}\Big) = \int_0^1 dz \cdot \tilde{h}'(z) \cdot \mathsf{e}^{-b \cdot \tilde{h}(z)} = \Gamma_0(\tau, b, \tilde{h}),$$

where the inequality above is due to $h(1) \ge 1 = \tilde{h}(1)$. Now assume that $\Gamma_\ell(\tau, b, h) \ge \Gamma_\ell(\tau, b, \tilde{h})$ for all $0 \le \ell \le b-2$, and we show the case for $\ell = b-1$. Note that when $\ell = b-1$,

$$\Gamma_\ell(\tau, b, h) \cdot (\ell!/(b-\tau)^\ell) = \int_0^1 dz \cdot h'(z) \cdot \mathsf{e}^{-b \cdot h(z)} \cdot h^\ell(z)$$
$$= \frac{-\mathsf{e}^{-b \cdot h(1)} \cdot h^\ell(1)}{b} + \frac{\ell}{b} \int_0^1 dz \cdot \mathsf{e}^{-b \cdot h(z)} \cdot h^{\ell-1}(z) \cdot h'(z)$$
$$\ge \frac{-\mathsf{e}^{-b \cdot \tilde{h}(1)} \cdot \tilde{h}^\ell(1)}{b} + \frac{\ell}{b} \int_0^1 dz \cdot \mathsf{e}^{-b \cdot \tilde{h}(z)} \cdot \tilde{h}^{\ell-1}(z) \cdot \tilde{h}'(z) = \Gamma_\ell(\tau, b, \tilde{h}) \cdot (\ell!/(b-\tau)^\ell), \tag{18}$$

which suggests that $\Gamma_\ell(\tau, b, h) \ge \Gamma_\ell(\tau, b, \tilde{h})$. Inequality (18) follows from (1) the function $\mathsf{e}^{-b \cdot x} \cdot x^\ell$ is decreasing over $x \ge 1$, $h(1) \ge \tilde{h}(1) \ge 1$, and parameters ℓ and b satisfies $\ell \le b-1$; and (2) the inductive assumption that $\Gamma_{\ell'}(\tau, b, h) \ge \Gamma_{\ell'}(\tau, b, \tilde{h})$ for all $0 \le \ell' \le b-2$. □

D.3. Proof of Lemma 6

Proof. Consider a given $\tau \in [0,1]$. We first prove that $\kappa(\tau, b) \ge (1 - \mathsf{e}^{-\tau})/\tau - \Theta(b^{-1/2})$.

$$\kappa(\tau, b) = \int_0^1 dz \cdot \mathsf{e}^{-z \cdot \tau} \cdot \Pr\Big[\mathsf{Pois}\big((b-\tau) \cdot z\big) \le b-1\Big] = \int_0^1 dz \cdot \mathsf{e}^{-z \cdot \tau} \cdot \Big(1 - \Pr\Big[\mathsf{Pois}\big((b-\tau) \cdot z\big) \ge b\Big]\Big)$$
$$\ge \int_0^1 dz \cdot \mathsf{e}^{-z \cdot \tau} \left(1 - \exp\left(\frac{-b \cdot (1-z)^2}{2}\right)\right) \Big(\text{by the upper tail bound of Poisson distribution (Canonne, 2020)}\Big)$$
$$= (1 - \mathsf{e}^{-\tau})/\tau - \int_0^1 d\xi \cdot \mathsf{e}^{-(1-\xi) \cdot \tau} \cdot \exp\left(\frac{-b \cdot \xi^2}{2}\right) \quad (\text{setting } 1 - z = \xi)$$
$$\ge (1 - \mathsf{e}^{-\tau})/\tau - \int_0^1 d\xi \cdot \exp\left(\frac{-b \cdot \xi^2}{2}\right)$$
$$= (1 - \mathsf{e}^{-\tau})/\tau - \sqrt{\frac{\pi}{2}} \frac{1}{\sqrt{b}}(1 + o(1)), \quad \big(\text{where } o(1) \text{ is vanishing when } b \to \infty\big).$$

Now we show that $\kappa(\tau, b) \le (1 - \mathsf{e}^{-\tau})/\tau - \Theta(b^{-1/2})$.

