Dynamic Matching with Post-allocation Service and its Application to Refugee Resettlement

Kirk Bansak
University of California, Berkeley, CA, kbansak@berkeley.edu

Soonbong Lee
Yale School of Management, New Haven, CT, soonbong.lee@yale.edu

Vahideh Manshadi
Yale School of Management, New Haven, CT, vahideh.manshadi@yale.edu

Rad Niazadeh
University of Chicago Booth School of Business, Chicago, IL, rad.niazadeh@chicagobooth.edu

Elisabeth Paulson
Harvard Business School, Boston, MA, epaulson@hbs.edu

Motivated by our collaboration with a major refugee resettlement agency in the U.S., we study a dynamic matching problem where each new arrival (a refugee case) must be matched immediately and irrevocably to one of the static resources (a location with a fixed annual quota). In addition to consuming the static resource, each case requires *post-allocation* service from a server, such as a translator. Given the time-consuming nature of service, a server may not be available at a given time, thus we refer to it as a *dynamic resource*. Upon matching, the case will wait to avail service in a first-come-first-serve manner. Bursty matching to a location may result in undesirable congestion at its corresponding server. Consequently, the central planner (the agency) faces a dynamic matching problem with an objective that combines the matching reward (captured by pair-specific employment outcomes) with the cost for congestion for dynamic resources and over-allocation for the static ones. Motivated by the observed fluctuations in the composition of refugee pools across the years, we design algorithms that do not rely on distributional knowledge constructed based on past years' data. To that end, we develop learning-based algorithms that are asymptotically optimal in certain regimes, easy to interpret, and computationally fast. Our design is based on learning the dual variables of the underlying optimization problem; however, the main challenge lies in the time-varying nature of the dual variables associated with dynamic resources. To overcome this challenge, our theoretical development brings together techniques from Lyapunov analysis, adversarial online learning, and stochastic optimization. On the application side, when tested on real data from our partner agency and incorporating practical considerations, our method outperforms existing ones making it a viable candidate for replacing the current practice upon experimentation.

Key words: refugee matching, post-allocation service, balanced matching, distribution-free algorithms, online learning, online allocation

1. Introduction

According to the United Nations High Commissioner for Refugees (UNHCR), the forcibly displaced population worldwide exceeded 123 million by the end of 2024. This number includes approximately 8 million asylum seekers and more than 37 million refugees (UNHCR 2024). The process of providing assistance to these individuals often involves making high-stakes decisions in the face of complex operational intricacies. Data-driven and algorithmic approaches that capture the unique features of these processes, yet yield easy-to-implement solutions, can significantly enhance these operations and improve the well-being of these vulnerable communities. In this paper, we demonstrate this potential within the context of refugee resettlement in the United States.

This work is motivated by our collaboration with a major refugee resettlement agency in the U.S. Refugee resettlement, recognized as a durable solution to the global refugee crisis, is a process largely overseen by the UNHCR. In this process, refugees are relocated to participating host countries and granted long-term or permanent residence. The U.S. resettles tens of thousands of refugees each year—the largest number among all host countries (UNHCR 2023)—through ten national non-profit resettlement agencies, one of which is our partner. In what follows, we first provide background on the resettlement process, highlighting key features that motivate our modeling framework and research questions. We then provide an overview of our contributions, followed by a discussion of related work.

1.1. Background on Refugee Resettlement

In the following, we briefly describe the refugee resettlement process in the U.S., which is relevant to our partner agency. We note that while exact details vary by country, refugee resettlement and/or asylum procedures of many host countries (e.g., Switzerland, Netherlands, Sweden, etc.) share important commonalities with the process described below.

After the UNHCR allocates a *refugee case*[1] to the U.S. for resettlement, the case is vetted with the help of the federal government[2] to ensure that the refugee case is eligible for resettlement in the U.S. and meets all security and legal criteria before admission and proceeding with the resettlement process. Upon admission to the U.S., the case is handed to one of the national resettlement agencies. The corresponding agency then assigns the case to a local service provider (often referred to as an *affiliate*) within its network. These affiliates typically provide job search and vocational services, as well as assistance in obtaining financial literacy and access to community resources.

Hereafter, for consistency, we refer to this assignment as a *matching*. Importantly, there is strong evidence that the initial matching significantly impacts finding employment within 90 days after

[1] A case typically includes multiple individuals, i.e., members of a family. For simplicity of mathematical exposition, we mostly consider cases with "size one." We incorporate this and other practical considerations later in the paper.

[2] In particular, three federal agencies: (i) the Department of Homeland Security's office of U.S. Citizenship and Immigration Services (USCIS), (ii) the U.S. Department of State's Bureau of Population, Refugees and Migration, and (iii) the U.S. Department of Health and Human Services.

resettlement (Bansak et al. 2018), which is the key integration metric tracked and reported to Congress in the U.S. Additionally, recent work has shown that employment outcomes for any case-affiliate pair can be reasonably predicted using machine learning (ML) models (Bansak et al. 2018, 2024). Equipped with these ML models, a resettlement agency can use the predictions to inform matching decisions.

In using ML predictions to find better matches between refugees and affiliates, the resettlement agency faces several operational considerations:

(i) If a refugee has *U.S. ties* (such as family or close friends) in a particular locality, the agency is required to match the case to the affiliate associated with that locality (Bruno 2017).

(ii) Each affiliate has a target annual *quota* for the number of refugees it receives in the matching, which is approved by the U.S. Department of State with input from the resettlement agencies. These annual quotas serve as soft upper-bounds on the number of refugees to be resettled in a given year. In light of this, the relevant decision-making horizon for the agency is one year.

(iii) When new cases (physically) arrive at matched affiliates, they must receive short-term onboarding services. For example, Global Refuge, a leading U.S. resettlement agency, notes that *"case managers help new arrivals navigate everything from enrolling in English classes and schools to securing jobs and learning how to use public transportation."* (Global Refuge 2024) These services require a nontrivial time investment, and each affiliate has a limited number of case managers. Furthermore, resettlement agencies are expected to provide these services in specific time frames after a case's arrival, and they must report on whether these standards were met. As a result, bursty placement of refugees at a single affiliate can overburden case workers and make it challenging to complete the required onboarding services as required.[3]

(iv) Finally, cases are admitted to the U.S. and handed to agencies over time. As such, the agency must make matching decisions *upon arrival* and without foreknowledge of future arrivals.

The lack of information about future arrivals presents a major challenge in optimizing matching decisions. One approach to addressing this challenge is to assume that the current year's arrival pool resembles that of previous years, and use that as "distributional knowledge" to simulate future arrivals. As we discuss later, several recent papers adopt such an approach; see, for example, the work of Bansak and Paulson (2024) and Ahani et al. (2023). Although natural, this approach falters if the pool's composition changes across years—a phenomenon observed in reality.

To illustrate this, in Figure 1 we focus on five selected affiliates of our partner agency. For each affiliate, we plot the (normalized) number of cases with U.S. ties to this affiliate over the years 2014-2016. We observe a substantial fluctuation in the number of such cases over these years; for example,

[3] Our first-hand communication with the partner agency indicates that avoiding such congestion is a first-order concern: placement officers, though unsystematically, often have an eye toward balancing workloads across affiliates. Beyond our partner, congestion has been a significant operational challenge in other resettlement programs as well. For instance, in Switzerland, localities have at times suspended new placements due to temporary overloads in reception infrastructure (Swiss Federal Parliament 2024).

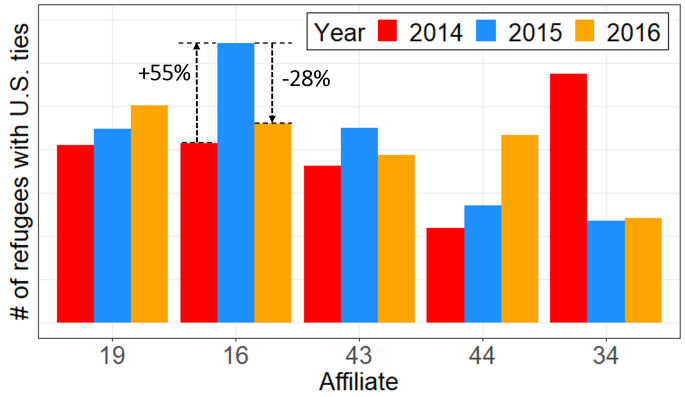


Figure 1 **The number of refugees with U.S. ties at five affiliates normalized by the total number of arrivals.**

the portion of cases tied to affiliate 16 increased by 55% in 2015 compared to 2014. This fluctuation can significantly impact matching decisions. As a toy example, consider this affiliate 16 and suppose that we are making decisions in 2015. If we use data from 2014 to simulate the number of cases with U.S. ties to this affiliate, we substantially underestimate this number. Given that these cases can only be matched to affiliate 16, we may reach the affiliate's quota well before the end of the year.[4]

The above background and observations motivate our main research question:

> *How can we design a dynamic matching algorithm that optimizes employment outcomes without relying on past years' data, while respecting affiliates' quotas and managing congestion for services?*

1.2. Our Contributions

Motivated by the above question, we introduce a framework for dynamic matching with post-allocation service, and develop two learning-based algorithms that do not rely on prior distributional knowledge, such as historical data. Under the assumption that the arrival sequence throughout the year is independent and identically distributed (i.i.d.) from an *unknown* distribution, we show that both algorithms achieve asymptotic optimality in certain regimes. In addition, using data from our partner agency, we conduct a case study demonstrating that our method outperforms existing approaches used by our partner agency or proposed in the literature for similar dynamic refugee matching problems.

A Model for Dynamic Matching with Post-allocation Service (Section 2): Following our motivating application, we use the terminology of refugee resettlement to introduce our model. We consider a discrete-time, finite-horizon model (e.g., one year) where exactly one refugee case arrives in each period.[5] Each case is either "tied" to a specific target affiliate[6] (reflecting the U.S. ties described

[4] In Section EC.11.1, we further elaborate on the distinction between within-year and across-year variation and highlight how this distinction motivates our distribution-free approach to algorithm design for this application.

[5] In some contexts, an agency may encounter batches of arrivals. In Section EC.14, we discuss modifications to our framework and algorithm to handle batched arrivals, and numerically explore their impact in our case study.

[6] Since the resettlement agency is mandated to settle refugees with U.S. ties at *the most proximate* affiliate to their ties (Bansak et al. 2018), we assume that each tied case has exactly one target affiliate.

in Section 1.1) or "free," meaning it can be matched to any affiliate. Upon arrival, the agency also observes the *rewards* of matching the case to each affiliate, representing the employment probabilities predicted by the ML models. Given these observed rewards and knowing whether the case is free or tied, a dynamic matching algorithm must irrevocably match the arriving case to an affiliate (or leave it unmatched),[7] while considering the resource limitations of the affiliates, which we describe next.

We model each affiliate as a *static* resource with fixed capacity, representing its annual quota. Each case uses one unit of the static resource.[8] The most novel aspect of our model is the introduction of *post-allocation* service, capturing the need for service *after matching*, as observed in applications such as refugee resettlement. To model this, we endow each affiliate with a server, also referred to as a *dynamic resource*, whose availability follows an i.i.d. Bernoulli process with a given service rate.[9] Upon becoming available, the server serves a matched case (if any) on a first-come-first-served (FCFS) basis. We measure server congestion via its *backlogs* throughout the horizon, representing the number of matched cases awaiting service at the affiliate in each period. The described service process resembles a classical queueing system, with one crucial distinction: the arrival rate is *endogenously* determined by matching decisions. Despite this distinction, a congested system is still undesirable.

Given the above resource limitations, the agency pursues multiple objectives: maximizing the total matching reward while avoiding excessive congestion for dynamic resources and over-allocation for static resources (i.e., exceeding an affiliate's annual quota).[10] To capture these objectives simultaneously, we define an objective function consisting of the total reward penalized by time-averaged backlog (dynamic resource) and total over-allocation (static resource); see (Objective). Motivated by significant year-to-year fluctuations in the refugee arrival pool (Figure 1), we study this dynamic matching problem with *no knowledge* of the underlying arrival distribution, though we assume that the arrival sequence is i.i.d. from an unknown distribution—and thus potentially learnable over the horizon.

To evaluate the performance of any dynamic matching algorithm, we compare it to a properly defined benchmark (see Definition 2) and introduce the notion of regret relative to such a benchmark. Within this framework, in Sections 3 and 4, we design two algorithms with sub-linear regrets in the horizon length—which are asymptotically optimal or near-optimal in almost all regimes (as established in Section EC.7). Importantly, our regret bounds hold when the congestion penalty parameter is "small enough," meaning that it grows sub-linearly with the horizon. We complement these positive results with an impossibility result (Proposition 2), showing that focusing on this regime is indeed a fundamental requirement: no algorithm can achieve sub-linear regret outside of this regime.

[7] In practice, all cases are matched. In our theoretical model, however, free cases may go unmatched, while we match all tied cases to their target affiliates. In our case study, aligned with practice, we require *all cases* to be matched.

[8] Our model can incorporate multiple-unit consumption (e.g., when a case is a family of multiple individuals, as alluded to earlier) and multiple types of static resources (e.g., school enrollment slots or temporary housing); see Section 2.1.

[9] As we discuss in Section 2, this model is equivalent to having geometric service times with the same service rate.

[10] Due to the presence of tied cases, some over-allocation may be unavoidable (see Section 2).

Before providing an overview of our algorithms, we note that, in the absence of dynamic resources, the above problem reduces to online resource allocation with static resources—a topic extensively explored in the literature (see Section 1.3 for further discussion). However, introducing congestion for dynamic resources changes the nature of the problem (see Proposition 1 and related discussions). Consequently, designing and analyzing algorithms requires new ideas and techniques. In the following, we highlight some of these main challenges and new ideas needed to address them.

Design and Analysis of Algorithms (Sections 3 & 4): The first step in designing our algorithm is to consider the omniscient offline benchmark for the problem—that is, the matching that maximizes our objective function, knowing the entire sample path of arrivals and server availabilities in advance. In Definition 2, we formulate this optimization problem as a simple convex program. In particular, the program includes time-dependent constraints, one for each affiliate at each time period, to track the backlog dynamics (see equation (1)), as well as a simple capacity constraint for each static resource. Consequently, the corresponding dual program has the time-varying dual variables for dynamic resources, in contrast to the time-invariant dual variables for static ones (see Proposition 1). Had we known the "optimal" values of these dual variables, we could use them and directly design a *score-based* optimal algorithm for the primal problem that simply matches each arriving case to the affiliate with the maximum dual-adjusted reward. In the absence of this knowledge, we aim to *learn* these dual variables while making matching decisions online.

The dynamic nature of backlogs, which leads to time-varying dual variables, presents fundamental challenges in designing and analyzing learning-based algorithms, and marks a major departure from the literature on online resource allocation (see Section 1.3 for further discussions). In our first algorithm (Algorithm 1), we address these challenges by introducing a *surrogate dual* problem (Definition 4), which imposes a time-invariance constraint on dual variables associated with dynamic resources. This algorithm maintains estimates of the solution to the surrogate dual problem and makes matching decisions based on the dual-adjusted scores. The key ingredient is then to properly update these dual variables to learn the optimal solution. For dual variables associated with static resources, we rely on classical techniques from online adversarial learning; however, for those associated with dynamic resources, we exploit a crucial structural property linking a specific learning rule to backlog dynamics. The resulting adjustment has an interpretable form: we penalize each affiliate's reward by *a scaled version of its current backlog* (see equation (8) and related discussions).

As our first technical contribution, we prove that this algorithm achieves a sub-linear regret in all regimes of interest (see Theorem 1 and Corollary 1). To establish this result, we combine techniques from adversarial online learning and the drift-plus-penalty method (Tassiulas and Ephremides 1990, Neely 2006, Neely et al. 2008, Neely 2022), defining an appropriate potential function and pseudo-rewards to analyze regret (see Section 3.3). This novel integration of the two frameworks, which to

the best of our knowledge is the first of its kind, enables us to obtain a simultaneous guarantee on the algorithm's total matching reward (Lemma 1) and the average backlog in expectation (Lemma 2). We complement our results by establishing matching regret lower bounds for any online algorithm, demonstrating the asymptotic optimality of the regret upper bounds achieved by our algorithm in nearly all parameter regimes (see Section EC.7 for technical details).

We call our first algorithm "*congestion-aware* dual learning" (`CA-DL`) because it requires exact information on the current backlog. Although this requirement is not a practical concern for our partner agency (see Section 5), such information may not be readily available in other contexts. For example, some agencies may only have access to unreliable or delayed backlog information, rendering it unsuitable for matching decisions. Motivated by this limitation, we develop a second algorithm called "*congestion-oblivious* dual-learning" (`CO-DL`), which learns the dual variables without relying on knowledge of the current backlogs. This algorithm is based on a *surrogate primal* problem (Definition 5), which disregards the congestion penalty in its objective. We design a dual-adjusted score-based algorithm (Algorithm 2) for this problem, similar to `CA-DL`, but with subtle differences in the learning process, including the use of a *time-varying learning rate*, to control the drift of the backlog.

Somewhat surprisingly, we show that even though `CO-DL` does use backlog information, it still achieves a sub-linear regret under most regimes, subject to mild regularity assumptions (Theorem 2). The main technical challenge is analyzing `CO-DL`'s backlog (Lemma 6), which in turn requires studying the "endogenous" arrival rates induced by this algorithm. We overcome this challenge by utilizing a high-probability last-iterate convergence property of the dual variables constructed by `CO-DL` (Proposition 3). Specifically, the dual learner employed by `CO-DL` is a particular variant of online stochastic mirror descent (OSMD) with an appropriately chosen *time-varying* learning rate. Although our proof involves several intricate steps, it crucially exploits the time-varying learning rate, and adapts a recent result on high-probability last-iterate convergence of the stochastic gradient descent from Harvey et al. (2019) to our OSMD variant (see Section 4.2.1 for technical details). While both `CA-DL` and `CO-DL` achieve sub-linear regret in many regimes, we also show that under certain conditions (specifically, slow service rates and certain ranges of penalty parameters), `CO-DL` cannot achieve a sub-linear regret (Theorem 3) unlike `CA-DL`, implying the inherent advantage of explicitly accounting for the backlog.

Finally, we note that our algorithms are computationally fast, and their score-based structure makes them easy to communicate to practitioners (including the managers of our partner agency). Additionally, as we discuss next, our learning-based method outperforms existing approaches (relying on past years data) in the context of our collaboration, when tested on actual data from our partner agency.

Case Study on Refugee Resettlement Data (Section 5): To demonstrate the practical effectiveness of our learning-based approach, we conduct a case study using data from our partner agency. In Section 5.1, we detail the construction of primitives and explain how to adapt our learning-based

approach to this specific context to comply with practical considerations of our partner agency. In Section 5.2, we show that, had the agency used our proposed algorithm on the *actual* sequence of arrivals, it would have led to substantial improvements in terms of predicted employment outcomes and average backlog. We further compare our proposal with another recent proposal based on Bansak and Paulson (2024), which relies on data from past year to predict future arrivals. Compared to this alternative proposal, we show that ours improves the objective function (combining total employment, backlog, and over-allocation) across a broad range of penalty parameters without negatively impacting any of the three outcomes (see Table 1 and Figure EC.5). These promising numerical results, together with the practical advantages of our method discussed earlier, position our method as a strong contender for experimentation to replace the current system.

Beyond refugee resettlement, post-allocation service may arise in assignment problems in other contexts, such as foster care and healthcare, where individuals not only consume long-term resources with fixed capacities (e.g., beds), but also require time-consuming on-boarding services (e.g., initial screenings). In these contexts, undesirable backlog may emerge due to shortages of dynamic resources—see Section 6 for a broader discussion on potential applications of our model. Our framework offers managers flexibility to balance allocation rewards and resource costs (both static and dynamic) to varying degrees. Moreover, the computational efficiency of our algorithms allows policymakers to understand the trade-offs between these objectives by experimenting with different penalty parameters.