$$\begin{aligned}
\kappa(\tau, b) &= \int_0^1 dz \cdot \mathsf{e}^{-\tau \cdot z} \cdot \Pr\left[\mathsf{Pois}\big((b-\tau)\cdot z\big) \le b-1\right] \\
&\le \int_0^1 dz \cdot \mathsf{e}^{-\tau\cdot z} \cdot \Phi\Big(\frac{b-1-(b-\tau)\cdot z}{\sqrt{(b-\tau)\cdot z}}\Big)\big(\text{where } \Phi(\cdot) \text{ is the CDF of the standard normal distribution}\big) \quad (19)\\
&\le \int_0^1 dz \cdot \mathsf{e}^{-\tau\cdot z} \cdot \Phi\Big(\frac{b\cdot(1-z)}{\sqrt{(b-\tau)\cdot z}}\Big) \\
&= (1-\mathsf{e}^{-\tau})/\tau - \frac{\mathsf{e}^{-\tau}}{\sqrt{2\pi}}\frac{1}{\sqrt{b}}(1+o(1)) \ \ \big(\text{where } o(1) \text{ is vanishing when } b \to \infty\big) \\
&\le (1-\mathsf{e}^{-\tau})/\tau - \frac{1}{\sqrt{2\pi \mathsf{e}^2}}\frac{1}{\sqrt{b}}(1+o(1)).
\end{aligned}$$

Note that in Inequality (19), we treat $\mathsf{Pois}((b-\tau)\cdot z)$ as the sum of T *i.i.d.* Bernoulli random variables each with mean $(b-\tau)\cdot z/T$, and then we apply Slud's Inequality (Slud, 1977) with $T \to \infty$. The last equality is obtained by applying techniques similar to the previous case of lower bounding $\kappa(\tau, b)$. ☐

E. Numerical Evaluation of $\kappa(\cdot,\cdot)$ and $\eta(\cdot,\cdot)$

For any given pair of parameters (τ, b), we can directly compute the value following its definition in (7). Thus, we focus on the computation of the functions h and η. Throughout this paper, by default, we maintain the accuracy of numerical values to the fourth decimal place. Due to the space limit, we offer numerical evaluations of κ and η when b and Δ each take values in $1, 2, 3, 4, 5, \infty$ only; see Table 4.

E.1. When $b = 1$: $\kappa(\tau, 1) = 1 - 1/\mathsf{e} \sim 0.6321, \eta(\tau, 1) \sim 0.6924$.

In this case, we see

$$\begin{aligned}
\kappa(\tau, b) &= \int_0^1 dz \cdot \Pr\left[\mathrm{Pois}\big((b-\tau)\cdot z\big) \le b-1\right] \cdot \mathsf{e}^{-z\cdot\tau} = \int_0^1 dz \cdot \mathsf{e}^{-z} = 1 - 1/\mathsf{e}. \\
\eta(\tau, b) &= \int_0^1 dz \cdot h'(z) \cdot \mathsf{e}^{-\tau\cdot h(z)} \cdot \Pr\left[\mathrm{Pois}\big((b-\tau)\cdot h(z)\big) \le b-1\right] \\
&= \int_0^1 dz \cdot \mathsf{e}^{-h(z)} \cdot h'(z) = 1 - \mathsf{e}^{-h(1)}.
\end{aligned}$$

Observe that the differential equation on h in (9) is reduced to $h'' = (h')^3 \cdot \mathsf{e}^{-h-1}, h(0) = 0, h'(0) = 1$ when $b = 1$. We can solve from it that

$$h(1) = 1 - (\mathsf{e}-1)\cdot \mathsf{IV}[\tilde{z}\cdot \mathsf{e}^{\tilde{z}}]\Big(\frac{1}{\mathsf{e}(\mathsf{e}-1)}\Big),$$

where $\mathsf{IV}[f](z) = \tilde{z}$ such that $f(\tilde{z}) = z$ for a function f. Substituting the value $h(1)$ back to η, we get $\eta(\tau, b) = 1 - \mathsf{e}^{-h(1)} \sim 0.6924$. We see that both κ and η have no dependence on τ when $b = 1$.

E.2. When $b = \infty$: $\kappa(\tau, \infty) = \eta(\tau, \infty) = (1 - \mathsf{e}^{-\tau})/\tau$.