1.3. Related Literature

Our work relates and contributes to the literature on refugee matching, online resource allocation, and queueing system control. Below, we highlight the most closely related work, referring the readers to Section EC.2 for a detailed literature review.

In refugee matching, the most relevant comparison is with Bansak and Paulson (2024), who also consider an objective that penalizes congestion but do not consider tied cases—and thus over-allocation is not included in their objective. We adapt their proposed sampling-based algorithm (relying on data from the past year) to handle tied cases in our case study (Section 5), and show that our prior-free algorithms substantially outperform their approach, thanks to its robustness. Unlike previous work including Bansak and Paulson (2024), we also show theoretical performance guarantees of our method.

Methodologically, our study is closely related to Agrawal and Devanur (2014) and Balseiro et al. (2023), who examine online resource allocation with objectives depending only on total allocation. In this special case, the dual problem only has time-invariant variables, which can be learned fast over time to ensure vanishing regret. However, this is not the case in our setting with the average backlog in the objective function, as we have time-varying dual variables associated with backlog dynamics that cannot be learned fast using similar methods. This critical difference necessitates new analytical techniques beyond these papers (and similar papers in the queuing literature). Specifically, our

analysis of `CA-DL` integrates drift-plus-penalty methods and adversarial online learning (Section 3.3, Section EC.5), which is the first of its kinds to the best of our knowledge. For analyzing `CO-DL`, we exploit high-probability last-iterate convergence of stochastic mirror descent (Proposition 3), an aspect less explored in prior literature on online resource allocation.

2. Model and Preliminaries

We start by formally introducing our model for dynamic refugee matching with unknown i.i.d. arrivals and post-allocation service. Specifically, we describe two types of resource limitations in this setting, which we refer to as static and dynamic resource limitations. We then present our objective function combining employment outcomes, over-allocation, and congestion. Next, we define an offline benchmark based on this objective, along with the notion of regret used for performance evaluation. A summary of the main notation and key modeling assumptions is provided in Section EC.1.

Notation: We use $[m]$ to denote $\{1,2,...,m\}$ for any $m \in \mathbb{N}$. We use bold cases to denote vectors. The non-negative orthant in the m-dimensional Euclidean space is denoted by $\mathbb{R}_+^m$, with $\mathbf{e}_i \in \mathbb{R}_+^m$ denoting the standard basis vector at coordinate i. We use $\Delta_m := \{\mathbf{z} \in \mathbb{R}_+^m : \sum_{i=1}^m z_i \leq 1\}$ to denote the m-dimensional standard simplex. The positive part of $x \in \mathbb{R}$ is denoted by $(x)_+$, and an indicator function is denoted by $\mathbb{1}[\cdot]$. Finally, we adopt the standard asymptotic notation. For functions $f, g : \mathbb{R} \to \mathbb{R}$, we write $f(x) = \mathcal{O}(g(x))$ (resp. $f = \Omega(g(x))$) if $|f(x)|$ is upper-bounded (resp. lower-bounded) by a positive constant multiple of $|g(x)|$ for all sufficiently large values of x. If $f(x) = \mathcal{O}(g(x))$ and $f(x) = \Omega(g(x))$, we write it as $f(x) = \Theta(g(x))$. Similarly, we write $f(x) = o(g(x))$ if $\frac{f(x)}{g(x)} \to 0$ as $x \to \infty$. Finally, throughout, a "constant" refers to any scalar independent of T.

Problem Setup: The problem consists of m affiliates and an arrival sequence of T refugee cases, where case t arrives at the beginning of period t. Upon arrival of case t, the agency observes its type denoted by $\mathbf{A}_t := (\mathbf{w}_t, i_t^\dagger)$, which we assume is an i.i.d. random variable drawn from an *unknown* distribution $\mathcal{F}$. The first component of the type is a vector $\mathbf{w}_t = (w_{t,i})_{i\in[m]}$, where $w_{t,i}$ is a case-affiliate pairwise *reward* from matching case t to affiliate i. This reward $w_{t,i}$ can be equivalently thought of as the predicted employment outcome if case t is matched to affiliate i. Without loss of generality, we assume $w_{t,i} \in [0,1]$. The second component of the type is a *target affiliate* denoted by $i_t^\dagger$, which is a predetermined affiliate to which the case must be matched. If $i_t^\dagger \in [m]$, arrival t is a *tied* case targeted to affiliate $i_t^\dagger$. In contrast, we use $i_t^\dagger = 0$ to denote a *free* case, which does not have any target affiliate.

Upon observing $\mathbf{A}_t$, the agency makes an irrevocable matching decision denoted by $\mathbf{z}_t = (z_{t,i})_{i\in[m]}$. To succinctly represent the feasibility set for both tied and free cases, for a given target affiliate $i^\dagger$, we define a *type-feasibility* set $\mathcal{Z}(i^\dagger)$ as $\mathcal{Z}(i^\dagger) = \Delta_m$ if $i^\dagger = 0$ and $\mathcal{Z}(i^\dagger) = \{\mathbf{e}_{i^\dagger}\}$ otherwise. Some comments are in order. First, while the type-feasibility set allows for fractional allocations, our algorithms always make integral decisions. Second, although inaction is not permitted for tied cases, we allow it for free cases for technical reasons. (In our case study in Section 5, consistent with practice, we do not allow

inaction and ensure that every refugee is matched to an actual affiliate.) For ease of exposition, we interpret inaction as matching the case to a dummy affiliate with zero reward and unlimited resources.

Resource Limitations: Each affiliate is endowed with two types of resources: static and dynamic. In the following, we elaborate on each type of resources separately, and formally explain their associated constraints and their corresponding penalty terms in the objective function of the agency.

(i) Static Resource (Capacity): Each affiliate i is endowed with capacity c_i, referred to as a *static* resource, which represents its annual quota. Matching a case to an affiliate consumes one unit of its static resource. We further define $\rho_i = c_i/T$, referred to as the *capacity ratio*. We use $\underline{\rho} = \min_{i\in[m]} \rho_i$ to denote the minimum capacity ratio and impose two mild assumptions on these ratios: (i) $\sum_{i=1}^{m} \rho_i \le 1$[11] and (ii) there exists a constant $d > 0$ such that $\rho_i - \mathbb{P}[i_t^\dagger = i] \ge d$ for all $i \in [m]$, meaning that, in expectation, each affiliate has capacity $\Theta(T)$ for free cases.[12]

The agency aims to respect the endowed capacities as much as possible. Without tied cases, we can use a standard packing constraint to ensure $\sum_{t=1}^{T} z_{t,i} \le c_i$ for all $i \in [m]$. However, due to uncertainty in the number of tied cases, such a constraint is overly stringent, as tied cases can cause over-allocation. To address this, we introduce two constraints in our model. First, we introduce a (weaker) hard constraint to capture that, at any time period, over-allocation can occur only due to the matching of tied cases. Formally, for any $t \in [T]$ and $i \in [m]$, any feasible matching decision must satisfy:

$$\sum_{\tau=1}^{t} \mathbb{1}[i_\tau^\dagger = 0] z_{\tau,i} \le \left(c_i - \sum_{\tau=1}^{t} \mathbb{1}[i_\tau^\dagger = i] z_{\tau,i}\right)_+, \quad \forall i \in [m]. \tag{Capacity Feasibility-t}$$

Because $\mathbf{0} \in \mathcal{Z}(0)$ by our earlier assumption, this constraint can always be satisfied for any arrival sequence. Moreover, it reduces to the aforementioned standard packing constraint for affiliates that do not face tied cases. Second, we introduce a soft constraint through a penalty $\alpha \ge 0$ per unit of over-allocation. Formally, for each affiliate i, the agency incurs an over-allocation cost given by $\alpha\left(\sum_{t=1}^{T} z_{t,i} - c_i\right)_+$. Here, α is a penalty parameter whose magnitude represents the agency's tolerance or physical cost of exceeding capacity.[13]

Throughout the paper, we often use the (total) *net matching reward* to refer to the total reward minus the over-allocation cost, that is, $\sum_{t=1}^{T}\sum_{i=1}^{m} w_{t,i} z_{t,i} - \alpha \sum_{i=1}^{m} \left(\sum_{t=1}^{T} z_{t,i} - c_i\right)_+$. It is important to highlight that, with knowledge of the entire arrival sequence in advance, we would never over-allocate unless the total number of tied cases for an affiliate exceeds its capacity.

[11] This assumption is merely for ease of exposition and without loss of generality, and all of our main results extend to problem instances with arbitrary capacities.

[12] Note that this implies that $\boldsymbol{\rho}$ is also a constant. In practice, capacities are determined by first creating a proportionality index across affiliates and hence scale with T by construction.

[13] For instance, the State Department provides per-capita funding to resettlement agencies in proportion to the approved capacity (U.S. State Department 2023). Hence, the over-allocation penalty parameter α directly captures the burden of securing additional funding per refugee case beyond the agency's initial budget.

(ii) Dynamic Resource (Server): Each case further requires *post-allocation* service. To model this service process, we endow each affiliate with a *server*. Upon matching, the case will wait to receive the service. Each server is a *dynamic* resource, as its availability changes stochastically over time. In particular, we assume that the service availability of affiliate i follows an i.i.d. Bernoulli process with success probability $r_i \in (0,1)$, denoted by $s_{t,i} \sim \texttt{Ber}(r_i)$. We also refer to r_i as the *service rate*. At the end of period t, if the server is available ($s_{t,i} = 1$), it processes one waiting case (if any) in a FCFS manner. This outlined process is equivalent to a (more conventional) service model with random service times, where each service duration at affiliate i is independently drawn from a geometric distribution with mean $1/r_i$. During each period, if the server is available, it initiates service for a new case and remains unavailable until that service is completed. If the server becomes available but no case is waiting, we assume it begins serving an "outside" task (e.g., external cases or other operational task within the agency) with the same service time distribution. Consequently, the generated service token does not carry over to the next time period when a new case arrives.[14]

As motivated in the introduction, server congestion is undesirable. To formalize this notion, we define the *backlog*, denoted by $\mathbf{b}_t = (b_{t,i})_{i\in[m]}$, where $b_{t,i}$ represents the number of cases waiting for service at affiliate i at the end of period t. Formally, the backlog process evolves recursively as:

$$\forall t \in [T]: \qquad b_{t,i} = (b_{t-1,i} + z_{t,i} - s_{t,i})_+ \tag{1}$$

with initial backlog set to zero ($\mathbf{b}_0 = \mathbf{0}$). The backlog grows large when an affiliate becomes "congested" due to bursty matching patterns. Therefore, we use the time-average backlog, that is $\frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{m} b_{t,i}$, as our overall measure of congestion. We further assume that the agency incurs a cost $\gamma \geq 0$ for each unit of the (time-)average backlog. Hence, at the end of the horizon, the agency pays a total congestion cost given by $\frac{\gamma}{T}\sum_{t=1}^{T}\sum_{i=1}^{m} b_{t,i}$. Similar to the over-allocation penalty parameter α, the congestion penalty parameter γ represents the agency's tolerance for congestion.

We assume that the service rate satisfies $r_i \geq \rho_i + \epsilon$ for all $i \in [m]$, where $\epsilon \geq 0$ is referred to as the *service slack*.[15] Assuming $r_i \geq \rho_i$ is well-motivated both technically and practically. From a technical perspective, this assumption comes from the following "stability sanity check": suppose that we match all cases to affiliates without over-allocation. Then, the total number of "arrivals" into affiliate i is $\rho_i T$. To have no backlog at the end of the horizon in expectation, we must have at least $\rho_i T$ periods when the server is available, requiring $r_i \geq \rho_i$. From a practical perspective, each affiliate's quota is

[14] For instance, case workers at our partner agency may engage in administrative tasks or assist other teams when not actively managing refugee cases. In other contexts, however, service staff may remain idle once they become available but find no waiting cases, allowing them to serve the next arrival immediately. This alternative service model is indeed more relaxed compared to our base model, allowing more cases to be served in every sample path. In Section EC.15, we describe how our model and analysis can be modified to obtain similar results in such a setting. In particular, see Section EC.15.1 for this new model. We then show that our algorithms obtain *exactly* the same performance guarantees (Section EC.15.3), and this setting, in some sense, does not make the problem any easier (Section EC.15.2).

[15] Since $r_i < 1$, we implicitly focus on service slack values $\epsilon < 1 - \rho_i$ for all $i \in [m]$.

partly determined based on its service resource availability, making the quota roughly proportional to the available service resources (Bansak and Paulson 2024). In evaluating our proposed algorithms, we distinguish two parameter regimes for the service slack ϵ, formally defined as follows.

DEFINITION 1 **(Stable vs. Near-Critical Regimes).** *Given the service slack $\epsilon \geq 0$, we categorize the problem into the "stable" regime if $\epsilon = \Omega(1)$, and the "near-critical" regime if $\epsilon = \mathcal{O}(1/\sqrt{T})$.*

Definition 1 is inspired by queueing theory, which distinguishes between regimes where the service "slack" (i.e., the difference between service and arrival rates) remains constant (stable regime) and those where it vanishes (heavy-traffic regime). These two regimes often exhibit distinct behaviors. We adopt a parallel distinction in Definition 1. As we show in Sections 3 and 4, our separate analyses under these regimes reveal similarly distinct behaviors.

Information Setting: Before formalizing the agency's objective, we describe the information available for decision-making. When matching case t to an affiliate, the exact realization of server availability $s_{t,i}$ is *not known*, reflecting uncertainty in service times. Regarding backlog information, we distinguish between two settings. In the *congestion-aware* setting (Section 3), the agency observes the current backlog $\mathbf{b}_{t-1}$ at the time of decision making. By contrast, in the *congestion-oblivious* setting (Section 4), the agency does not have any backlog information.

Objective: We now formally define the agency's matching process and its objective. The agency employs a dynamic matching algorithm π, which upon the arrival of case t, makes an immediate and irrevocable matching decision $\mathbf{z}_t^\pi$. To maintain consistency with the literature, we hereafter refer to such a dynamic matching algorithm π as an *online* algorithm. Formally, an online algorithm π is a mapping from an observable history $\mathcal{H}_{t-1}$ and current arrival type $\mathbf{A}_t$ to a matching decision $\mathbf{z}_t^\pi$. In light of the above two information settings, the observable history is given by $\mathcal{H}_{t-1} := \{\mathbf{A}_\tau, \mathbf{s}_\tau, \mathbf{z}_\tau^\pi\}_{\tau=1}^{t-1}$ (resp. $\{\mathbf{A}_\tau, \mathbf{z}_\tau^\pi\}_{\tau=1}^{t-1}$) in the congestion-aware (resp. congestion-oblivious) setting. For a given penalty parameters α and γ, the agency's objective is given by:[16]

$$\mathsf{ALG}^\pi(\alpha,\gamma) := \sum_{i=1}^{m}\sum_{t=1}^{T} w_{t,i} z_{t,i}^\pi - \alpha \sum_{i=1}^{m} \left(\sum_{t=1}^{T} z_{t,i}^\pi - c_i \right)_+ - \frac{\gamma}{T} \sum_{i=1}^{m}\sum_{t=1}^{T} b_{t,i}^\pi \tag{Objective}$$

where $\{\mathbf{b}_t^\pi\}_{t=1}^T$ are the backlog vectors induced by the matching decisions $\{\mathbf{z}_t^\pi\}_{t=1}^T$ through the backlog dynamics in (1). We refer to $\{\mathbf{z}_t^\pi\}_{t=1}^T$ as a *matching profile*. As we highlighted earlier, a feasible matching profile in our model must satisfy (i) $\mathbf{z}_t^\pi \in \mathcal{Z}(i_t^\dagger)$ and (ii) constraints Capacity Feasibility-t, for all $t \in [T]$.

Performance Metric: We compare the performance of any online algorithm against an *optimal offline* benchmark. This benchmark, formally defined below, solves the same optimization problem as online algorithms, but with full foreknowledge of the sequence of arrivals and service availabilities.

[16] We highlight that managing congestion and respecting the annual quota are two completely distinct and incomparable goals. The former is captured by the final term in (Objective), while the latter is captured by the hard constraints Capacity Feasibility-t for all $t \in [T]$ and the second term in (Objective), which functions as a soft constraint.

DEFINITION 2 **(Optimal Offline Benchmark).** *Given knowledge of the sample-path* $\{\mathbf{A}_t, \mathbf{s}_t\}_{t=1}^T$ *of arrival and service availability sequences, the optimal offline benchmark solves this convex program:*[17]

$$\begin{aligned}
\mathsf{OPT}(\alpha,\gamma) := \max_{\substack{\mathbf{z}_t \in \mathcal{Z}(i_t^\dagger) \\ \mathbf{b}_t \geq \mathbf{0}}} & \sum_{i=1}^m \sum_{t=1}^T w_{t,i} z_{t,i} - \alpha \sum_{i=1}^m \left(\sum_{t=1}^T z_{t,i} - c_i \right)_+ - \frac{\gamma}{T} \sum_{i=1}^m \sum_{t=1}^T b_{t,i} \\
s.t. \quad & \sum_{t=1}^T \mathbb{1}[i_t^\dagger = 0] z_{t,i} \leq \left(c_i - \sum_{t=1}^T \mathbb{1}[i_t^\dagger = i] z_{t,i} \right)_+ && \forall i \in [m] && \text{(Capacity Feasibility-}T\text{)} \\
& b_{t,i} \geq b_{t-1,i} + z_{t,i} - s_{t,i} && \forall t \in [T], i \in [m] && \text{(Backlog Inequality)}
\end{aligned}$$

where we define $b_{0,i} = 0$ *for all* $i \in [m]$ *by convention.*

Some comments are in order. First, note that we impose the capacity constraint *only at* $t = T$ when defining the offline benchmark. Hence, the above convex program is a relaxation of our original problem, that is, for any given sample path, $\mathsf{ALG}^\pi(\alpha,\gamma) \leq \mathsf{OPT}(\alpha,\gamma)$. Second, it is straightforward to see that in any optimal solution, at least one of these constraints is binding: $b_{t,i} \geq 0$ or $b_{t,i} \geq b_{t-1,i} + z_{t,i} - s_{t,i}$. Therefore, the optimal solution for $b_{t,i}$ must satisfy the backlog dynamics in (1).

We evaluate an algorithm based on its worst-case performance across all instances, where each instance $\mathcal{I}$ comprises (i) the set of affiliates $[m]$, (ii) the capacity ratios $\boldsymbol{\rho}$, and (iii) the (unknown) arrival type distribution $\mathcal{F}$. With the above benchmark, we define *regret* as the worst-case difference between the expected objective value achieved by the algorithm and that of the optimal offline benchmark.

DEFINITION 3 **(Worst-case Regret).** *The regret of an online algorithm* π *is given by*

$$\mathsf{Regret}_T^\pi := \sup_{\mathcal{I}} \ \mathbb{E}\left[\mathsf{OPT}(\alpha,\gamma) - \mathsf{ALG}^\pi(\alpha,\gamma)\right]. \tag{2}$$

where the expectation is over the arrival distribution $\mathcal{F}$*, the Bernoulli service process with a service rate vector* $\mathbf{r}$*, and (potential) randomness of the algorithm itself.*[18]

Given our benchmark and notion of regret, our goal is to design algorithms that are asymptotically optimal, meaning that their regret grows sub-linearly with T, ideally at a (near) optimal rate.[19]

2.1. Modeling Assumptions and Generalizations with Practical Considerations

Our baseline model imposes the following assumption: (i) each affiliate is endowed with a single type of static resource, and (ii) each case consumes exactly one unit of that resource. These assumptions align with current practice. For example, the annual quota is the only static resource explicitly and currently tracked by our partner agency. Moreover, certain assignment procedures—such as an ongoing

[17] By introducing auxiliary variables, this convex program can equivalently be formulated as a linear program.

[18] For brevity, we omit the dependence of expectation on the arrival and service distributions.

[19] Another common performance measure in the online algorithms literature is the *competitive ratio*—the worst-case ratio of an algorithm's objective to that of an omniscient optimal benchmark. Under our framework, sublinear regret implies a competitive ratio of $1 - o(1)$ as long as the expected value of the offline benchmark (Definition 2) linearly grows in T.

pilot for algorithm-assisted resettlement in Switzerland (Bansak and Paulson 2024)—track capacities at the case level, for which our baseline model remains directly applicable.

Although these assumptions were adopted for simplicity of exposition, our framework can easily be extended to relax them. In Section EC.12, we show how our framework can accommodate *multiple knapsack constraints*, allowing each affiliate to manage several types of static resources (e.g., school enrollment slots in addition to annual quota). Moreover, we show extensions where each case consumes a *varying amount of each resource*. This generalization is particularly relevant to the U.S. refugee resettlement context, where cases often consist of multiple family members, and may consume multiple units of an affiliates annual quota. We also numerically explore this aspect of varying family sizes—alongside other practical considerations relevant to our collaboration—in our case study in Section 5.3.

3. Algorithm Design for Congestion-Aware Setting

In this section, we propose our first learning-based algorithm for the congestion-aware setting. In Section 3.1, we first study the dual program of the optimal offline (Definition 2) and introduce our learning-based approach. Next, in Section 3.2, we formally present the congestion-aware dual-learning algorithm (`CA-DL`) and show its (asymptotically optimal) sub-linear regret guarantees in both stable and near-critical regimes, provided the congestion penalty parameter satisfies $\gamma = o(T)$. We also show an impossibility result (Proposition 2), establishing that the condition $\gamma = o(T)$ is indeed unavoidable to achieve sub-linear regret. Finally, in Section 3.3, we provide an overview of our proof technique.

3.1. Motivation: Surrogate Dual Problem

Our first step is to study the optimal offline benchmark through duality. Observe that, due to our assumption $\rho_i - \mathbb{P}[i_t^\dagger = i] = \Omega(1)$, the total number of tied cases during T periods at each affiliate does not exceed the capacity with high probability for sufficiently large T.[20] Furthermore, under this event, the capacity constraint of the offline benchmark, i.e., (Capacity Feasibility-T) in Definition 2, reduces to the standard packing constraint $\sum_{t=1}^T z_{t,i} \le c_i, \forall i \in [m]$. The following proposition, which we prove in Section EC.3, characterizes the dual program of the optimal offline under this high probability event.

PROPOSITION 1 **(Dual of Optimal Offline).** *For any given sample path $\{\mathbf{A}_t, \mathbf{s}_t\}_{t=1}^T$, consider the following dual program:*

$$\begin{aligned}
\mathsf{Dual}(\alpha,\gamma) := \min_{\boldsymbol{\theta},\boldsymbol{\lambda},\boldsymbol{\beta}_t \ge \mathbf{0}} \sum_{t=1}^T \Big\{ \max_{\mathbf{z}_t \in \mathcal{Z}(i_t^\dagger)} (\mathbf{w}_t - \boldsymbol{\theta} - \boldsymbol{\lambda} - \boldsymbol{\beta}_t)\cdot \mathbf{z}_t + \boldsymbol{\rho}\cdot\boldsymbol{\theta} + \boldsymbol{\rho}\cdot\boldsymbol{\lambda} + \mathbf{s}_t \cdot \boldsymbol{\beta}_t \Big\} \\
s.t. \quad \theta_i \le \alpha \quad \forall i \in [m] \qquad (3) \\
\beta_{t,i} - \beta_{t+1,i} \le \frac{\gamma}{T} \quad \forall t \in [T-1], i \in [m], \qquad \beta_{T,i} \le \frac{\gamma}{T} \quad \forall i \in [m] \qquad (4)
\end{aligned}$$

Then we have the following strong duality: $\mathsf{OPT}(\alpha,\gamma)\mathbb{1}[G_T] = \mathsf{Dual}(\alpha,\gamma)\mathbb{1}[G_T]$ *where G_T is the event that sample path $(\mathbf{A}_t, \mathbf{s}_t)_{t=1}^T$ satisfies $\sum_{t=1}^T \mathbb{1}[i_t^\dagger = i] \le c_i, \forall i \in [m]$.*

[20] Specifically, by the Azuma-Hoeffding inequality and union bound over all affiliates, this event occurs with probability $1 - \mathcal{O}(m\exp(-T))$. Note that our assumption $\underline{\rho} = \Theta(1)$ implies $m = \Theta(1)$ and hence we suppress dependences on m.

In Proposition 1, the dual variables $(\boldsymbol{\theta}, \boldsymbol{\lambda})$ are associated with static resources, corresponding respectively to the over-allocation penalty in the objective (via Fenchel duality) and the hard constraint (Capacity Feasibility-T) (after Lagrangifying this constraint). The vector sequence $\{\boldsymbol{\beta}_t\}_{t=1}^T$ consists of dual variables associated with dynamic resources, each corresponding to constraint (Backlog Inequality) in Definition 2 (after Lagrangifying these constraints). Notably, the sequence $\boldsymbol{\beta}_t$ is time-dependent, in contrast to $(\boldsymbol{\theta}, \boldsymbol{\lambda})$, which arises from constraints (4) that are coupling consecutive $\boldsymbol{\beta}_t$ and $\boldsymbol{\beta}_{t+1}$.[21]

Proposition 1 implies a simple method for optimal matching decisions given the optimal dual variables $(\boldsymbol{\theta}^*, \boldsymbol{\lambda}^*, \{\boldsymbol{\beta}_t^*\}_{t=1}^T)$: at time t, choose the decision $\mathbf{z}_t \in \mathcal{Z}(i_t^\dagger)$ that maximizes the *dual-adjusted score* $(\mathbf{w}_t - \boldsymbol{\theta}^* - \boldsymbol{\lambda}^* - \boldsymbol{\beta}_t^*) \cdot \mathbf{z}_t$. Motivated by this observation, we adopt a *learning-based* approach, iteratively updating the dual variables and choosing the matching decisions that maximize the dual-adjusted score based on the learned values. Due to the time invariance of $(\boldsymbol{\theta}^*, \boldsymbol{\lambda}^*)$ and the stationarity of arrivals and service availabilities, we can efficiently learn the static dual variables, similar to Agrawal and Devanur (2014), using a first-order method that updates via the (stochastic) gradient of the dual function upon each arrival. However, directly learning the time-varying sequence $\{\boldsymbol{\beta}_t^*\}_{t=1}^T$ subject to the inter-temporal constraints (4) appears challenging, if not infeasible. To overcome this challenge, we introduce the following surrogate dual problem.

Definition 4 **(Surrogate Dual Problem)**. *For any given sample path* $\{\mathbf{A}_t, \mathbf{s}_t\}_{t=1}^T$, *define the per-period surrogate dual function for period* t *as*

$$g_t(\boldsymbol{\theta}, \boldsymbol{\lambda}, \boldsymbol{\beta}) := \max_{\mathbf{z}_t \in \mathcal{Z}(i_t^\dagger)} \left\{ (\mathbf{w}_t - \boldsymbol{\theta} - \boldsymbol{\lambda} - \boldsymbol{\beta}) \cdot \mathbf{z}_t \right\} + \boldsymbol{\rho} \cdot \boldsymbol{\theta} + \boldsymbol{\rho} \cdot \boldsymbol{\lambda} + \mathbf{s}_t \cdot \boldsymbol{\beta}. \tag{5}$$

The surrogate dual problem is then defined by

$$\mathsf{Dual}^{\mathsf{S}}(\alpha) := \min_{\boldsymbol{\theta}, \boldsymbol{\lambda}, \boldsymbol{\beta} \geq \mathbf{0}} \sum_{t=1}^T g_t(\boldsymbol{\theta}, \boldsymbol{\lambda}, \boldsymbol{\beta}) \quad s.t. \quad \theta_i \leq \alpha \quad \forall i \in [m] \tag{6}$$

Compared to the original dual program, the surrogate dual problem addresses the time-varying nature of $\{\boldsymbol{\beta}_t^*\}_{t=1}^T$ by considering a restricted program in which these dual variables remain *identical* across periods. As we show later, this restriction leads to an important advantage in controlling the congestion. Specifically, it enables us to uncover a crucial structural connection between the backlog dynamics of each affiliate and the trajectory of an online projected subgradient descent used to learn the time-invariant dual solution $\boldsymbol{\beta}^*$ of this surrogate problem.

3.2. Algorithm and Analysis

We formally present our first algorithm, called `CA-DL`, in Algorithm 1. At a high level, `CA-DL` maintains dual variables for the surrogate dual problem, upon which it bases its primal decision.

[21] Indeed, by complementary slackness, we can show that the optimal dual variables satisfy $\beta_{t,i}^* - \beta_{t+1,i}^* > 0$ if the backlog of the optimal offline at period t and affiliate i is nonzero.

Algorithm 1 Congestion-Aware Dual-Learning (`CA-DL`) Algorithm

1: **Input:** T, $\boldsymbol{\rho}$, η, and ζ

2: Initialize $\theta_{1,i} \leftarrow \exp(-1)$, $\lambda_{1,i} \leftarrow \exp(-1)$, $c_{0,i} \leftarrow \rho_i T$, and $b_{0,i} \leftarrow 0$ for all $i \in [m]$

3: **for** each arrival $\{\mathbf{A}_t\}_{t=1}^T$ **do**

4: **if** $i_t^\dagger \in [m]$ **then** set $\mathbf{z}_t = \mathbf{e}_{i_t^\dagger}$

5: **else if** $\min_{i\in[m]} c_{t-1,i} > 0$ **then** set $\mathbf{z}_t \in \arg\max_{\mathbf{z}\in\Delta_m} (\mathbf{w}_t - \boldsymbol{\theta}_t - \boldsymbol{\lambda}_t - \zeta \mathbf{b}_{t-1}) \cdot \mathbf{z}$

6: **else** set $\mathbf{z}_t = \mathbf{0}$

7: Update for all $i \in [m]$: $c_{t,i} \leftarrow c_{t-1,i} - z_{t,i}, \quad b_{t,i} \leftarrow (b_{t-1,i} + z_{t,i} - s_{t,i})_+$

8: Update dual variables for all $i \in [m]$:

$$\theta_{t+1,i} \leftarrow \min\{\theta_{t,i} \exp(\eta(z_{t,i} - \rho_i)), \alpha\}, \quad \lambda_{t+1,i} \leftarrow \min\left\{\lambda_{t,i} \exp(\eta(z_{t,i} - \rho_i)), \frac{1+2\alpha}{\underline{\rho}}\right\} \tag{7}$$

`// The dual variable for the dynamic resource is implicitly updated as` $\beta_{t+1,i} = (\beta_{t,i} + \zeta(z_{t,i} - s_{t,i}))_+$ `with` $\beta_{0,i} = 0$`, so` $\beta_{t+1,i} = \zeta b_{t,i}$.

9: **end for**

Specifically, the algorithm uses primal decisions to obtain gradient information for updating dual variables via online learning. It explicitly updates dual variables $(\boldsymbol{\theta}, \boldsymbol{\lambda})$ associated with static resources using online mirror descent, and implicitly updates dual variables $\boldsymbol{\beta}$ for dynamic resources (in the surrogate dual problem) through backlog dynamics. Consequently, the scaled backlog effectively serves as the "correct" dual variable for the dynamic resource. We also note that the algorithm is initialized with certain dual values, and also and learning rates $\eta \geq 0$ for static resources and $\zeta \geq 0$ for dynamic resources. For each arrival t, the algorithm proceeds in two phases.

Primal Phase (line 4-6): The algorithm selects the decision that maximizes the dual-adjusted score based on the current dual variables $(\boldsymbol{\theta}_t, \boldsymbol{\lambda}_t, \zeta \mathbf{b}_{t-1})$, subject to the capacity constraint and $\mathbf{z}_t \in \mathcal{Z}(i_t^\dagger)$. Specifically, the algorithm matches the arriving free case to the affiliate i with the highest dual-adjusted score $w_{t,i} - \theta_{t,i} - \lambda_{t,i} - \zeta b_{t-1,i}$ (with arbitrary selection among all affiliates that achieve the maximum), provided that (i) the chosen affiliate's adjusted score is non-negative and (ii) every affiliate has remaining capacity.[22] Otherwise, the case is matched to the dummy affiliate. By construction, our algorithm satisfies (Capacity Feasibility-t) for all $t \in [T]$.

Dual Phase (line 8): Based on the primal decision $\mathbf{z}_t$, we update the dual variables using online learning. Focusing first on the update rule (7), note that $\boldsymbol{\rho} - \mathbf{z}_t$ is a subgradient of the per-period surrogate dual function g_t (see equation (5) in Definition 4) with respect to $\boldsymbol{\theta}$, evaluated at the current assignment of dual variables $(\boldsymbol{\theta}_t, \boldsymbol{\lambda}_t, \zeta \mathbf{b}_{t-1})$. We use this gradient to perform the multiplicative weight update (Freund et al. 1997), which is a special case of online mirror descent. The same principle applies

[22] This second condition is mostly needed for a technical reason in our theoretical analysis. See Remark 2.

to the update rule for $\boldsymbol{\lambda}$.[23] The upper bound on $\boldsymbol{\theta}$ comes from the domain of $\boldsymbol{\theta}$ in the surrogate dual problem (6). The upper bound on $\boldsymbol{\lambda}$ is imposed for technical reasons.[24]

In addition to explicitly tracking the dual variables $(\boldsymbol{\theta}, \boldsymbol{\lambda})$ for the static resource, our algorithm also *implicitly* tracks the dual variable $\boldsymbol{\beta}$ (in the surrogate dual problem) for dynamic resources via the backlog dynamics. To see this, suppose we update the sequence $\{\boldsymbol{\beta}_t\}_{t=1}^T$, initialized with $\boldsymbol{\beta}_1 = \mathbf{0}$, using a slightly different (and simpler) update rule—specifically, vanilla projected gradient descent (Zinkevich 2003) with the non-negative orthant $\boldsymbol{\beta}_t \geq \mathbf{0}$ as the constraint set.[25] Note that $\mathbf{s}_t - \mathbf{z}_t$ is a subgradient of g_t with respect to $\boldsymbol{\beta}$. Given a learning rate ζ, we obtain:

$$\beta_{t+1,i} = (\beta_{t,i} + \zeta(z_{t,i} - s_{t,i}))_+. \tag{8}$$

We observe that, up to a scalar factor ζ, the update rule (8) is equivalent to the backlog dynamics (1). By induction over period t, we can show that $\beta_{t,i} = \zeta b_{t-1,i}$ for all $t \in [T]$ and $i \in [m]$. In other words, the scaled backlog acts as the dual variable for dynamic resources. Intuitively, `CO-DL` manages congestion by "penalizing" a scaled version of the current backlog.[26]

We now formally analyze the regret of `CA-DL`, first focusing on the stable regime (i.e., $\epsilon = \Omega(1)$).

THEOREM 1 **(Regret of `CA-DL` under Stable Regime).** *Let $\eta = \Theta(1/\sqrt{T})$ and $\zeta = \Theta(1/\sqrt{T})$. The regret (Definition 3) of `CA-DL` is $\mathcal{O}(\sqrt{T} + \frac{\gamma}{\epsilon})$ for any service slack parameter $\epsilon > 0$. In particular, under the stable regime (Definition 1), the regret of `CA-DL` is $\mathcal{O}(\sqrt{T} + \gamma)$.*

We sketch the proof of Theorem 1 in Section 3.3. For the near-critical regime (i.e., $\epsilon = \mathcal{O}(1/\sqrt{T})$), we complement Theorem 1 with the following corollary, proved in Section EC.6.6. This corollary establishes that `CA-DL` also achieves sub-linear regret in this regime (with a different choice of ζ).

COROLLARY 1 **(Regret of `CA-DL` under Near-Critical Regime).** *Let $\eta = \Theta(1/\sqrt{T})$ and $\zeta = \Theta(\sqrt{\gamma/T})$. Under the near-critical regime (Definition 1), the regret (Definition 3) of `CA-DL` is $\mathcal{O}(\sqrt{\gamma T})$.*

Theorem 1 and Corollary 1 imply that `CA-DL`'s regret is sub-linear in T (in both stable and near-critical regimes) unless $\gamma = \Omega(T)$. A natural question, therefore, is whether sub-linear regret is achievable when $\gamma = \Omega(T)$. We rule out this possibility with the following lower bound result, concluding that `CA-DL` attains sub-linear regret for *all* feasible values of γ for which such a small regret is achievable.

PROPOSITION 2 **(Lower Bound on Achievable Regret).** *For $\gamma = \Omega(T)$ and any given $\epsilon \geq 0$ such that the resulting service rates satisfy $r_i \in [\rho_i + \epsilon,\ 1)$ for all $i \in [m]$, there exists an instance for which the regret (Definition 3) of any online algorithm (even in the congestion-aware setting) is $\Omega(T)$.*

[23] Specifically, the multiplicative dual update in equation (7) is a special case of mirror descent with the mirror map $h(\mathbf{x})$ being the negative entropy function (i.e., $h(\mathbf{x}) = \sum_i x_i \log(x_i)$). See Bubeck (2011) for details.

[24] Since negative entropy is strongly convex only within a bounded domain, we impose an upper bound on $\boldsymbol{\lambda}$.

[25] This update rule is a special case of mirror descent when the mirror map $h(\mathbf{x})$ is the squared Euclidean norm.

[26] Our algorithm design also has an intimate connection to Lyapunov optimization, particularly the drift-plus-penalty method (Neely 2022). We elaborate on this connection in Section EC.5.

We provide a formal proof of Proposition 2 in Section EC.4. To sketch the proof, we construct an instance with a single affiliate with capacity ratio $\rho = 0.5$ and service rate $r = \rho + \epsilon$, where $\epsilon \in [0, 1-\rho)$. Each arriving case t has type $\mathbf{A}_t = (1, 0)$—that is, each arrival is a free case with a reward of one. Since there is only one actual affiliate, the decision reduces to whether or not to match each case to this affiliate. In this instance, the offline optimal benchmark can leverage its foreknowledge of service availability to avoid backlogs, matching cases to the affiliate only when the server is available. In the full proof, we show that this offline strategy yields an expected total reward of at least $0.5T - \Theta(\sqrt{T})$ with zero average backlog (note that the maximum expected total reward is $0.5T$). In contrast, because online algorithms do not have information about current service availability, they inevitably incur backlog without achieving greater reward. Based on this intuition, we show that any online algorithm attaining $\Theta(T)$ reward must incur congestion cost of $\Theta(\gamma)$, leading to the desired result.

We dedicate the next subsection to outlining the proof of Theorem 1. The proof of Corollary 1 follows similar ideas. The main challenge in establishing the regret guarantee arises from the objective function defined in (Objective), which consists of two fundamentally different components: the net matching reward and the time-average backlog. The key innovation of our analysis is to (i) define pseudo-rewards that implicitly incorporate both components and (ii) combine techniques from the celebrated drift-plus-penalty method (Neely 2022) and adversarial online learning to bound these pseudo-rewards in terms of the optimal offline solution (Definition 2), and thus establish low regret. Roughly speaking, the two terms appearing in our regret bound correspond respectively to separate bounds we establish for the net matching reward and congestion cost (see Lemmas 1 and 2 in Section 3.3).