From Lemma 6, we see that when $b = \infty$ and any $\tau \in [0, 1]$,

$$\kappa(\tau, b) = (1 - \mathsf{e}^{-\tau})/\tau.$$

Now we focus on the computation of $\eta(\tau, \infty)$. Observe that $\Psi(z) = \psi_t \in [1, 1/(1-1/\mathsf{e}) \sim 1.582]$ with $z = t/T$ and $\{\psi_t\}$ defined in (11). Thus, we claim that $h(z) \in [z, z/(1-1/\mathsf{e})]$ for any $z \in [0, 1]$. Recall that $h(0) = 0$ and $h'(0) = 1$, which satisfies the below equation

$$\begin{aligned}
h'' &= (h')^3 \cdot h^{b-1} \cdot \mathsf{e}^{-b\cdot h-1} \frac{b^b}{(b-1)!} \\
&= (h')^3 \cdot h^{b-1} \cdot \mathsf{e}^{-b\cdot h} \cdot \frac{b}{\mathsf{e}} \cdot \frac{\mathsf{e}^b}{\sqrt{2\pi b}} \quad \big(\text{by applying Stirling's formula to } b!\big) \\
&= (h')^3 \cdot \mathsf{e}^{-b\cdot h+b+(b-1)\ln h} \cdot \sqrt{\frac{b}{2\pi}}\frac{1}{\mathsf{e}} = (h')^3 \cdot h^{-1} \cdot \mathsf{e}^{b\cdot(1+\ln h-h)} \cdot \sqrt{\frac{b}{2\pi}}\frac{1}{\mathsf{e}}.
\end{aligned}$$

We can verify that for any $h \in (0,1)$, $1+\ln h - h < 0$, which implies $h'' = 0$ when $b \to \infty$. Thus, we claim that $h' = 1 = \Psi(z)$ and $h(z) = z$ when $b \to \infty$ and $h \in (0,1)$. By continuity of h, we have $h(z) = z$ for $z \in [0,1]$. Therefore,

$$\begin{aligned}
\eta(\tau, b) &= \int_0^1 dz \cdot h'(z) \cdot \mathsf{e}^{-\tau\cdot h(z)} \cdot \Pr\Big[\mathrm{Pois}\big((b-\tau)\cdot h(z)\big) \le b-1\Big] \\
&= \int_0^1 dz \cdot \mathsf{e}^{-\tau\cdot z} \cdot \Pr\Big[\mathrm{Pois}\big((b-\tau)\cdot z\big) \le b-1\Big] \\
&= (1-\mathsf{e}^{-\tau})/\tau, \quad \big(\text{after taking } b \to \infty\big).
\end{aligned}$$

Thus, we see that $\kappa(\tau,\infty) = \eta(\tau,\infty) = (1-\mathsf{e}^{-\tau})/\tau$.

E.3. When $2 \le b < \infty$

$$\begin{aligned}
\kappa(\tau, b) &= \int_0^1 dz \cdot \Pr\Big[\mathrm{Pois}\big((b-\tau)\cdot z\big) \le b-1\Big] \cdot \mathsf{e}^{-z\cdot\tau} = \int_0^1 dz \cdot \sum_{\ell=0}^{b-1} \mathsf{e}^{-b\cdot z} \cdot \frac{(b-\tau)^\ell \cdot z^\ell}{\ell!}. \\
\eta(\tau, b) &= \int_0^1 dz \cdot h'(z) \cdot \mathsf{e}^{-\tau\cdot h(z)} \cdot \Pr\Big[\mathrm{Pois}\big((b-\tau)\cdot h(z)\big) \le b-1\Big] \\
&= \int_0^1 dz \cdot \sum_{\ell=0}^{b-1} h'(z) \cdot \mathsf{e}^{-b\cdot h(z)} \cdot \frac{(b-\tau)^\ell \cdot h^\ell(z)}{\ell!},
\end{aligned}$$

where h can be solved from the differential equation in (9) as follows:

$$\begin{aligned}
h(z) &= \mathsf{IV}\Big[\big(1-\mathsf{e}^{-1}\big)\cdot\tilde{z} - \mathsf{e}^{-2\tilde{z}-1}\cdot(1+\tilde{z})\Big](z-1/\mathsf{e}) && b=2 \\
&= \mathsf{IV}\Big[\big(1-\mathsf{e}^{-1}\big)\cdot\tilde{z} - \mathsf{e}^{-3\tilde{z}-1}\cdot\Big(1+2\tilde{z}+\frac{3\tilde{z}^2}{2}\Big)\Big](z-1/\mathsf{e}) && b=3 \\
&= \mathsf{IV}\Big[\big(1-\mathsf{e}^{-1}\big)\cdot\tilde{z} - \mathsf{e}^{-4\tilde{z}-1}\cdot\Big(1+3\tilde{z}+4\tilde{z}^2+\frac{8\tilde{z}^3}{3}\Big)\Big](z-1/\mathsf{e}) && b=4 \\
&= \mathsf{IV}\Big[\big(1-\mathsf{e}^{-1}\big)\cdot\tilde{z} - \mathsf{e}^{-5\tilde{z}-1}\cdot\Big(1+4\tilde{z}+\frac{15}{2}\tilde{z}^2+\frac{25\tilde{z}^3}{3}+\frac{125}{24}\tilde{z}^4\Big)\Big](z-1/\mathsf{e}) && b=5
\end{aligned}$$

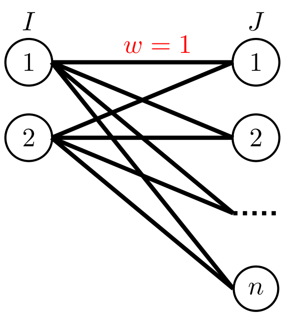
Remarks on the results of Table 4. (1) For each row with a given b value, both $\kappa(\tau,b)$ and $\eta(\tau,b)$ are decreasing over $\tau \in [0,1]$, which is consistent with Theorem 3. (2) The adaptivity gap $\mathsf{AG}(\tau,b) = \eta(\tau,b) - \kappa(\tau,b)$ is a constant of 0.0603 at $b=1$ and 0 at $b=1$, respectively; and in both cases, AG is irrespective of $\tau \in [0,1]$. Additionally, for each given $\tau \in [0,1]$, AG keeps decreasing from $\mathsf{AG} = 0.0603$ at $b=1$ to 0 at $b=\infty$.

F. Proof of Theorem 4

Proof. Consider the example shown in Figure 2. In this scenario, there are two offline tasks, each with a capacity of b, and a total of $|J| = n = 2b$ worker types, each with a unit capacity. There are $K = n = 2b$ skills, and each worker of type j possesses a single skill $k = j \in [n]$ that applies to both tasks. Thus, for each pair $\lambda = (i,k)$, we have $\mathcal{N}_\lambda = \{j = k\}$, and

Table 4. Values of $\kappa(\tau, b)$ and $\eta(\tau, b)$ for different parameters (τ, b) with $\tau = 1 - \mathsf{e}^{-\Delta}$. All fractional values are rounded to the fourth decimal place.

(κ, η)	$\Delta = 1$ $\tau = 0.6321$	2 0.8647	3 0.9502	4 0.9817	5 0.9933	$\cdots$ $\cdots$	∞ 1
$b = 1$	(0.6321, 0.6924)	(0.6321, 0.6924)	(0.6321, 0.6924)	(0.6321, 0.6924)	(0.6321, 0.6924)	$\cdots$	(0.6321, 0.6924)
2	(0.6355, 0.6733)	(0.6009,0.6350)	(0.5882,0.6209)	(0.5836,0.6157)	(0.5818,0.6138)	$\cdots$	(0.5808,0.6127)
3	(0.6472, 0.6771)	(0.6042,0.6303)	(0.5889,0.6137)	(0.5834,0.6077)	(0.5813,0.6055)	$\cdots$	(0.5802,0.6042)
4	(0.6562, 0.6816)	(0.6087,0.6305)	(0.5921,0.6128)	(0.5861,0.6063)	(0.5839,0.6040)	$\cdots$	(0.5826,0.6026)
5	(0.6631, 0.6855)	(0.6127,0.6318)	(0.5952,0.6132)	(0.5889,0.6066)	(0.5867,0.6042)	$\cdots$	(0.5853,0.6027)
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
∞	(0.7412, 0.7412)	(0.6694,0.6694)	(0.6455,0.6455)	(0.6370,0.6370)	(0.6339,0.6339)	$\cdots$	(0.6321,0.6321)