Remark 1 (**Matching Lower Bounds**). While our regret definition uses $\mathsf{OPT}(\alpha, \gamma)$ as the benchmark, the proof of Theorem 1 establishes a stronger result: the regret bound holds even when compared to $\mathsf{OPT}(\alpha, 0)$, which does not incur the congestion penalty and solely maximizes the net matching reward. In Section EC.7, we provide matching lower bounds for this stronger benchmark—corresponding to the upper bounds in Theorem 1 (the stable regime) and Corollary 1 (the near-critical regime when $\epsilon = 0$)—that (almost) establish the asymptotic optimality of our upper-bounds. We also provide a near-matching lower bound for the weaker benchmark $\mathsf{OPT}(\alpha, \gamma)$ in the stable regime.

Remark 2 (**Stopping Time**). For technical reasons, we assume that Algorithm 1 matches free cases to a dummy affiliate once any affiliate's capacity is depleted. In Section EC.6.5.1, we prove that this "stopping time" occurs near the end of the process. Furthermore, we show that Theorem 1 remains unchanged even if the dummy affiliate is removed before this stopping time (see Section EC.6.5.2).

3.3. Proof Sketch of Theorem 1

We first introduce some notation. Hereafter, a sample path refers to a sequence of arrivals and service availabilities over the horizon, i.e., $\{\mathbf{A}_t, \mathbf{s}_t\}_{t=1}^T$. We use $\{\mathbf{z}_t\}_{t=1}^T$ and $\{\mathbf{z}_t^*\}_{t=1}^T$ to denote the matching profile of `CA-DL` and the optimal offline benchmark (Definition 2), respectively. Let $\mathsf{NMR}(\cdot; \alpha)$ denote the net matching reward of a feasible matching profile given the over-allocation penalty parameter α:

$$\mathsf{NMR}(\{\hat{\mathbf{z}}_t\}_{t=1}^T;\alpha)=\sum_{t=1}^{T}\sum_{i=1}^{m} w_{t,i}\hat{z}_{t,i}-\alpha\sum_{i=1}^{m}\left(\sum_{t=1}^{T}\hat{z}_{t,i}-c_i\right)_+. \tag{9}$$

Because the congestion cost is non-negative, $\mathsf{NMR}(\{\mathbf{z}_t^*\}_{t=1}^T;\alpha)\geq\mathsf{OPT}(\alpha,\gamma)$ for every sample path and every penalty parameter pair (α,γ). Hence, for any arrival distribution $\mathcal{F}$, we have this decomposition:

$$\mathbb{E}[\mathsf{OPT}(\alpha,\gamma)-\mathsf{ALG}^{\texttt{CA-DL}}(\alpha,\gamma)]\leq\underbrace{\mathbb{E}[\mathsf{NMR}(\{\mathbf{z}_t^*\}_{t=1}^T;\alpha)]-\mathbb{E}[\mathsf{NMR}(\{\mathbf{z}_t\}_{t=1}^T;\alpha)]}_{\text{(A)=Loss of net matching reward}}+\gamma\cdot\underbrace{\mathbb{E}\Big[\frac{1}{T}\sum_{t=1}^{T}\|\mathbf{b}_t\|_1\Big]}_{\text{(B)=Average backlog}}. \tag{10}$$

In light of this decomposition, it suffices to separately upper bound terms (A) and (B). The following two key lemmas establish the desired upper bounds for each term.

LEMMA 1 **(Bounding Loss of Net Matching Reward).** *For any arrival distribution $\mathcal{F}$ and $\epsilon\geq 0$, we have*

$$\mathbb{E}[\mathsf{NMR}(\{\mathbf{z}_t^*\}_{t=1}^T;\alpha)]-\mathbb{E}[\mathsf{NMR}(\{\mathbf{z}_t\}_{t=1}^T;\alpha)]\leq\mathcal{O}(\sqrt{T}). \tag{11}$$

LEMMA 2 **(Bounding Average Backlog).** *For any arrival distribution $\mathcal{F}$ and $\epsilon>0$, we have*

$$\mathbb{E}\Big[\frac{1}{T}\sum_{t=1}^{T}\|\mathbf{b}_t\|_1\Big]\leq\mathcal{O}\Big(\frac{1}{\epsilon}\Big). \tag{12}$$

Combined with the decomposition (10), the two lemmas imply Theorem 1. The remainder of this section is devoted to sketching the proofs of Lemmas 1 and 2. The analysis consists of three steps: In Step 1, we define a pseudo-reward—a stochastic process designed to facilitate the comparison between the objective values of `CA-DL` and the optimal offline benchmark. This pseudo-reward accounts not only for the immediate reward of a match, but also for the opportunity costs of using a unit of capacity at each affiliate, the immediate marginal cost of over-allocation, and the marginal effect of a new match on congestion at the affiliates. In Step 2, we establish lower and upper bounds on the pseudo-rewards that crucially use the design of `CA-DL`. Finally, Step 3, we combine these bounds to complete the proof.

Step 1: Defining a Pseudo-reward. Consider the following potential function, which is a commonly used Lyapunov function in the literature to show stability in dynamical systems (e.g., switched queuing networks; see Stolyar (2004), Eryilmaz and Srikant (2007), Neely et al. (2008)), and its drift.

$$\psi(\mathbf{b}_t):=\frac{1}{2}\|\mathbf{b}_t\|_2^2,\qquad D_t:=\psi(\mathbf{b}_t)-\psi(\mathbf{b}_{t-1}). \tag{13}$$

One simple yet powerful property of this potential function ψ is the following lemma.

LEMMA 3 **(Drift Lemma).** *For all sample paths,*

$$\mathbf{b}_{t-1}\cdot(\mathbf{z}_t-\mathbf{s}_t)\leq D_t\leq\mathbf{b}_{t-1}\cdot(\mathbf{z}_t-\mathbf{s}_t)+\mathcal{O}(1).$$

We prove Lemma 3 in Section EC.6.1. Lemma 3 states that the drift of the potential function ψ can be upper and lower bounded by a linear function of the current backlog. With this background, we are ready to define the pseudo-reward at time t, denoted by K_t, as follows:

$$K_t := \mathbf{w}_t \cdot \mathbf{z}_t + \boldsymbol{\theta}_t \cdot (\boldsymbol{\rho} - \mathbf{z}_t) + \boldsymbol{\lambda}_t \cdot (\boldsymbol{\rho} - \mathbf{z}_t) - \zeta D_t \tag{14}$$

To better understand our design of pseudo-rewards, we observe that Lemma 3 implies

$$K_t = (\mathbf{w}_t - \boldsymbol{\theta}_t - \boldsymbol{\lambda}_t - \zeta \mathbf{b}_{t-1}) \cdot \mathbf{z}_t + \boldsymbol{\theta}_t \cdot \boldsymbol{\rho} + \boldsymbol{\lambda}_t \cdot \boldsymbol{\rho} + \zeta \mathbf{b}_{t-1} \cdot \mathbf{s}_t - \mathcal{O}(\zeta) \tag{15}$$

for every sample path. The first term $(\mathbf{w}_t - \boldsymbol{\theta}_t - \boldsymbol{\lambda}_t - \zeta \mathbf{b}_{t-1}) \cdot \mathbf{z}_t$ is the adjusted score that `CA-DL` maximizes (in the primal phase), allowing us to use the optimality criterion of `CA-DL` to lower bound the sum of pseudo-rewards. Moreover, setting $\boldsymbol{\beta}_t = \zeta \mathbf{b}_{t-1}$ and ignoring the additive error $\mathcal{O}(\zeta)$, the pseudo-reward is *exactly* $g_t(\boldsymbol{\theta}_t, \boldsymbol{\lambda}_t, \boldsymbol{\beta}_t)$, the per-period surrogate dual function evaluated at the current estimate of dual variables (see equation (5) in Definition 4). This connection enables us to use online learning (in the dual phase) to upper bound the sum of pseudo-rewards. By carefully comparing these lower and upper bounds, we establish the desired upper bounds on both the loss in net matching reward (Lemma 1) and the average backlog (Lemma 2). We elaborate on these steps next.

Step 2: Lower and Upper Bounding the Pseudo-rewards. We provide a lower bound on the expected cumulative pseudo-reward up to the last time that no hard resource constraint, i.e., (Capacity Feasibility-t) for $t \in [T]$, is binding for `CA-DL`. Formally, we define the *stopping time* as follows:

$$T_A := \min \Big\{ t \le T : \sum_{\tau=1}^{t} z_{\tau,i} \ge c_i \text{ for some } i \in [m] \Big\}. \tag{16}$$

For all $1 \le t \le T_A$, the primal phase of `CA-DL` (lines 4-6) can be succinctly written as

$$\mathbf{z}_t \in \arg\max_{\mathbf{z} \in \mathcal{Z}(i_t^\dagger)} (\mathbf{w}_t - \boldsymbol{\theta}_t - \boldsymbol{\lambda}_t - \zeta \mathbf{b}_{t-1}) \cdot \mathbf{z}. \tag{17}$$

The following lemma provides a lower bound on the total expected pseudo-reward up to the stopping time, expressed (in part) in terms of (i) the expected net matching reward of *any* matching profile subject to the same constraint as the offline benchmark (Definition 2) and (ii) the backlog of `CA-DL`.

LEMMA 4 **(Lower Bound on Pseudo-Rewards).** *For any feasible matching profile $\{\hat{\mathbf{z}}_t\}_{t=1}^T$ that satisfies $\hat{\mathbf{z}}_t \in \mathcal{Z}(i_t^\dagger)$ for all $t \in [T]$ and* (Capacity Feasibility-T)*, we have*

$$\mathbb{E}\Big[\sum_{t=1}^{T_A} K_t\Big] \ge \mathbb{E}[\mathsf{NMR}(\{\hat{\mathbf{z}}_t\}_{t=1}^T; \alpha)] + \zeta\epsilon \mathbb{E}\Big[\sum_{t=1}^{T_A - 1} \|\mathbf{b}_t\|_1\Big] - (T - T_A) - \mathcal{O}(\zeta T) \tag{18}$$

We prove Lemma 4 in Section EC.6.2. The proof crucially relies on the optimality criterion (17) of `CA-DL`. Specifically, we compare the decision of `CA-DL` at time t to that of a *static* control (see Claim EC.5 in Section EC.6.2), whose expected value serves as an upper bound on the per-period net matching reward of any matching profile subject to the same constraint as the offline benchmark.

The following lemma upper bounds the pseudo-rewards via the net matching reward of `CA-DL`.

LEMMA 5 **(Upper Bound on Pseudo-Rewards).** *For every sample path, we have*

$$\sum_{t=1}^{T_A} K_t \le \mathsf{NMR}(\{\mathbf{z}_t\}_{t=1}^T;\alpha) - (T-T_A) - \zeta\psi(\mathbf{b}_{T_A}) + \mathcal{O}(\sqrt{T}) \tag{19}$$

The proof of Lemma 5 is presented in Section EC.6.3. Similar to Agrawal and Devanur (2014) and Balseiro et al. (2023), our proof leverages the adversarial online learning regret guarantee of the online mirror descent algorithm used to update the dual variables for static resources (see update rules (7) and related discussion). Importantly, this regret guarantee holds for *every* sample path and is oblivious to the primal decisions made by `CA-DL`, allowing us to directly apply this guarantee in our analysis.

Step 3: Putting Everything Together. Combining Lemma 4 and Lemma 5 and taking expectations, we arrive at the following crucial inequality: For any matching profile $\{\hat{\mathbf{z}}_t\}_{t=1}^T$ that satisfies $\hat{\mathbf{z}}_t \in \mathcal{Z}(i_t^\dagger)$ for all $t \in [T]$ and constraint (Capacity Feasibility-T),

$$\underbrace{\mathbb{E}[\mathsf{NMR}(\{\hat{\mathbf{z}}_t\}_{t=1}^T;\alpha)] - \mathbb{E}[\mathsf{NMR}(\{\mathbf{z}_t\}_{t=1}^T;\alpha)]}_{\text{(A): Lemma 1}} + \underbrace{\zeta\epsilon\mathbb{E}\Big[\sum_{t=1}^{T_A-1}\|\mathbf{b}_t\|_1\Big] + \zeta\mathbb{E}[\psi(\mathbf{b}_{T_A})]}_{\text{(B'): Lemma 2}} \le \mathcal{O}(\sqrt{T}+\zeta T) \tag{20}$$

The remaining steps of the proof are (i) establishing upper bounds on terms (A) and (B') by choosing an appropriate feasible matching profile $\{\hat{\mathbf{z}}_t\}_{t=1}^T$, leading to Lemmas 1 and 2, respectively, and (ii) controlling the backlog accrued after the stopping time. We elaborate these steps in Section EC.6.4.

4. Algorithm Design for Congestion-Oblivious Setting

As evident from Algorithm 1, `CA-DL` requires access to up-to-date backlog information. While this is not a practical concern for our partner agency (see Section 5), such information might not be available in other contexts. Motivated by this, we consider the congestion-oblivious setting and propose a dual-learning algorithm (`CO-DL`) that does not rely on backlog information. In Section 4.1, we motivate our technical approach. In Section 4.2, we formally introduce `CO-DL` and establish its sub-linear regret in the stable regime under mild conditions. We further show that in certain regimes where `CA-DL` still achieves sub-linear regret, `CO-DL` does not—highlighting the inherent benefit of accounting for backlog.

4.1. Motivation: Surrogate Primal Problem

To motivate our approach, consider a *fluid* approximation of the offline optimum, where we replace random arrivals and service availabilities with their expectations—or equivalently, relax the capacity and backlog constraints to hold in expectation, replace rewards with their expected values, and assume that a r_i fraction of matched cases is deterministically processed at each affiliate i in every period. Under this approximation, the stationary (fractional) solution $z_{t,i}^* = \rho_i$ is feasible and yields zero backlog. This observation motivates our investigation of a sample-path-based *surrogate primal* program that *disregards* the backlog penalty. Our goal is to design an algorithm that competes with this benchmark in net reward while maintaining bounded average backlog.

DEFINITION 5 **(Surrogate Primal Problem).** *The surrgoate primal problem is defined by*

$$\begin{aligned}\mathsf{Primal}^{\mathsf{S}}(\alpha) := \max_{\mathbf{z_t}\in\mathcal{Z}(i_t^\dagger)} & \sum_{t=1}^{T}\sum_{i=1}^{m} w_{t,i}z_{t,i} - \alpha\sum_{i=1}^{m}\left(\sum_{t=1}^{T} z_{t,i} - c_i\right)_+ \\ s.t. \quad & \sum_{t=1}^{T}\mathbb{1}[i_t^\dagger = 0]z_{t,i} \le \left(c_i - \sum_{t=1}^{T}\mathbb{1}[i_t^\dagger = i]z_{t,i}\right)_+ \quad \forall i\in[m]\end{aligned}$$

Similar to the surrogate dual problem (Definition 4), the surrogate primal also addresses the time-varying nature of optimal dual variables for dynamic resources in the dual of the optimal offline (see Proposition 1). This time, it does so by disregarding the backlog penalty in the objective function. Consequently, a learning-based algorithm for the surrogate primal ignores dual variables associated with dynamic resources and focuses solely on maintaining the dual variables for static resources.

This simple idea—ignoring backlogs and relying on surrogate primal problems—faces important technical obstacles. Prior literature (e.g., Agrawal and Devanur (2014)) has explored online-learning-based approaches for solving problems similar to our surrogate primal. However, despite the fluid approximation having no backlog, it is unclear whether a learning-based algorithm can effectively control backlog under random arrivals and service availability. Controlling backlog requires the algorithm-induced arrival rates (a function of matching decisions of the algorithm) to have sufficiently strong convergence to the stationary solution $\boldsymbol{\rho}$. In particular, the convergence must be (i) fast enough, (ii) almost uniform over time, and (iii) hold with high probability (we will formalize these requirements later). Achieving such strong convergence is not immediately obvious, but we show that it is possible using a *proper time-dependent learning rate*, as discussed next.

4.2. Algorithm and Analysis

We now introduce `CO-DL`, our congestion-oblivious dual-learning algorithm, formally described in Section EC.8.1 (Algorithm 2). The algorithm's structure closely follows that of `CA-DL`. For brevity, we only highlight the main differences compared to `CA-DL` in the following.

Primal Phase: `CO-DL` matches the arriving case t to the affiliate i_t that maximizes the new adjusted score, that is, $\mathbf{z}_t \in \arg\max_{\mathbf{z}\in\Delta_m}(\mathbf{w}_t - \boldsymbol{\theta}_t - \boldsymbol{\lambda}_t)\cdot\mathbf{z}_t$, which no longer depends on backlog.

Dual Phase: Similar to `CA-DL`, updates the dual variables for static resources using the multiplicative weight method (Freund et al. 1997). The key difference is that `CO-DL` uses a *time-varying* step size $\{\eta_t\}_{t=1}^T$ to achieve the strong convergence requirements discussed earlier (and further detailed in Section 4.2.1). Specifically, the update rules (7) from Algorithm 1 are now replaced by:

$$\theta_{t+1,i} = \min\{\theta_{t,i}\exp(\eta_t(z_{t,i}-\rho_i)),\alpha\}, \quad \lambda_{t+1,i} = \min\left\{\lambda_{t,i}\exp(\eta_t(z_{t,i}-\rho_i)), \frac{1+2\alpha}{\underline{\rho}}\right\} \tag{21}$$

where $\eta_t = k/\sqrt{t}$ is the time-varying learning rate with an input parameter $k>0$.

We now present our main result for `CO-DL` in Theorems 2. This result relies on the following regularity assumption on the reward distribution, whose implications are discussed later in our analysis.

ASSUMPTION 1 **(Lipschitz Continuous Reward Distribution)**. *The PDF of the reward vector is L-Lipschitz continuous with respect to ℓ_1-norm for some constant $L > 0$.*

THEOREM 2 **(Regret of CO-DL Under Stable Regime)**. *Let $\eta_t = k/\sqrt{t}$ with a constant $k > 0$. Under Assumption 1 and stable regime (Definition 1), the regret of CO-DL is $\mathcal{O}(\sqrt{T} + \frac{\gamma}{\epsilon})$.*

Theorem 2 shows that, under the stable regime, CO-DL matches the performance of CA-DL, achieving the same order of regret despite being oblivious to backlog information. However, the following theorem establishes that CO-DL cannot achieve the same regret performance as CA-DL in the near-critical regime.

THEOREM 3 **(Lower bound on Regret of CO-DL under Near-critical Regime)**. *There exists a constant $q > 0$ and an instance such that for any given $\epsilon \le \frac{q}{\sqrt{T}}$, the regret of CO-DL is $\Omega(\gamma\sqrt{T})$.*

We prove Theorem 3 by constructing a simple single-affiliate instance similar to Example 1 (will be discussed shortly). The detailed proof, along with numerical illustrations, is presented in Section EC.9.

REMARK 3 (**Benefit of Backlog Information**). Together with Corollary 1 (see Section 3.3), which states that CA-DL achieves a regret of $\mathcal{O}(\sqrt{\gamma T})$ in the near-critical regime, Theorem 3 implies that CA-DL outperforms CO-DL in terms of regret by at least a factor of $\sqrt{\gamma}$. In particular, CO-DL fails to provide a sub-linear regret if $\gamma = \Theta(T^\delta)$ with $\delta \in [1/2, 1)$. Intuitively, this difference arises because CA-DL can adapt its matching decisions by explicitly penalizing the adjusted scores via (scaled version of) backlogs, underscoring the inherent advantage of explicitly accounting for congestion.

We now turn our focus to the proof of Theorem 2. The key step is to establish that CO-DL achieves a constant average backlog in expectation under the stable regime, even though *completely* ignores backlog information. As the main building block, we show that a sufficient condition for this result is for the dual variables to converge to their optimal values *in the last iteration (i.e., in every iteration) with high probability*, which we formalize and verify for CO-DL in Section 4.2.1. The following example helps in building an intuition about why this strong convergence guarantee is indeed useful.

EXAMPLE 1. Consider an instance with $T = 5000$, a single affiliate, and no tied cases. For notational convenience, we omit the subscript $i = 1$. The rewards are uniformly distributed on $(0, 1)$, the capacity ratio is $\rho = 0.5$, and the service slack is $\epsilon = 0.1$. For this instance, CO-DLs matching decision reduces to $z_t = \mathbb{1}[w_t \ge \phi_t]$, where $\phi_t \triangleq \theta_t + \lambda_t$. The optimal dual variable $\hat{\phi}$ of the surrogate primal is the median of $(w_1, .., w_T)$, [27] which is "highly concentrated" around 0.5 (the median of reward distribution). With 10,000 sample paths, the 5% quantile of $\hat{\phi}$ is 0.5 ± 0.01. Thus, we let $\hat{\phi} = 0.5$ for ease of exposition.

With post-allocation service and stochastic service time, we can interpret the above example as a simple queueing system where the arrival rate at each period is time-varying and determined *endogenously* by CO-DL's matching decisions. Specifically, the conditional arrival rate is given by $\mathbb{E}[z_t|\mathcal{H}_{t-1}] =$

[27] For this arrival sequence, the dual of the surrogate primal problem is $\min_{\phi \ge 0} D(\phi) := \sum_{t=1}^{T} \{(w_t - \phi)_+ + 0.5\phi\}$. Its derivative (when differentiable) is $-\sum_{t=1}^{T} \mathbb{1}[w_t \ge \phi] + 0.5T$, which is negative (positive) if ϕ is below (above) the sample median. Hence, the sample median of $(w_1, .., w_T)$ minimizes this dual function.

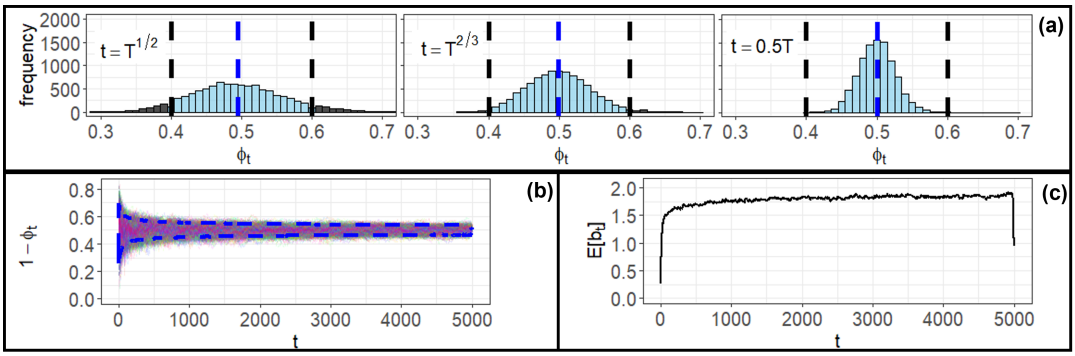


Figure 2 **Numerical illustrations for Example 1 with $\eta_t = 1/\sqrt{t}$, $T = 5000$, and 10,000 sample paths. Panel (a) shows histograms of ϕ_t for three values of t. Panel (b) displays sample paths of the endogenous arrival rates $1 - \phi_t$, with 5% and 95% quantiles (blue dashed lines). Panel (c) depicts the expected backlog for each t.**

$\mathbb{E}[\mathbb{1}[w_t \geq \phi_t]|\mathcal{H}_{t-1}] = 1 - \phi_t$. Suppose for a moment we choose decisions according to $\hat{z}_t = \mathbb{1}[w_t \geq 0.5]$. It is straightforward to show that the resulting average backlog remains constant (in expectation), as its backlog process $\hat{b}_t$ behaves like a non-negative random walk with negative drift at each period t.[28]

Returning to `CO-DL`, although ϕ_t is not exactly equal to 0.5, we have designed its learning process (particularly the time-varying learning rates) so that ϕ_t highly concentrates around 0.5 for sufficiently large t, and gets even *more* concentrated as time progresses. To illustrate this convergence, Figure 2-(a) shows the distribution of ϕ_t at three different periods $t \in \{T^{1/2}, T^{2/3}, 0.5T\}$. The strong concentration around 0.5 implies that the induced arrival rate $\mathbb{E}[z_t|\mathcal{H}_{t-1}] = 1 - \phi_t$ consistently stays close to 0.5, which is well below the service rate $r = 0.6$, with high probability (as illustrated in Figure 2-(b)). This ensures, with high probability, a negative drift in backlog process at almost every period, which is sufficient to show a constant expected average backlog (as confirmed in Figure 2-(c)).

Building on these intuitions, in Section 4.2.1, we first formalize a high probability last-iterate convergence property for the dual variables constructed by `CO-DL`. Then, in Section 4.2.2, we show that this property ensures a negative drift in backlog for `CO-DL` with high probability at nearly every period. This enables us to finish the proof Theorem 2 by showing the constant average backlog of `CO-DL`.

4.2.1. High Probability Last-iterate Convergence of Dual Variables

To establish our desired convergence of the dual variables in `CO-DL`, we first define a *static dual problem*.

Definition 6 **(Static Dual Problem & Dual-based Decision)**. *The static dual problem and its optimal solution $\boldsymbol{\nu}^*$ are defined as follows.*

$$D(\boldsymbol{\nu}) := \mathbb{E}\Big[\max_{\mathbf{z} \in \mathcal{Z}(i^\dagger)} (\mathbf{w} - \boldsymbol{\theta} - \boldsymbol{\lambda}) \cdot \mathbf{z} + \boldsymbol{\rho} \cdot \boldsymbol{\theta} + \boldsymbol{\rho} \cdot \boldsymbol{\lambda}\Big], \quad D(\boldsymbol{\nu}^*) := \min_{\boldsymbol{\nu} \in \mathcal{V}} D(\boldsymbol{\nu})$$

[28] More precisely, using the drift lemma (Lemma 3), we have, for any t, that $\mathbb{E}[\psi(\hat{b}_t) - \psi(\hat{b}_{t-1})|\mathcal{H}_{t-1}] \leq b_{t-1}\mathbb{E}[\hat{z}_t - s_t|\mathcal{H}_{t-1}] + \mathcal{O}(1) \leq -0.1 b_{t-1} + \mathcal{O}(1)$. Summing this inequality over all periods $t \in [T]$ and taking the expectations, we deduce that the average backlog is $\mathcal{O}(1)$ in expectation.

where $\boldsymbol{\nu} := (\boldsymbol{\theta}, \boldsymbol{\lambda})$, $\mathcal{V} := [0, \alpha]^m \times [0, \frac{1+2\alpha}{\underline{\rho}}]^m$, *and the expectation is taken over the stochastic arrival* $\mathbf{A} = (\mathbf{w}, i^\dagger) \sim \mathcal{F}$. *Further, for a given dual variable* $\boldsymbol{\nu} \in \mathcal{V}$ *and arrival type* $\mathbf{A} \in \mathcal{A}$, *we define its corresponding (primal) decision as*

$$\tilde{\mathbf{z}}(\boldsymbol{\nu}, \mathbf{A}) := \arg\max_{\mathbf{z} \in \mathcal{Z}(i^\dagger)} \{(\mathbf{w} - \boldsymbol{\theta} - \boldsymbol{\lambda}) \cdot \mathbf{z}\}.$$

The following proposition (formally proven in detail in Section EC.8.3) establishes the desired high probability last-iterate convergence of dual variables $\boldsymbol{\nu}_t$ constructed by `CO-DL` to $\boldsymbol{\nu}^* \in \arg\min_{\boldsymbol{\nu} \in \mathcal{V}} D(\boldsymbol{\nu})$.

Proposition 3 **(High-probability Last Iterate Convergence of Dual Variables)**. *Let* T_A *be the stopping time of* `CO-DL`.[29] *For any fixed* $t \le T_A$ *and* $\delta \in (0, 1)$, *we have*

$$\mathbb{P}\left[D(\boldsymbol{\nu}_t) - D(\boldsymbol{\nu}^*) \le \mathcal{O}\Big(\frac{\log(t)\log(1/\delta)}{\sqrt{t}}\Big)\right] \ge 1 - \delta.$$

We sketch the proof of Proposition 3, which relies on the connection between the dual update rule of `CO-DL` in (21) and the online stochastic mirror descent (OSMD) (Nemirovski et al. 2009) for solving the static dual problem (Definition 6). Specifically, the gradient of the static dual function $D(\cdot)$ is $\nabla D(\boldsymbol{\nu}) = (\mathbb{E}_{\mathbf{A}}[\boldsymbol{\rho} - \tilde{\mathbf{z}}(\boldsymbol{\nu}, \mathbf{A})], \mathbb{E}_{\mathbf{A}}[\boldsymbol{\rho} - \tilde{\mathbf{z}}(\boldsymbol{\nu}, \mathbf{A})])$. Therefore, up to $t \le T_A$, the dual update in (21) is a mirror descent update, with negative entropy as the mirror map $h(\boldsymbol{\nu}) = \sum_{i=1}^m \theta_i \log(\theta_i) + \sum_{i=1}^m \lambda_i \log(\lambda_i)$:

$$\boldsymbol{\nu}_{t+1} = \arg\min_{\boldsymbol{\nu} \in \mathcal{V}} \ \hat{\mathbf{g}}_t \cdot \boldsymbol{\nu} + \frac{1}{\eta_t} V_h(\boldsymbol{\nu}, \boldsymbol{\nu}_t), \tag{22}$$

where $V_h(\cdot, \cdot)$ is the Bregman distance with respect to h (Bubeck 2015). Here, the stochastic gradient is $\hat{\mathbf{g}}_t = (\boldsymbol{\rho} - \tilde{\mathbf{z}}(\boldsymbol{\nu}_t, \mathbf{A}_t), \boldsymbol{\rho} - \tilde{\mathbf{z}}(\boldsymbol{\nu}_t, \mathbf{A}_t))$, which is an unbiased estimator for the true gradient $\nabla D(\boldsymbol{\nu}_t)$ for all $t \le T_A$. This is because $\mathbb{E}[\hat{\mathbf{g}}_t | \mathcal{H}_{t-1}] = \nabla D(\boldsymbol{\nu}_t)$ for all $t \le T_A$, as (i) each arrival $\mathbf{A}_t$ is an i.i.d. sample from the distribution $\mathcal{F}$ and (ii) $\boldsymbol{\nu}_t$ is $\mathcal{H}_{t-1}$-measurable. Given this connection, the key step in our proof is establishing the last-iterate high probability convergence for our variant of OSMD. For this, we closely follow the approach of Harvey et al. (2019), who proved an analogous result for stochastic gradient descent—a special case of OSMD—and adapt their proof to our variant of OSMD.

Remark 4 (**Fixed versus Time-varying Learning Rates**). So far, we have established the convergence of $\boldsymbol{\nu}_t$ at *any* t (up to the stopping time), using the time-varying learning rate $\eta_t = \Theta(1/\sqrt{t})$. In Section 4.2.2, we leverage this uniform-in-time convergence to prove a constant average backlog for `CO-DL` via drift analysis. At a high level, this drift analysis involves comparing the endogenous arrival rate—determined by $\boldsymbol{\nu}_t$—to the service rate $\rho_i + \epsilon$ at each period t, thus requiring consistent concentration of $\boldsymbol{\nu}_t$ throughout (nearly) the entire horizon. Such uniform per-period drift control is standard in queueing theory, where stability typically follows from uniform-in-time drift bounds. By contrast, using a fixed learning rate (e.g., $\eta = \Theta(1/\sqrt{T})$) only guarantees convergence at the *final* time T, which is insufficient for our analysis. Hence, the time-varying step size is crucial for our results.

[29] The stopping T_A for `CO-DL` is identically defined as the one for `CA-DL` given in equation (16).

4.2.2. Proof Sketch of Theorem 2

With all the ingredients discussed in Section 4.2.1, we are now ready to sketch the proof of Theorem 2. As highlighted earlier, the main part is the following lemma, which establishes an upper bound on the expected average backlog in the stable regime via drift analysis (formally proven in Section EC.8.4).

LEMMA 6 **(Bounding Average Backlog of `CO-DL` Under Stable Regime).** *Under Assumption 1 and stable regime (Definition 1), for any arrival distribution $\mathcal{F}$ and service slack $\epsilon > 0$, we have* $\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T}\|\mathbf{b}_t\|_1\right] \le \mathcal{O}\left(\frac{1}{\epsilon}\right)$.

With the above lemma, the remaining step is to bound the loss of the net matching reward (Lemma EC.42, Section EC.8.4.1), similar to Lemma 1 for `CA-DL`. We defer proof details of Theorem 2 to Section EC.8.4, and dedicate the remainder of this section to outlining the proof of Lemma 6.

The crux of the proof is establishing an $\mathcal{O}(1/\epsilon)$ bound on the average backlog up to the stopping time. As alluded to earlier, it suffices to show that the backlog has a negative drift with high probability in (almost) every period before the stopping time. To that end, define the following "good event" $\mathcal{B}_T$:

$$\mathcal{B}_T := \left\{\mathbb{E}_{\mathbf{A}}[\tilde{z}_i(\boldsymbol{\nu}_t, \mathbf{A})] - (\rho_i + \epsilon) \le -\frac{\epsilon}{2} \text{ for all } \sqrt{T} \le t \le T_A \text{ and } i \in [m]\right\}. \tag{23}$$

First, let us unpack why the occurrence of the good event $\mathcal{B}_T$ is sufficient to establish a constant average backlog. Recall that the endogenous arrival rate at period $t \le T_A$ is given by $\mathbb{E}[\mathbf{z}_t|\mathcal{H}_{t-1}] = \mathbb{E}_{\mathbf{A}}[\tilde{\mathbf{z}}(\boldsymbol{\nu}_t, \mathbf{A})]$, while the service rate for each affiliate i is given by $\rho_i + \epsilon$. Therefore, $\mathbb{E}_{\mathbf{A}}[\tilde{z}_i(\boldsymbol{\nu}_t, \mathbf{A})] - (\rho_i + \epsilon)$ roughly serves as the drift of the backlog process for each affiliate i, which is always negative under the good event. In the proof of Lemma EC.44 (a required technical lemma, Section EC.8.4.3), we use a Lyapunov argument with the quadratic potential function ψ (equation (13)) to formally show that such negative drift leads to an $\mathcal{O}(1/\epsilon)$ bound on the average backlog up to the stopping time.

The remainder of the proof of Lemma 6 involves establishing that the good event occurs with high probability, as formalized in Lemma EC.43. The proof is fairly intricate and here we only provide a high-level overview. Our proof crucially relies on Assumption 1 and the convergence result of Proposition 3. Under Assumption 1, we can translate the high probability convergence of $D(\boldsymbol{\nu}_t)$ to $D(\boldsymbol{\nu}^*)$ (Proposition 3) into convergence of $\mathbb{E}_{\mathbf{A}}[\tilde{\mathbf{z}}(\boldsymbol{\nu}_t, \mathbf{A})]$ to $\mathbb{E}_{\mathbf{A}}[\tilde{\mathbf{z}}(\boldsymbol{\nu}^*, \mathbf{A})]$. Furthermore, as shown in Section EC.8.2, the optimal solution $\boldsymbol{\nu}^*$ of the static dual problem (Definition 6) satisfies $\mathbb{E}_{\mathbf{A}}[\tilde{\mathbf{z}}(\boldsymbol{\nu}^*, \mathbf{A})] \le \boldsymbol{\rho}$ (with equality whenever $\theta_i^* > 0$). Hence, the endogenous arrival rates $\mathbb{E}_{\mathbf{A}}[\tilde{\mathbf{z}}(\boldsymbol{\nu}_t, \mathbf{A})]$ concentrate around a value at most $\boldsymbol{\rho}$ for (almost) every period $t \le T_A$, ensuring the good event occurs with high probability.

5. Case Study on Refugee Resettlement Data

In this section, we numerically evaluate our learning-based approach using data from our partner agency. We first describe the construction of primitives and minor algorithmic adaptations made to meet practical requirements. We then present numerical results showing that our algorithm outperforms both the current practice and other algorithms proposed in the refugee matching literature.

5.1. Data and Benchmarks

Data and Primitives. We use data on working-age refugees (18 to 64 years old) resettled by our partner agency during 2014–2016. In 2015, our partner resettled $T = 3819$ cases across $m = 45$ affiliates. We set the affiliate capacities to the actual number of cases resettled in each affiliate (institutional restrictions prevent us from accessing exact knowledge of initial capacities used by our partner). We use the *actual sequence* of refugee arrivals to construct a sequence of type arrivals $\mathbf{A}_t = (\mathbf{w}_t, i_t^\dagger)$. For the reward vector $\mathbf{w}_t$, we follow the methodology of Bansak et al. (2018), using a supervised machine learning model to predict employment probabilities for each case–affiliate pair based on demographic characteristics. Specifically, our partner has trained a gradient boosting model on historical data from prior years, to estimate these probabilities for 2015 arrivals.[30] Our data also includes indicators for cases with U.S. ties, which must be matched to predetermined affiliates. For these tied cases, we set the target affiliate to be the one where the case was actually placed. The remaining cases are considered free and can be matched to any affiliate. To measure backlog, following the approach of Bansak and Paulson (2024), we adopt a deterministic service flow with service rates equal to $\boldsymbol{\rho}$. This particular backlog measure is a metric currently used by our partner agency to assess congestion. We apply the same procedure to construct primitives for 2016 ($T = 4980$). For both 2015 and 2016, roughly 70% of arrivals were tied cases. Data from 2014 is reserved for parameter tuning, as will be discussed shortly.

Algorithms under Evaluation. We compare the performance of the following policies:

(i) Robust Online Learning-based Algorithm (`RO-Learning`). This policy is an adaptation of our `CA-DL` (Algorithm 1), modified to remove the dummy affiliate entirely. Specifically, each arriving free case is matched to an actual affiliate i with the highest adjusted score (possibly negative) among those with remaining capacity, and dual variables are updated according to line (7) of Algorithm 1. Under our primitives, since $\sum_{i=1}^{m} \rho_i = 1$, there is always at least one affiliate with remaining capacity at each period $t \in [T]$. This adaptation mirrors real-world heuristics, as each case must be matched to an actual affiliate (see also Remark 2 on incorporating this adaptation into our theoretical analysis). `CA-DL` relies on two learning rates, η and ζ. We tune them in a data-driven manner to achieve reasonable ranges of over-allocation and average backlog, using 2014 data. Further details are provided in Section EC.11.2.

(ii) Congestion-Oblivious Learning-based Algorithm (`CO-DL`). This policy is a minor modification of `CO-DL` (Sections 4 and EC.11.2, Algorithm 2), similarly ensuring that cases are matched only to actual affiliates. The step size η_t in line (21) is tuned using the same data-driven approach for `RO-Learning`.

(iii) Sampling-based Algorithm (`Sampling`). As motivated in Section 1, the existing proposals in the refugee matching literature are based on a *sampling* approach, which simulates future arrivals using past years' data. We use (an adaptation of) the algorithm from Bansak and Paulson (2024) as our

[30] We emphasize that historical data from prior years is used solely to estimate employment probabilities, not to predict future arrival patterns. As previously discussed (e.g., Figure 1) and further illustrated later (e.g., Figure EC.2a), predicting future arrivals is considerably more challenging since the composition of arrivals varies from year to year.

Table 1 Results for year 2015 (2016, resp.) for penalty parameters $\alpha = 3$ and $\gamma = 5$. The total number of cases is $T = 3819$ (4980, resp.) for year 2015 (2016, resp.). For `Sampling`, we take an average over five simulations.

	Employment Rate (%)	Total Over-allocation	Average Backlog
`Actual`	37.3 (37.7)	0 (0)	226.6 (323.5)
`Sampling`	45.0 (46.3)	107.6 (99.8)	180.1 (225.2)
`RO-Learning`	44.6 (46.0)	71 (81)	151 (199)
`CO-DL`	44.4 (46.3)	123 (84)	202.5 (251.0)

main benchmark based on this approach. Their original algorithm is developed for the case with no tied cases. See Sections 1.3 and EC.11.3, detailing how we modify this algorithm to handle tied cases.