$$
\begin{aligned}
& I = \{1,2\}, J = [n]; \\
& |J| = n = K = 2b; \\
& b_{i=1} = b_{i=2} = b; \\
& b_j = 1, \forall j; \\
& w_\lambda = 1, \forall \lambda = (i,k);
\end{aligned}
$$

$$
\begin{aligned}
& x^*_{ij} = 1/2, \forall (i,j); \\
& y^*_{i,k} = 1/2, \forall (i,k); \\
& \mathsf{Val}(\text{LP -(1)}) = 2b; \\
& \text{OPT} = \mathsf{E}\Big[\min\Big(2b, \sum_{j=1}^{2b} \min(2, \mathsf{Pois}(1))\Big)\Big].
\end{aligned}
$$

Figure 2. An example illustrating the upper bounds on Competitive Ratio (**CR**) for Online Capacitated Coverage Maximization (**OCCM**) with $(\Delta, \tau) = (1, 1 - 1/\mathsf{e})$ due to Benchmark LP (1).

the maximum value of $\Delta = \max_\lambda |\mathcal{N}_\lambda|$ is equal to 1. We set $w_\lambda = 1$ for all $\lambda = (i,k)$. We can verify that: (1) An optimal solution to LP (1) is as follows: $x^*_{ij} = y^*_{ik} = 1/2$ for all (i,j) and (i,k), with an optimal value of $\mathsf{Val}(\text{LP -(1)}) = 2b$. (2) Both an *offline* optimal policy and an *online* optimal policy, denoted by OPT, operate in the same way. They assign each arriving worker j to task $i = 1$ for the first arrival and to task $i = 2$ for the second one, ignoring all future arrivals of j. The expected performance is given by $\text{OPT} = \mathsf{E}\Big[\min\Big(2b, \sum_{j=1}^{2b} \min(2, \mathsf{Pois}(1))\Big)\Big]$. Thus, we claim that any optimal policy can never achieve a competitive ratio better than

$$
\frac{\text{OPT}}{\mathsf{Val}(\text{LP -(1)})} = \frac{\mathsf{E}\Big[\min\Big(2b, \sum_{j=1}^{2b} \min(2, \mathsf{Pois}(1))\Big)\Big]}{2b} = \mathsf{E}\Big[\min\Big(1, \frac{1}{2b}\sum_{j=1}^{2b} \min(2, \mathsf{Pois}(1))\Big)\Big] := \bar{\eta}(b).
$$

□

G. Further Remarks on the Differences between OCCM and Online b-Matching

When every task takes a unit matching capacity ($b = 1$), our model (**OCCM**) is reduced to the well-studied online (bipartite) matching under KIID; see the survey book (Mehta, 2013) for different variants of online matching problems. In this case, each edge $e = (i,j)$ is associated with a positive weight that captures the total utility of all skills covered by j. Note that it may be tempting to cast online b-matching (under KIID) as a special case of **OCCM** as follows. Consider a special setting of **OCCM**, where every task takes a uniform capacity of b and for each task i, every neighbor worker $j \in \mathcal{N}_i$ possesses a set of skills disjoint from each other such that the total weight of skills covered by j for i is equal to the edge weight w_{ij}. In this way, we see that for any $S \subseteq \mathcal{N}_i$, assigning S to i yields a total utility equal to the sum of all edge weights involving i and S. Unfortunately, the reduction above is not correct. For every single edge e with weight w_e, matching e for ℓ times could yield a total weight of $\ell \cdot w_e$ in online b-matching, while it can return at most a utility of w_e in **OCCM**.

H. Generalization from One-Sided Capacitated to Two-Sided Capacitated Case

Recall that in the current setting of **OCCM**, every online agent has a unit matching capacity. Now, we consider the two-sided capacitated case when each online agent (of type) j has a matching capacity $b_j \in \mathbb{Z}^+$, the exact setting considered in (Xu, 2023b). The updated benchmark LP is as follows.