`Sampling` was also designed to maximize the same objective (Objective) as ours, with the current congestion level as a penalty term scaled with γ. However, a crucial difference is that `Sampling` relies on sample trajectories of future arrivals: to match case t, `Sampling` solves an offline problem for periods t and onward by sampling future arrivals from a given sampling pool. Following Bansak and Paulson (2024), we use arrivals from *the previous year* as this sampling pool. For example, when tested on the 2015 data, `Sampling` uses 2014 arrivals to simulate future cases. Thus, the performance of `Sampling` intuitively depends on how representative the sampling pool is for the current arrival sequence.

(iv) Actual Placements (`Actual`). Under current practice, case officers at our partner agency manually determine initial case placements. As highlighted in Section 1, the agency provides essential services (such as case management) to refugees after placement. Beyond respecting annual quotas, preventing congestion in these services is a critical consideration for the agency. Hence, current placements have been primarily driven by workload balancing and capacity constraints, without systematically accounting for employment outcomes (Bansak and Paulson 2024). We use actual case officer decisions as a benchmark to assess how our learning-based approach can improve the agency's current practice.

5.2. Simulation Results

Main Result: Table 1 reports results for penalty parameters $\alpha = 3$ and $\gamma = 5$. The first column shows the employment rate, measured as the percentage of total employment outcomes (including tied cases) relative to total arrivals T. The second and third columns show the total over-allocation (summed across all affiliates) and the average backlog, respectively. Since affiliate capacities were set based on actual placements, `Actual` incurs no over-allocation by construction. Nevertheless, `RO-Learning` significantly improves upon `Actual` in both employment rates and the average backlog—for example, by roughly 20% and 33%, respectively, in 2015. We also highlight that the employment rate in Table 1 includes both tied and free cases. If we exclude tied cases (for which all algorithms make the same decision), `RO-Learning` improves employment rate of free cases by roughly 50% compared to `Actual`.[31]

[31] That said, we remark that `Actual` may have been influenced by certain unobserved constraints not accounted for in this case study (e.g., unobserved incompatibility between certain free cases and affiliates). Therefore, the improvement over `Actual` should not be interpreted at face value but rather as an indication of the potential benefits that `RO-Learning` could offer when implemented in real-world scenarios.

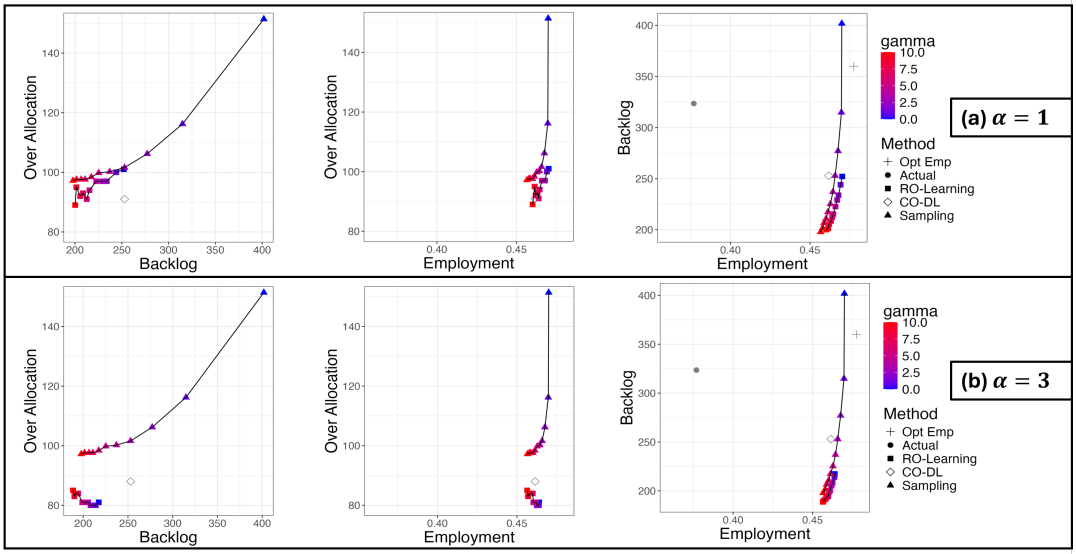


Figure 3 **Numerical results for year 2016 with $\alpha \in \{1,3\}$ and $\gamma \in \{0,\ldots,10\}$.**

Moving beyond the comparison to actual placements, we now evaluate `RO-Learning` against the more sophisticated `Sampling` benchmark. For $\alpha = 3$ and $\gamma = 5$, `RO-Learning` improves the combined objective (employment, backlog, and over-allocation; see (Objective)) by roughly 48% for 2015 (20% for 2016). To understand this improvement further, we compare each of the three outcomes separately in Table 1. `RO-Learning` significantly reduces both over-allocation and average backlog in both years, with minimal impact (less than 1%) on employment outcomes. For instance, in 2015, `RO-Learning` achieves roughly 34% lower total over-allocation and 16% lower average backlog compared to `Sampling`. Lastly, although `CO-DL` does not use backlog information, it still substantially improves employment outcomes (by 19.0%22.8%) and average backlog (by 10.6%22.4%) relative to `Actual`. However, it underperforms `RO-Learning`, incurring higher over-allocation and average backlog.[32] Thus, our primary recommendation to the partner agency is `RO-Learning` (an adaptation of `CA-DL`).

To further understand how penalty parameters impact the three outcomes under different algorithms, we simulate over a range of values $\alpha \in \{1,2,..,5\}$ and $\gamma \in \{0,1,...,10\}$. In Figure 3, we present trajectories of the three outcomes for 2016 with $\alpha = 1$ and 3 while varying γ, deferring a more comprehensive analysis to Section EC.11.5. Each subfigure includes three panels, each comparing two of the three key outcomes. Results are shown for all of the benchmarks introduced in Section 5.1. We also include `Opt Emp`, which corresponds to the surrogate primal problem in Definition 5 and represents the sample-path-based maximum employment outcome attainable by any (offline or online) algorithm. Note that, by construction of our arrival sequence, `Opt Emp` incurs no over-allocation.

[32] In Section EC.11.4, we further show that `RO-Learning` consistently outperforms `CO-DL` over a wide range of penalty parameters, although this performance gap narrows as the congestion penalty parameter γ decreases.

Focusing on the right panels, we first observe that the employment outcomes for both `RO-Learning` and `Sampling` are close to `Opt Emp`, indicating that the accounting for backlogs does not necessarily compromise employment outcomes significantly. However, increasing the backlog penalty parameter γ slightly reduces the employment rate. Additionally, compared to `Actual`, both `RO-Learning` and `Sampling` achieve substantially higher employment rates while maintaining lower average backlogs.

Comparing `RO-Learning` and `Sampling`, we observe that the trajectory of the outcome metrics for `RO-Learning` generally dominates that of `Sampling`. For example, given a specific employment outcome, `RO-Learning` consistently achieves a lower over-allocation and average backlog. Similar trends are evident in the tradeoff between over-allocation and backlog. Furthermore, over-allocation for `RO-Learning` tends to decrease as α increases. Taken together, these improvements lead to a consistently better overall objective value (Objective) for `RO-Learning` over `Sampling`, with a median improvement of 37% in 2015 and 17% in 2016, across parameters $\alpha \in \{1, 2, \ldots, 5\}$ and $\gamma \in \{0, 1, \ldots, 10\}$.

Performance Improvement through Robustness: To understand the primary source of performance gains, we highlight the *robustness* inherent in our learning-based approach. `Sampling` heavily relies on a sampling pool of arrivals from the previous year, making it vulnerable to discrepancies between this pool and actual current-year arrivals (as evidenced in Figure 1 in Section 1.1). In contrast, `RO-Learning` avoids this pitfall by adaptively learning arrival patterns directly from current-year data.

Practical Benefits of `RO-Learning`: We highlight additional practical advantages of adversarial online learning employed by `RO-Learning`. First, the algorithm is inherently balanced due to its *self-correcting* nature. Specifically, beyond explicitly penalizing backlog, the dual-variable update rule (line (7) in Algorithm 1) naturally prevents bursty matchings to each affiliate, as dual variables corresponding to an affiliate automatically increase when a case is matched to that affiliate, and decrease otherwise. This self-correction tends to align the algorithm's endogenous arrival rate closely with $\boldsymbol{\rho}$. In contrast, `Sampling` would maintain this property *only* if the sampling pool accurately represents the true underlying distribution, which is not the case in our data. Indeed, as shown in the right panels of Figure EC.5, `RO-Learning` maintains a relatively low backlog even when $\gamma = 0$, comparable to `Sampling` at higher values of γ. Second, `RO-Learning` is significantly advantageous over `Sampling` in terms of computational efficiency; Thanks to online learning, unlike `Sampling`, it does not need to solve any (heavy) auxiliary optimization problem in each period—which can be valuable since, in practice, policymakers may need to run multiple simulations for parameter tuning or back-testing on various datasets. Lastly, the simplicity and interpretability of `RO-Learning`'s score-based rule makes it easy to communicate with the practitioners at our partner agency.

5.3. Additional Numerical Results with Practical Considerations

In this subsection, to enhance the practical relevance of our proposed algorithm, we extend our case study to incorporate additional practical considerations relevant to our partner agency.

Table 2 Over-allocation outcomes for affiliates exceeding 110% of their capacity for year 2015 (2016, resp.). We use the same penalty parameters in Table 1 and take an average over five simulations for `Sampling`.

Method	# Flagged Affiliates with >10% Over-allocation	Max Over-allocation above 110% Capacity	Total Over-allocation above 110% Capacity
`Sampling`	2 (3)	15.2 (5.4)	19.8 (11.7)
`RO-Learning`	2 (5)	10.8 (4.10)	12.6 (9.3)

(i) Beyond Aggregate Over-allocation. While resettlement agencies aim to adhere to the annual quotas, they are not enforced as hard constraints: affiliates are typically allowed to exceed their stated capacity by up to 10% without additional approval (Ahani et al. 2021). However, any resettlement exceeding 110% of an affiliate's capacity requires formal approval by the State Department (U.S. Department of State 2011), incurring extra administrative costs for the agency.[33] For this reason, it is important to examine not only total over-allocation (aggregated across all affiliates, as discussed in Section 2), but also how it is distributed—particularly whether, and to what extent, specific affiliates are "flagged" for exceeding the 10% threshold. Table 2 reports the over-allocation for affiliates exceeding 110% of their capacity, using the same penalty parameters as in Table 1. We focus on comparing `RO-Learning` (our main proposal) with `Sampling`. The first column lists the number of flagged affiliates. While `RO-Learning` flags more affiliates than `Sampling` in 2016 (5 vs. 3), the largest over-allocation above threshold among these flagged affiliates is smaller. Furthermore, summing all placements above threshold—each requiring formal reporting—`RO-Learning` yields a lower total. Thus, `RO-Learning` not only incurs less total over-allocation (as previously shown in Table 1), but also reduces the administrative burden associated with exceeding the 10% threshold.

(ii) Varying Case Size. In the U.S., annual quotas are tracked at the individual level, meaning that each refugee case may consume multiple units of capacity depending on family size. Here, we extend our simulations to incorporate varying case sizes. As noted in Section 2.1, our algorithmic framework can accommodate this variation via multiple knapsack constraints (Section EC.12). We build on this generalized model and algorithms in Section EC.12. Specifically, our data includes the number of family members per case. Based on this, we augment each case t with a family size n_t. We retain the same case-level employment probability vector $\mathbf{w}_t$ as described in Section 5.1.[34] Affiliate capacity c_i is set to the actual number of individuals placed at affiliate i, with each case consuming n_t units. Table 3 reports the results under the same penalty parameters as in Table 1. `RO-Learning` continues to substantially improve employment outcomes and reduce average backlog compared to the

[33] That said, over-allocation within the 10% threshold is still costly, as the annual quota is directly linked to the per-capita funding provided by the government (See Footnote 13). Thus, exceeding quotas—even within the permitted range—still places a burden on affiliate resources.

[34] Our partner agency's in-house machine learning model predicts employment probabilities for each individual family member of each refugee case t at affiliate i. Following the approach of Bansak et al. (2018), we calculate the case-level employment probability $w_{t,i}$ as the probability that at least one family member secures employment at affiliate i, assuming independence across family members employment outcomes.

Table 3 **Results with varying case size for 2015 (2016, resp.) under the same penalty parameter in Table 1.**

	Employment Rate (%)	Total Over- allocation	Average Backlog
`Actual`	37.3 (37.9)	0 (0)	641.2 (996.8)
`Sampling`	45.7 (47.5)	308.4 (366.2)	526.9 (792.6)
`RO-Learning`	44.3 (45.2)	239 (194)	414.3 (615.5)
`CO-DL`	45.5 (45.2)	284 (314)	542.9 (852.7)

current practice (`Actual`). Relative to `Sampling`, it reduces over-allocation by 22–47% and backlog by over 20%, with only a small loss in employment (within 5%). Overall, these findings show that our proposed algorithms remain effective even when accounting for varying case sizes.

(iii) Mid-Year Disruptions and Capacity Revisions. While our main numerical study uses data from 2014–2016, recent years have seen significant disruptions in U.S. refugee resettlement policy. Typically, the federal government sets an admissions ceiling at the start of each fiscal year, based on which the agencies plan their total number of arrivals T and annual quotas. However, this ceiling—and accordingly T and capacities—can change mid-year due to policy shifts. Recently, such disruptions have also been accompanied by major shifts in the *composition* of arrivals. For example, in 2017, an executive order reduced the admissions ceiling and banned free-case admissions from several countries (Howe 2017, Asian Americans Advancing Justice 2019). To assess robustness under such disruptions, in Section EC.13, we provide simulation results based on a synthetic dataset that mimics the 2017 mid-year disruption. Our results indicate that our proposal continues to outperform other benchmarks.

6. Conclusion and Future Directions

Motivated by our collaboration with a major U.S. refugee resettlement agency, we introduced a new dynamic matching problem capturing key practical considerations: resettling refugees consumes static resources (e.g., annual quotas) and requires dynamic, post-matching services (e.g., translation). The agency must therefore balance matching rewards against congestion costs of dynamic resources, while respecting constraints on static resources. Drawing insights from our partner's data, we developed prior-free, learning-based algorithms that do not rely on historical arrival data and provide provable performance guarantees. As demonstrated in Section 5, our robust approach significantly improves outcomes and offers several additional practical advantages.

From a theoretical perspective, while our framework shares features with models studied in online (static) resource allocation and matching queues, it differs fundamentally from both. As a result, designing and analyzing low-regret online algorithms required developing novel technical ideas, potentially of independent interest in other problems. Leveraging these new techniques, we established sub-linear regret guarantees against a strong benchmark whenever possible (i.e., when $\gamma = o(T)$). Exploring weaker benchmarks is an interesting theoretical direction for future research.

Beyond Refugee Resettlement. Our framework can be applied to general dynamic resource allocation problems sharing two structural features: (i) allocation of long-term resources with fixed

capacities; and (ii) the need for short-term, local onboarding services as an essential component of such allocation. For example, in hospital admission control, incoming patients must be assigned to one of the specialized units (e.g., cardiac medicine or general surgery)—each with a fixed number of beds that serve as long-term resources—or be rejected. Each admitted patient then requires local onboarding services, such as initial testing upon arrival to the unit. In these contexts, undesirable backlog may arise due to a shortage of dynamic resources. (In the context of emergency departments, see Shi et al. (2016), which discusses post-bed-allocation delay caused by factors such as nurse shortages.)

A similar structure arises in adult foster care: when individuals in need of long-term care are discharged from hospitals, discharge coordinators must assign them to adult foster care facilities, each with a limited intake capacity (Bartle et al. 2025). Each assignment initiates local onboarding processes, such as an initial assessment at the assigned facility. Our framework can help managers balance the reward from consuming static resources against the congestion of dynamic resources. Furthermore, in the aforementioned applications, historical data from prior planning horizons may be unreliable. For example, patient flows in hospitals can fluctuate due to seasonal illnesses or emergent events. Our prior-free approach offers robust and computationally efficient decision support by not relying on solving auxiliary optimization problems.

Acknowledgment

Vahideh Manshadi and Rad Niazadeh gratefully acknowledge the Simons Institute for the Theory of Computing, as this work was done in part while attending the program on Data Driven Decision Processes. The authors thank Global Refuge for access to data and guidance. The data used in this study were provided under a collaboration research agreement with Global Refuge, which requires that these data not be transferred or disclosed. Kirk Bansak and Elisabeth Paulson are Faculty Affiliates of the Immigration Policy Lab (IPL) at Stanford University and ETH Zurich. This work is associated with IPL's GeoMatch project, for which the authors acknowledge funding from the Charles Koch Foundation, Stanford Institute for Human-Centered Artificial Intelligence, Google.org, and Stanford Impact Labs. The authors extend their gratitude to the entire GeoMatch team for helpful feedback. Soonbong Lee and Vahideh Manshadi also extend their gratitude to the Management Science and Engineering Department at Stanford University for hosting them during part of this research. Rad Niazadehs research is partially supported by an Asness Junior Faculty Fellowship at Chicago Booth School of Business.

References

Philipp Afeche, Rene Caldentey, and Varun Gupta. On the optimal design of a bipartite matching queueing system. *Operations Research*, 70(1):363–401, 2022.

Shipra Agrawal and Nikhil R Devanur. Fast algorithms for online stochastic convex programming. In *Proceedings of the twenty-sixth annual ACM-SIAM symposium on Discrete algorithms*, pages 1405–1424. SIAM, 2014.

Shipra Agrawal, Zizhuo Wang, and Yinyu Ye. A dynamic near-optimal algorithm for online linear programming. *Operations Research*, 62(4):876–890, 2014.

Narges Ahani, Tommy Andersson, Alessandro Martinello, Alexander Teytelboym, and Andrew C Trapp. Placement optimization in refugee resettlement. *Operations Research*, 69(5):1468–1486, 2021.

Narges Ahani, Paul Gölz, Ariel D Procaccia, Alexander Teytelboym, and Andrew C Trapp. Dynamic placement in refugee resettlement. *Operations Research*, 2023.

Ross Anderson, Itai Ashlagi, David Gamarnik, and Yash Kanoria. Efficient dynamic barter exchange. *Operations Research*, 65(6):1446–1459, 2017.

Ali Aouad and Ömer Saritaç. Dynamic stochastic matching under limited time. In *Proceedings of the 21st ACM Conference on Economics and Computation*, pages 789–790, 2020.

Alessandro Arlotto and Itai Gurvich. Uniformly bounded regret in the multisecretary problem. *Stochastic Systems*, 9(3):231–260, 2019.

Itai Ashlagi, Patrick Jaillet, and Vahideh H Manshadi. Kidney exchange in dynamic sparse heterogenous pools. *arXiv preprint arXiv:1301.3509*, 2013.

Itai Ashlagi, Maximilien Burq, Patrick Jaillet, and Vahideh Manshadi. On matching and thickness in heterogeneous dynamic markets. *Operations Research*, 67(4):927–949, 2019.

Itai Ashlagi, Jacob Leshno, Pengyu Qian, and Amin Saberi. Price discovery in waiting lists: A connection to stochastic gradient descent. *Available at SSRN 4192003*, 2022.

Asian Americans Advancing Justice. Issue brief: Muslim and refugee bans, 2019. URL `https://www.advancingjustice-aajc.org/sites/default/files/2019-09/5_1153_AAJC_Immigration_Muslim%26RefugeeBans.pdf`.

Angelos Aveklouris, Levi DeValve, and Amy R Ward. Matching impatient and heterogeneous demand and supply. *arXiv preprint arXiv:2102.02710*, 2021.

Ashwinkumar Badanidiyuru, Robert Kleinberg, and Aleksandrs Slivkins. Bandits with knapsacks. *Journal of the ACM (JACM)*, 65(3):1–55, 2018.

Santiago R Balseiro, Haihao Lu, and Vahab Mirrokni. The best of many worlds: Dual mirror descent for online allocation problems. *Operations Research*, 71(1):101–119, 2023.

Kirk Bansak and Elisabeth Paulson. Outcome-driven dynamic refugee assignment with allocation balancing. *Operations Research*, 2024.

Kirk Bansak, Jeremy Ferwerda, Jens Hainmueller, Andrea Dillon, Dominik Hangartner, Duncan Lawrence, and Jeremy Weinstein. Improving refugee integration through data-driven algorithmic assignment. *Science*, 359(6373):325–329, 2018.

Kirk Bansak, Elisabeth Paulson, and Dominik Rothenhäusler. Learning under random distributional shifts. In *International Conference on Artificial Intelligence and Statistics*, pages 3943–3951. PMLR, 2024.

Vince Bartle, Ashley Shearer, Alexandra Wroe, Nicola Dell, and Nikhil Garg. Faster information for effective long-term discharge: A field study in adult foster care. *Proceedings of the ACM on Human-Computer Interaction*, 9(2):1–29, 2025.

Amir Beck. *First-order methods in optimization*. SIAM, 2017.

Stephen Boyd, Stephen P Boyd, and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

Andorra Bruno. Reception and placement of refugees in the United States. Congressional Research Service (CRS) Report No. R44878, 2017. URL `https://sgp.fas.org/crs/homesec/R44878.pdf`.

Sébastien Bubeck. Introduction to online optimization. *Lecture notes*, 2:1–86, 2011.

Sébastien Bubeck. Convex optimization: Algorithms and complexity. *Foundations and Trends® in Machine Learning*, 8(3-4):231–357, 2015.

George Casella and Roger L Berger. *Statistical inference*. Cengage Learning, 2021.

Yiwei Chen, Retsef Levi, and Cong Shi. Revenue management of reusable resources with advanced reservations. *Production and Operations Management*, 26(5):836–859, 2017.

Alain Comtet and Satya N Majumdar. Precise asymptotics for a random walkers maximum. *Journal of Statistical Mechanics: Theory and Experiment*, 2005(06):P06013, 2005.

David Delacrétaz, Scott Duke Kominers, and Alexander Teytelboym. Matching mechanisms for refugee resettlement. *American Economic Review*, 113(10):2689–2717, 2023.

Steven Delong, Alireza Farhadi, Rad Niazadeh, Balasubramanian Sivan, and Rajan Udwani. Online bipartite matching with reusable resources. *Mathematics of Operations Research*, 49(3):1825–1854, 2024.

Nikhil R Devanur and Thomas P Hayes. The adwords problem: online keyword matching with budgeted bidders under random permutations. In *Proceedings of the 10th ACM conference on Electronic commerce*, pages 71–78, 2009.

Nikhil R Devanur, Kamal Jain, Balasubramanian Sivan, and Christopher A Wilkens. Near optimal online algorithms and fast approximation algorithms for resource allocation problems. In *Proceedings of the 12th ACM conference on Electronic commerce*, pages 29–38, 2011.

Shaddin Dughmi, Jason Hartline, Robert D Kleinberg, and Rad Niazadeh. Bernoulli factories and black-box reductions in mechanism design. *Journal of the ACM (JACM)*, 68(2):1–30, 2021.

Rick Durrett. *Probability: theory and examples*, volume 49. Cambridge university press, 2019.

Adam N Elmachtoub and Paul Grigas. Smart predict, then optimize. *Management Science*, 68(1):9–26, 2022.

Atilla Eryilmaz and Rayadurgam Srikant. Fair resource allocation in wireless networks using queue-length-based scheduling and congestion control. *IEEE/ACM transactions on networking*, 15(6):1333–1344, 2007.

Jon Feldman, Monika Henzinger, Nitish Korula, Vahab S Mirrokni, and Cliff Stein. Online stochastic packing applied to display ad allocation. In *European Symposium on Algorithms*, pages 182–194. Springer, 2010.

Yiding Feng, Rad Niazadeh, and Amin Saberi. Robustness of online inventory balancing algorithm to inventory shocks. *Available at SSRN 3795056*, 2021.

Yiding Feng, Rad Niazadeh, and Amin Saberi. Technical notenear-optimal bayesian online assortment of reusable resources. *Operations Research*, 72(5):1861–1873, 2024. doi: 10.1287/opre.2023.2512.

Daniel Freund and Siddhartha Banerjee. Good prophets know when the end is near. *Available at SSRN 3479189*, 2019.

Daniel Freund, Thodoris Lykouris, Elisabeth Paulson, Bradley Sturt, and Wentao Weng. Group fairness in dynamic refugee assignment, 2024.

Yoav Freund, Robert E Schapire, Yoram Singer, and Manfred K Warmuth. Using and combining predictors that specialize. In *Proceedings of the twenty-ninth annual ACM symposium on Theory of computing*, pages 334–343, 1997.

Global Refuge. How refugee resettlement works. `Global Refuge website`, accessed June 7, 2025, 2024. `https://www.globalrefuge.org/what-we-do/refugee-resettlement/how/`.

Paul Gölz and Ariel D Procaccia. Migration as submodular optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 549–556, 2019.

Xiao-Yue Gong, Vineet Goyal, Garud N Iyengar, David Simchi-Levi, Rajan Udwani, and Shuangyu Wang. Online assortment optimization with reusable resources. *Management Science*, 68(7):4772–4785, 2022.

Vineet Goyal, Garud Iyengar, and Rajan Udwani. Asymptotically optimal competitive ratio for online allocation of reusable resources. *Operations Research*, 2025.

Donald Gross, John F Shortle, James M Thompson, and Carl M Harris. *Fundamentals of queueing theory*, volume 627. John wiley & sons, 2011.

Vincent Guigues, Anatoli Juditsky, and Arkadi Nemirovski. Non-asymptotic confidence bounds for the optimal value of a stochastic program. *Optimization Methods and Software*, 32(5):1033–1058, 2017.

Anupam Gupta and Marco Molinaro. How the experts algorithm can help solve lps online. *Mathematics of Operations Research*, 41(4):1404–1431, 2016.

Varun Gupta. Greedy algorithm for multiway matching with bounded regret. *Operations Research*, 2022.

Nicholas JA Harvey, Christopher Liaw, Yaniv Plan, and Sikander Randhawa. Tight analyses for non-smooth stochastic gradient descent. In *Conference on Learning Theory*, pages 1579–1613. PMLR, 2019.

Elad Hazan et al. Introduction to online convex optimization. *Foundations and Trends® in Optimization*, 2 (3-4):157–325, 2016.

Amy Howe. Dispute over travel ban could return to supreme court (updated), July 2017. URL `https://www.scotusblog.com/2017/07/dispute-travel-ban-return-supreme-court/`. Accessed: 2025-05-25.

Ming Hu and Yun Zhou. Dynamic type matching. *Manufacturing & Service Operations Management*, 24(1): 125–142, 2022.

Zhiyi Huang, Zhihao Gavin Tang, and David Wajc. Online matching: A brief survey. *ACM SIGecom Exchanges*, 22(1):135–158, 2024.

Yash Kanoria and Pengyu Qian. Blind dynamic resource allocation in closed networks via mirror backpressure. *Management Science*, 2023.

Süleyman Kerimov, Itai Ashlagi, and Itai Gurvich. Dynamic matching: Characterizing and achieving constant regret. *Management Science*, 2023.

Thomas Kesselheim, Klaus Radke, Andreas Tonnis, and Berthold Vocking. Primal beats dual on online packing lps in the random-order model. *SIAM Journal on Computing*, 47(5):1939–1964, 2018.

David A Levin and Yuval Peres. *Markov chains and mixing times*, volume 107. American Mathematical Soc., 2017.

Xiaocheng Li and Yinyu Ye. Online linear programming: Dual convergence, new algorithms, and regret bounds. *Operations Research*, 70(5):2948–2966, 2022.

Aleksandr Mikhailovich Lyapunov. The general problem of the stability of motion. *International journal of control*, 55(3):531–534, 1992.

Aranyak Mehta, Amin Saberi, Umesh Vazirani, and Vijay Vazirani. Adwords and generalized online matching. *Journal of the ACM (JACM)*, 54(5):22–es, 2007.

Marco Molinaro and Ramamoorthi Ravi. The geometry of online packing linear programs. *Mathematics of Operations Research*, 39(1):46–59, 2014.

Michael Neely. *Stochastic network optimization with application to communication and queueing systems*. Springer Nature, 2022.

Michael J Neely. Energy optimal control for time-varying wireless networks. *IEEE transactions on Information Theory*, 52(7):2915–2934, 2006.

Michael J Neely, Eytan Modiano, and Chih-Ping Li. Fairness and optimal stochastic control for heterogeneous networks. *IEEE/ACM Transactions On Networking*, 16(2):396–409, 2008.

Arkadi Nemirovski, Anatoli Juditsky, Guanghui Lan, and Alexander Shapiro. Robust stochastic approximation approach to stochastic programming. *SIAM Journal on optimization*, 19(4):1574–1609, 2009.

Hai Nguyen, Thành Nguyen, and Alexander Teytelboym. Stability in matching markets with complex constraints. *Management Science*, 67(12):7438–7454, 2021.

Martin L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, New York, 1994. ISBN 978-0-471-61977-2.

Sebastian Ruder. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*, 2016.

Moshe Shaked and J George Shanthikumar. *Stochastic orders*. Springer, 2007.

Alexander Shapiro, Darinka Dentcheva, and Andrzej Ruszczynski. *Lectures on stochastic programming: modeling and theory*. SIAM, 2021.

Pengyi Shi, Mabel C Chou, Jim G Dai, Ding Ding, and Joe Sim. Models and insights for hospital inpatient operations: Time-dependent ed boarding time. *Management Science*, 62(1):1–28, 2016.

Maurice Sion. On general minimax theorems. 1958.

Alexander L Stolyar. Maxweight scheduling in a generalized switch: State space collapse and workload minimization in heavy traffic. *The Annals of Applied Probability*, 14(1):1–53, 2004.

Swiss Federal Parliament. Répartition des personnes requérantes dasile entre les cantons: Évaluation du contrôle parlementaire de ladministration à lintention de la commission de gestion du conseil des États [distribution of asylum seekers among the cantons: Evaluation of the parliamentary control of the administration for the management commission of the council of states]. Technical report, Swiss Federal Parliament, 2024. URL `https://www.parlament.ch/centers/documents/fr/Bericht%20vom%2021.6.2024%20PVK%20-%20Asylverteilung%20F20250224.pdf`. Report in French. Accessed May 2025.

Kalyan Talluri and Garrett Van Ryzin. An analysis of bid-price controls for network revenue management. *Management science*, 44(11-part-1):1577–1593, 1998.

Leandros Tassiulas and Anthony Ephremides. Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks. In *29th IEEE Conference on Decision and Control*, pages 2130–2132. IEEE, 1990.

UNHCR. Refugee resettlement facts, 2023. URL `https://www.unhcr.org/us/media/refugee-resettlement-facts`.

UNHCR. Refugee population statistics database, 2024. URL `https://www.unhcr.org/refugee-statistics/`.

U.S. Department of State. Fy 2011 reception and placement basic terms of the cooperative agreement. https://2009-2017.state.gov/j/prm/releases/sample/181172.htm, 2011. Accessed June 5, 2025.

U.S. State Department. Fy 2023 notice of funding opportunity for reception and placement program, 2023. URL https://www.state.gov/fy-2023-notice-of-funding-opportunity-for-reception-and-placement-program/.

Alberto Vera and Siddhartha Banerjee. The bayesian prophet: A low-regret framework for online decision making. *ACM SIGMETRICS Performance Evaluation Review*, 47(1):81–82, 2019.

Zizhuo Wang, Shiming Deng, and Yinyu Ye. Close the gaps: A learning-while-doing algorithm for single-product revenue management problems. *Operations Research*, 62(2):318–331, 2014.

Yehua Wei, Jiaming Xu, and Sophie H Yu. Constant regret primal-dual policy for multi-way dynamic matching. *Available at SSRN 4357216*, 2023.

Uri Yechiali. On optimal balking rules and toll charges in the gi/m/1 queuing process. *Operations Research*, 19(2):349–370, 1971.

Mohammad Zhalechian, Esmaeil Keyvanshokooh, Cong Shi, and Mark P Van Oyen. Online resource allocation with personalized learning. *Operations Research*, 70(4):2138–2161, 2022.

Martin Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *Proceedings of the 20th international conference on machine learning (icml-03)*, pages 928–936, 2003.

EC.1. Summary of Notation and Assumptions

Table EC.1 Summary of Main Notation

Symbol	Description
T	Total number of arrivals.
m	Number of affiliates.
Δ_m	m-dimensional standard simplex ($\{\mathbf{z} \in \mathbb{R}_+^m : \sum_{i=1}^m z_i \le 1\}$).
$x_+ := \max\{x, 0\}$	Positive part of scalar $x \in \mathbb{R}$.
$t \in [T]$	Index for arrival period (one arrival per period).
$i \in [m]$	Index for affiliates.
$\mathbf{w}_t \in [0,1]^m$	Reward vector for arrival t, where $w_{t,i}$ indicates employment outcome (estimated probability of employment) at affiliate i.
$i_t^\dagger \in [m] \cup \{0\}$	Target affiliate for arrival t: $i_t^\dagger = 0$ indicates a free case; $i_t^\dagger = i \in [m]$ indicates a tied case for affiliate i.
$\mathbf{A}_t = (\mathbf{w}_t, i_t^\dagger)$	Type of arrival t.
$\mathbf{z}_t \in \mathbb{R}^m$	Matching decision for arrival t; $z_{t,i}$ indicates matching to affiliate i.
$\mathcal{Z}(i_t^\dagger)$	Type feasibility set for arrival t: equals the simplex Δ_m if $i_t^\dagger = 0$ (free case), and the singleton set $\{\mathbf{e}_{i_t^\dagger}\}$ if $i_t^\dagger \in [m]$ (tied case).
$\{\mathbf{z}_1, \dots, \mathbf{z}_T\}$	Matching profile over the horizon. A profile is *feasible* if each $\mathbf{z}_t \in \mathcal{Z}(i_t^\dagger)$ and (Capacity Feasibility-t) holds for all $t \in [T]$.
c_i	Total capacity (annual quota) of affiliate i.
$\rho_i = c_i/T$	Normalized capacity ratio of affiliate i.
$\underline{\rho} = \min_i \rho_i$	Minimum capacity ratio across affiliates.
r_i	Service rate of affiliate i.
$s_{t,i} \sim \mathrm{Ber}(r_i)$	Service availability indicator for affiliate i at time t, modeled as an i.i.d. Bernoulli random variable with success probability r_i.
$b_{t,i}$	Backlog at affiliate i at time t.
ϵ	Service slack parameter: $r_i \ge \rho_i + \epsilon$ (Definition 1).
α	Penalty parameter for over-allocation.
γ	Penalty parameter for average backlog (congestion).
$\mathsf{OPT}(\alpha, \gamma)$	Offline benchmark (Definition 2).
$\mathsf{ALG}^\pi(\alpha, \gamma)$	Objective achieved by algorithm π (equation (Objective)).
Regret_T^π	Regret of algorithm π (Definition 3).
$\boldsymbol{\theta} \in \mathbb{R}_+^m$	Dual variables corresponding to over-allocation cost of static resources (Proposition 1).
$\boldsymbol{\lambda} \in \mathbb{R}_+^m$	Dual variables corresponding to the capacity constraint for static resources in the offline benchmark (Proposition 1).
$\boldsymbol{\beta}_t \in \mathbb{R}_+^m$	Time-varying dual variables associated with backlog dynamics at time t (Proposition 1).
η (resp. ζ)	Learning rates for dual variables for static (resp. dynamic resources) (Algorithm 1).
$\psi(\mathbf{b}_t)$	Lyapunov function for backlog: $\psi(\mathbf{b}_t) = \frac{1}{2}\|\mathbf{b}_t\|_2^2$ (see Section 3.3).
$D(\boldsymbol{\nu})$	Objective value of the static dual problem given dual variable for static resources $\boldsymbol{\nu} = (\boldsymbol{\theta}, \boldsymbol{\lambda})$ (Definition 6).
$\boldsymbol{\nu}^* = (\boldsymbol{\theta}^*, \boldsymbol{\lambda}^*)$	Optimal dual variables for static resources solving the static dual problem (Definition 6).
$\tilde{\mathbf{z}}(\boldsymbol{\nu}, \mathbf{A})$	$\arg\max_{\mathbf{z} \in \mathcal{Z}(i^\dagger)} (\mathbf{w} - \boldsymbol{\theta} - \boldsymbol{\lambda}) \cdot \mathbf{z}$—primal decision in the static dual problem (Definition 6) given dual variable $\boldsymbol{\nu}$ for static resources and arrival type $\mathbf{A} = (\mathbf{w}, i^\dagger)$.

Table EC.2 Summary of Key Assumptions

Label	Description	Justification / Interpretation
A1	**Unknown i.i.d. Arrivals:** Each arrival $A_t = (\mathbf{w}_t, i_t^\dagger)$ is drawn i.i.d. from an unknown distribution.	Stationarity enables learning of arrival patterns (see Section EC.11.1 for empirical support). The unknown distribution is motivated by year-to-year variation in arrival composition (see Figures 1).
A2	**Service Slack:** There exists $\epsilon \geq 0$ such that $r_i \geq \rho_i + \epsilon$ for all $i \in [m]$.	Non-negativity of ϵ is required for backlog stability. Parameter ϵ distinguishes stable ($\epsilon = \Omega(1)$) vs. near-critical ($\epsilon = O(1/\sqrt{T})$) regimes (Definition 1).
A3	**Bernoulli Service Availability:** Server availability $s_{t,i}$ follows an i.i.d. Bernoulli process with success probability r_i.	This assumption is analogous to assuming that service times follow a geometric distribution with mean $1/r_i$. A practical motivation in the context of our application is provided in Footnote 14. An extension allowing for server idleness is discussed in Section EC.15.
A4	**Buffer for Free Cases:** For each $i \in [m]$, we assume $\rho_i - \mathbb{P}[i_t^\dagger = i] \geq d$ for some constant $d > 0$.	Ensures each affiliate has capacity for free cases with high probability (used in Proposition 1).
A5	**Reward Regularity:** The distribution of reward vectors $\mathbf{w}_t$ has a Lipschitz-continuous density (Assumption 1).	Technical assumption required to connect the convergence of dual variables to that of endogenous arrival rates for `CO-DL` (see Theorem 2 and Section 4.2.2).

EC.2. Further Related Work

In this section, we provide a more extensive review of related literature to our work.

Matching Policies for Refugee Resettlement: There has been growing academic interest in designing matching policies for refugee resettlement, leading to the two main approaches. The first approach is to consider a centralized *outcome-based* system, focusing on integration outcomes such as employment within 90 days, and design matching algorithms that target such objectives. Central to the outcome-based approach is the development of ML models that predict employment outcomes for a pair of case and affiliate (Bansak et al. 2018, 2024). Equipped with such models, several recent papers, including ours, aim to optimize matching decisions to improve employment, taking into account some of the operational considerations discussed in or fairness criteria Section 1.1 (Gölz and Procaccia 2019, Ahani et al. 2021, 2023, Bansak and Paulson 2024, Freund et al. 2024). The second approach focuses on designing a centralized *preference-based* matching system that respects the preferences of refugees and/or affiliates (see the work of Nguyen et al. (2021) and Delacrétaz et al. (2023) and references therein). While this approach has promise, most of the current programs, including the U.S. system, lack systematic data on preferences. Instead, they operate as a centralized outcome-based system.