$$
\begin{aligned}
&\max \sum_{\lambda \in \Lambda} w_\lambda \cdot y_\lambda && \\
&y_\lambda \leq 1 - \mathsf{e}^{-\Delta} = \tau && \forall \lambda \in \Lambda \\
&y_\lambda \leq x_\lambda := \sum_{j \in \mathcal{N}_\lambda} x_{ij} && \forall \lambda = (i,k) \in \Lambda \\
&x_i := \sum_{j \in \mathcal{N}_i} x_{ij} \leq b_i && \forall i \in I \\
&x_j := \sum_{i \in \mathcal{N}_j} x_{ij} \leq b_j && \forall j \in J \qquad (20) \\
&0 \leq x_e \leq 1 - 1/\mathsf{e}, y_\lambda \leq 1 && \forall e \in E, \lambda \in \Lambda.
\end{aligned}
$$

The only change made to the previous benchmark LP (1) pertains to Constraint (20), where the right-hand-side value of 1 has been replaced with b_j. The result of Lemma 1 is also applicable in this context. The validity of Constraint (20) can be justified as follows: For each online agent j, the expected number of arrivals is 1, and upon each arrival, it can be matched with up to b_j different offline agents. Consequently, the expected total number of matches for each agent j should not exceed b_j.

We can employ the same Dependent-Rounding (DR)-based approach outlined in (Xu, 2023b) to address the challenges arising from the requirement of multiple matchings for each arrival of an online agent. Specifically, we can adapt SM-A and SM-B to accommodate the non-unit matching capacities for each online agent as follows. All edits added to the previous versions are marked in purple.

Algorithm 3 The modified version of the non-adaptive sampling policy (SM-A) for the two-sided capacitated case.

Offline Phase:
Solve LP (1) for an optimal solution $\{x_e^* | e \in E\}$.
Online Phase:
for $t = 1, 2, \ldots, T$ **do**
  Let an online worker (of type) j arrive at time t.
  Apply Dependent Rounding (DR) (Gandhi et al., 2006) to the static vector $\mathbf{x}_j^* := \big(x_{ij}^* | i \in \mathcal{N}_j\big)$, and let $\mathbf{X}_j = (X_{ij} | i \in \mathcal{N}_j)$ be the rounded binary vector.
  If $X_{ij} = 1$ and i is safe at t (*i.e.,* i's capacity remains), then assign j to i; otherwise, reject j.
end for

Algorithm 4 The modified version of the adaptive policy SM-B parameterized with auxiliary sequences $\{\phi_t, \psi_t\}$, as defined in Equation (11).

Offline Phase:
Solve LP (1) for an optimal solution $\{x_e^* | e \in E\}$.
Online Phase:
for $t = 1, 2, \ldots, T$ **do**
  The first type of attenuations (ATT-I): Apply simulation-based attenuations such that each task is *safe* at (the beginning of) t with probability *equal* to ϕ_t. ▷ A task being safe means its capacity remains then.◁
  Let an online worker (of type) j arrive at t, and $\mathcal{N}_{j,t}$ be the set of *safe* neighbors at t.
  Adaptive boosting (RTB): Apply Dependent Rounding to the boosted vector $\mathbf{x}_{jt}^* := \big(x_{ij}^* \cdot b_j / \sum_{\tilde{i} \in \mathcal{N}_{jt}} x_{\tilde{i}j}^* \mid i \in \mathcal{N}_{jt}\big)$. Let $\mathbf{X}_{jt} = (X_{ijt} | i \in \mathcal{N}_i)$ be the rounded binary vector.
  The second type of attenuations (ATT-II): If $X_{ijt} \geq 1$, match i and j with a certain attenuation factor such that each safe neighbor $i \in \mathcal{N}_{j,t}$ gets sampled and matched with j during t with probability *equal* to $\psi_t \cdot x_{ij}^*$.
end for

Remarks on Algorithms 3 and 4. (1) For Algorithm 3: Due to the degree-preservation property of DR, it is guaranteed that, with probability one, $\sum_{i \in \mathcal{N}_j} X_{ij} \leq \sum_{i \in \mathcal{N}_j} x_{ij}^* \leq b_j$. This ensures that an agent j is assigned to at most b_j neighbors upon every arrival in the updated version of SM-A, as each assignment requires $X_{ij} = 1$. The same analysis leads to the validity of Algorithm 4. (2) DR maintains the marginal distribution, such that $\mathsf{E}[X_{ij}] = x^*_{ij}$ and enforces negative correlations among $\{X_{ij} | i \in \mathcal{N}_j\}$. These two properties ensure that all previous analyses for SM-A continue to be applicable here. The same applies to SM-B.