Among the existing works, the most relevant work to ours is Bansak and Paulson (2024) and we highlight key differences from this work in Section 1.3. Beyond this comparison, our paper departs from prior work in several important ways: (i) While these works focused on designing effective heuristics,

we take a theoretical approach and propose algorithms with provable guarantees; (ii) Our work focuses on operational and algorithmic intricacies due to post-allocation service and dynamics of backlogs, which have not been studied before; (iii) The existing work designed algorithms that crucially rely on a pool of past years' data to either estimate the arrival distribution or simulate future arrivals. However, as elaborated in Section 1.1, such an approach can be *non-robust* to discrepancies between the data from past years and the arrivals from the current year. In contrast, our algorithms are robust to such discrepancies, as they do not rely on data from previous years; and (iv) As a result of such robustness, and in addition to enjoying theoretical guarantees, our method outperforms its counterpart in practical situations, as we discuss next.

Online Resource Allocation: Online resource allocation problems have been extensively studied in operations and computer science. Given the broad scope of the literature, we only highlight the stream most closely related to our setting. For an overview of other streams of work in the literature, we refer the readers to Mehta et al. (2007) and Huang et al. (2024) for an informative survey.

As mentioned in Section 1.2, without dynamic resources, our setting reduces to the online resource allocation problem with an unknown i.i.d. arrival model. Several papers designed and analyzed algorithms for such settings. The primary intuition guiding the design is that, under the stochastic input model, we can *learn* the arrival pattern by observing a small portion of the arrival data. Building on this intuition, earlier studies have explored primal methods (Kesselheim et al. 2018) that periodically solve a linear program using the observed data, or dual-based methods that seek to efficiently learn the dual variables of the associated offline problem and avoid the computational costs associated with resolving. These methods work by one-time dual learning (Devanur and Hayes 2009, Feldman et al. 2010, Molinaro and Ravi 2014) (which is similar in nature to the "predict-then-optimize" approach, e.g., Elmachtoub and Grigas (2022)), multiple-time dual learning (Agrawal et al. 2014, Li and Ye 2022), or using adversarial online learning throughout the horizon (Devanur et al. 2011, Agrawal and Devanur 2014, Wang et al. 2014, Gupta and Molinaro 2016, Badanidiyuru et al. 2018, Dughmi et al. 2021, Balseiro et al. 2023). All of these methods have also found further applications in various operational scenarios (Chen et al. 2017, Zhalechian et al. 2022). As highlighted in Section 1.3, our paper departs from this literature due to the time-varying nature of the dual problems introduced by the dynamic resources.

Another marginally related line of work is the literature on minimizing regret for Bayesian online resource allocation (see, for example, the work of Arlotto and Gurvich (2019) and Vera and Banerjee (2019)). Our work diverges from this line of work as the arrival distribution is assumed to be known in these models, and the main focus of the algorithms is to control the error propagation of decisions due to stochastic arrivals instead of learning the arrival. Our dynamic resources are also similar to the reusable resources studied in recent literature (see, for example, Gong et al. (2022), Feng et al. (2024), Delong et al. (2024), Goyal et al. (2025), Feng et al. (2021) and references therein). However,

the models in this stream of work do not allow one to wait for such resources. As such, there is no notion of congestion or desire for a smooth matching.

Dynamic Matching and Control of Queueing Systems: In queuing literature, a stream of work studies matching queues; in these systems, agents arrive over time and wait to be matched according to a deterministic compatibility structure (Aouad and Saritaç 2020, Aveklouris et al. 2021, Afeche et al. 2022, Gupta 2022, Hu and Zhou 2022, Kerimov et al. 2023, Wei et al. 2023) or a random one (Ashlagi et al. 2013, Anderson et al. 2017, Ashlagi et al. 2019). While most of these papers focus on maximizing total (match-dependent) rewards, the work of Aveklouris et al. (2021) considers an objective similar to ours that penalizes the matching reward by the cost of waiting. However, the crucial distinction of our paper from the aforementioned works is the *reverse order of matching and waiting*. As discussed in Section 1.2, our model can be viewed as a queuing system with an endogenous arrival rate controlled by the matching process. Once again, this subtle difference presents technical challenges in, for example, controlling the size of the backlog while performing learning and optimization.

From a methodological standpoint, the design and analysis of our first algorithm (Algorithm 1) partly rely on the celebrated drift-plus-penalty method (Tassiulas and Ephremides 1990, Lyapunov 1992, Neely 2006, Neely et al. 2008, Neely 2022) for stochastic network optimization and stability. Our work complements this approach by combining the drift-plus-penalty method with adversarial online learning. Furthermore, our first algorithm implicitly utilizes the scaled backlog as a dual variable, establishing a crucial structural connection to the projected subgradient method (Zinkevich 2003). A similar connection was observed in Ashlagi et al. (2022) and Kanoria and Qian (2023), albeit in a context different from the setup of our paper. We also note that, in a different context, Wei et al. (2023) also integrates the drift-plus-penalty method with the resolving techniques.

EC.3. Proof of Proposition 1

Fix a sample path $\{\mathbf{A}_t, \mathbf{s}_t\}_{t=1}^T$ under the event G_T. Given the matching profile $\{\mathbf{z}_t\}_{t=1}^T$, let us denote its objective value as

$$f(\{\mathbf{z}_t\}_{t=1}^T) = \sum_{t=1}^T \sum_{i=1}^m w_{t,i} z_{t,i} - \alpha \sum_{i=1}^m \left(\sum_{t=1}^T z_{t,i} - c_i \right)_+ - \frac{\gamma}{T} \sum_{t=1}^T \sum_{i=1}^m b_{t,i}$$

where $b_{t,i} = (b_{t,i} + z_{t,i} - s_{t,i})_+$ with $b_{0,i} = 0$ for all $t \in [T]$ and $i \in [m]$.[35] For brevity, we also define a primal and dual feasible set as:

$$\mathcal{P} := \left\{ \{\mathbf{z}_t\}_{t=1}^T \mid \mathbf{z}_t \in \mathcal{Z}(i_t^\dagger), \forall t \in [T] \right\}$$

$$\mathcal{D} := \Big\{ (\boldsymbol{\theta}, \boldsymbol{\lambda}, \{\boldsymbol{\beta}_t\}_{t=1}^T) \geq \mathbf{0} \mid \ \theta_i \leq \alpha \ \forall i \in [m], \ \beta_{t,i} - \beta_{t+1,i} \leq \frac{\gamma}{T} \ \forall t \in [T-1] \ i \in [m], \\ \beta_{T,i} \leq \frac{\gamma}{T} \ \forall i \in [m] \Big\}$$

[35] Note that this is equivalent to our original objective in Definition 2 by taking the maximum of (Objective) over the decision variables $\{\mathbf{b}_t\}_{t=1}^T$ for any given $\{\mathbf{z}_t\}_{t=1}^T$ subject to $b_{t,i} \geq 0$ and $b_{t,i} \geq b_{t-1,i} + z_{t,i} - s_{t,i}$ for all $t \in [T]$ and $i \in [m]$.

Consider the following Lagrangian function.

$$L\Big(\{\mathbf{z}_t\}_{t=1}^T, \boldsymbol{\theta}, \boldsymbol{\lambda}, \{\boldsymbol{\beta}_t\}_{t=1}^T\Big) = \sum_{t=1}^T \mathbf{w}_t \cdot \mathbf{z}_t + \underbrace{\sum_{t=1}^T \boldsymbol{\theta} \cdot (\boldsymbol{\rho} - \mathbf{z}_t)}_{\mathsf{CS}(\boldsymbol{\theta})} + \underbrace{\sum_{t=1}^T \boldsymbol{\lambda} \cdot (\boldsymbol{\rho} - \mathbf{z}_t)}_{\mathsf{CS}(\boldsymbol{\lambda})} + \underbrace{\sum_{t=1}^T \boldsymbol{\beta}_t \cdot (\mathbf{s}_t - \mathbf{z}_t)}_{\mathsf{CS}(\{\boldsymbol{\beta}_t\}_{t=1}^T)}.$$

Here, each $\mathsf{CS}(\cdot)$ term can be seen as a complementary slackness term corresponding to each dual variable. We will show that, under event G_T,

$$\begin{aligned}
\mathsf{Dual}(\alpha, \gamma) &= \min_{(\boldsymbol{\theta}, \boldsymbol{\lambda}, \{\boldsymbol{\beta}_t\}_{t=1}^T) \in \mathcal{D}} \max_{(\{\mathbf{z}_t\}_{t=1}^T) \in \mathcal{P}} L\Big(\{\mathbf{z}_t\}_{t=1}^T, \boldsymbol{\theta}, \boldsymbol{\lambda}, \{\boldsymbol{\beta}_t\}_{t=1}^T\Big) \\
&= \max_{(\{\mathbf{z}_t\}_{t=1}^T) \in \mathcal{P}} \min_{(\boldsymbol{\theta}, \boldsymbol{\lambda}, \{\boldsymbol{\beta}_t\}_{t=1}^T) \in \mathcal{D}} L\Big(\{\mathbf{z}_t\}_{t=1}^T, \boldsymbol{\theta}, \boldsymbol{\lambda}, \{\boldsymbol{\beta}_t\}_{t=1}^T\Big) \qquad \text{(EC.1)} \\
&= \mathsf{OPT}(\alpha, \gamma). \qquad \text{(EC.2)}
\end{aligned}$$

The first line follows from the definition of the dual function $\mathsf{Dual}(\alpha, \gamma)$ in the main statement. The second line is the consequence of Sion's min-max theorem (Sion 1958).[36] To show the last equality, we show that

$$\min_{(\boldsymbol{\theta}, \boldsymbol{\lambda}, \boldsymbol{\beta}_t) \in \mathcal{D}} L\Big(\{\mathbf{z}_t\}_{t=1}^T, \boldsymbol{\theta}, \boldsymbol{\lambda}, \{\boldsymbol{\beta}_t\}_{t=1}^T\Big) = \begin{cases} -\infty & \text{if } \sum_{t=1}^T \mathbf{z}_t > \mathbf{c} \\ f(\{\mathbf{z}_t\}_{t=1}^T) & \text{otherwise.} \end{cases} \qquad \text{(EC.3)}$$

Note that line (EC.2) directly follows from line (EC.3) because the capacity constraint of the offline benchmark (Definition 2) is equivalent to $\sum_{t=1}^T z_{t,i} \le c_i$ for all $i \in [m]$ under the event G_T. Hence, it suffices to show line (EC.3). Recall that $\mathbf{c} = T\boldsymbol{\rho}$ by definition of $\boldsymbol{\rho}$. If $\sum_{t=1}^T \mathbf{z}_t > \mathbf{c}$, then the optimal solution $\boldsymbol{\lambda}^* \ge \mathbf{0}$ of the inner minimization problem in line (EC.1) can grow arbitrarily large, hence making $\mathsf{CS}(\boldsymbol{\lambda})$ unbounded below. Otherwise, it is straightforward to see $\mathsf{CS}(\boldsymbol{\lambda}^*) = 0$ at the optimal $\boldsymbol{\lambda}^*$ of the inner minimization problem in line (EC.1). We now investigate the minimum value of $\mathsf{CS}(\boldsymbol{\theta})$ and $\mathsf{CS}(\{\boldsymbol{\beta}_t\}_{t=1}^T)$ over the dual feasible set $\mathcal{D}$. We first focus on the former. Note that the inner minimization problem in line (EC.1) is separable over $\boldsymbol{\theta}$ and $\{\boldsymbol{\beta}_t\}_{t=1}^T$. Hence, we have:

$$\min_{\boldsymbol{\theta} \in [0,\alpha]^m} \mathsf{CS}(\boldsymbol{\theta}) = -\max_{\boldsymbol{\theta} \in [0,\alpha]^m} \sum_{i=1}^m \theta_i \left(\sum_{t=1}^T z_{t,i} - c_i\right) = -\alpha \sum_{i=1}^m \left(\sum_{t=1}^T z_{t,i} - c_i\right)_+.$$

The the last inequality follows from the fact that $\alpha(x)_+ = \max_{\theta \in [0,\alpha]} \theta x$ for any $x \in \mathbb{R}$. In the similar way,

$$\begin{aligned}
&\min_{\{\boldsymbol{\beta}\}_{t=1}^T \ge \mathbf{0}} \mathsf{CS}(\{\boldsymbol{\beta}_t\}_{t=1}^T) \quad \text{s.t.} \quad \beta_{t,i} - \beta_{t+1,i} \le \frac{\gamma}{T} \ \forall t \in [T-1] \text{ and } i \in [m], \ \beta_{T,i} \le \frac{\gamma}{T} \ \forall i \in [m] \\
&= -\left[\max_{\{\boldsymbol{\beta}_t\}_{t=1}^T \ge \mathbf{0}} \sum_{t=1}^T \sum_{i=1}^m \beta_{t,i}(z_{t,i} - s_{t,i}) \quad \text{s.t} \quad \beta_{t,i} - \beta_{t+1,i} \le \frac{\gamma}{T} \ \forall t \in [T-1] \text{ and } i \in [m], \ \beta_{T,i} \le \frac{\gamma}{T} \ \forall i \in [m]\right] \\
&= -\left[\frac{\gamma}{T} \min_{\{\mathbf{b}_t\}_{t=1}^T \ge \mathbf{0}} \sum_{t=1}^T \sum_{i=1}^m b_{t,i} \quad \text{s.t.} \quad b_{t,i} \ge b_{t-1,i} + z_{t,i} - s_{t,i} \quad \forall t \in [T], i \in [m]\right] \\
&= -\frac{\gamma}{T} \sum_{t=1}^T \sum_{i=1}^m (b_{t-1,i} + z_{t,i} - s_{t,i})_+.
\end{aligned}$$

[36] More specifically, the Lagrangian is a linear function of the primal and dual variables. The primal feasible set $\mathcal{P}$ is convex and compact and the dual feasible set is also convex. Hence, we can switch the minimum and maximum operator by applying Sion's min-max theorem.

In the third line, we used the strong duality of the linear program. Note that, at the optimal solution of the third line, either of $b_{t,i} \geq 0$ or $b_{t,i} \geq b_{t-1,i} + z_{t,i} - s_{t,i}$ must bind. Hence, the last line follows. Combining, we have proved line (EC.3). This completes the proof.

EC.4. Proof of Proposition 2

We consider an instance with $m = 1$. For brevity, we omit the subscript for i. There are T arrivals with deterministic reward $w_t = 1$ for all $t \in [T]$. There are no tied cases and the capacity is $c = 0.5T$. The service rate is $r = 0.5 + \epsilon$ where $\epsilon \in [0, 0.5)$. Note that α does not play any role here. Hence, we denote the expected objective value of offline benchmark (Definition 2) by $\mathsf{OPT}(\gamma)$. In the following, we first obtain a lower bound on $\mathsf{OPT}(\gamma)$ and then an upper bound on the objective value of any online algorithm.

Step 1: Lower-bounding Objective Value of Optimal Offline.

We first analyze the objective value of the offline.

Claim EC.1. *For any* $\epsilon \geq 0$,

$$\mathbb{E}[\mathsf{OPT}(\gamma)] \geq 0.5T - \Theta(\sqrt{T})$$

Proof of Claim EC.1. Consider the following feasible solution of the offline program: set $z_t = \mathbb{1}[s_t = 1]$ while satisfying the capacity constraint (i,e, $\sum_{t=1}^{T} z_t \leq 0.5T$). Let S_T^ϵ be a binomial random variable with T trials and success probability $0.5 + \epsilon$. The proposed feasible solution never incurs the backlog, while obtaining $\min(S_T^\epsilon, 0.5T)$ reward. Hence, we have $\mathsf{OPT}(\gamma) \geq \min(S_T^\epsilon, 0.5T)$ for every sample path of arrivals and services. Taking the expectation of the preceding inequality, we have

$$\mathbb{E}[\mathsf{OPT}(\gamma)] \geq \mathbb{E}[\min(S_T^\epsilon, 0.5T)] \geq \mathbb{E}[\min(S_T^0, 0.5T)]$$

where the last inequality is because S_T^ϵ stochastically dominates S_T^0 for all $\epsilon \geq 0$ in the first order.[37] We now lower-bound the right-hand side of the above inequality by $0.5T - \Theta(\sqrt{T})$. Let us define $Z_T^0 := \dfrac{S_T^0 - 0.5T}{\sqrt{\mathrm{Var}[S_T^0]}}$. Then we have

$$\mathbb{E}[\min(S_T^0, 0.5T)] = 0.5T - \mathbb{E}[(0.5T - S_T^0)_+] = 0.5T - \sqrt{\mathrm{Var}[S_T^0]}\,\mathbb{E}[(-Z_T^0)_+] = 0.5T - \Theta(\sqrt{T}).$$

where the last equality is because $\mathbb{E}[(-Z_T^0)_+]$ has a constant mean for large T by the central limit theorem. This completes the proof of Claim EC.1. □

Step 2: Upper-bounding Objective Value of Any Online Algorithm.

We now turn our attention to upper-bounding the objective value of any online algorithm, denoted by ALG. Fix an online algorithm and let z_t denote the decision at time t. Without loss, we assume that $\mathbb{E}[\sum_{t=1}^{T} z_t] \geq \frac{T}{4}$ because otherwise, we have $\mathbb{E}[\mathsf{OPT}(\gamma)] - \mathbb{E}[\mathsf{ALG}] \geq 0.25T - \Theta(\sqrt{T}) = \Omega(T)$ from Claim EC.1.

[37] Random variable X stochastically dominates (in the first-order) if any only if $\mathbb{E}[u(X)] \geq \mathbb{E}[u(Y)]$ for all non-decreasing function u. Furthermore, let X and Y be binomial random variables with the respective success probability p_X and p_Y and the common number of trials T. If $p_X \geq p_Y$, X stochastically dominates Y in the first order (Shaked and Shanthikumar 2007).

CLAIM EC.2. *The expected value of any online algorithm for which* $\mathbb{E}[\sum_{t=1}^{T} z_t] \geq \frac{T}{4}$ *satisfies:*

$$\mathbb{E}[\mathsf{ALG}] \leq 0.5T - 0.25\gamma(0.5-\epsilon)$$

Proof of Claim EC.2. We first lower-bound the total cumulative backlog by

$$\mathbb{E}\left[\sum_{t=1}^{T} b_t\right] \geq \mathbb{E}\left[\sum_{t=1}^{T} z_t\right](0.5-\epsilon)\,. \tag{EC.4}$$

To see this, we first observe that

$$b_t \geq \mathbb{1}[s_t = 0] z_t$$

for every sample path. More specifically, if $s_t = 0$, then we have at least z_t amount of backlog at the end of period t. Otherwise, the bound is trivial. Taking the expectation of the preceding inequality conditional on history $\mathcal{H}_{t-1}$, we have

$$\begin{aligned}
\mathbb{E}[b_t|\mathcal{H}_{t-1}] &\geq \mathbb{E}[\mathbb{1}[s_t=0] z_t|\mathcal{H}_{t-1}] \\
&= \mathbb{E}\left[\mathbb{1}[s_t=0]|\mathcal{H}_{t-1}\right]\mathbb{E}[z_t|\mathcal{H}_{t-1}] \\
&= (0.5-\epsilon)\,\mathbb{E}[z_t|\mathcal{H}_{t-1}]
\end{aligned}$$

For the second line, we used the following fact: since any online algorithm does not know the realization of s_t at the time of making the decision z_t, the random variables z_t and s_t are conditionally independent given $\mathcal{H}_{t-1}$. The last line follows because s_t is an i.i.d. Bernoulli random variable with mean $0.5+\epsilon$. Taking outer expectation of the preceding inequality, we obtain that

$$\mathbb{E}[b_t] \geq (0.5-\epsilon)\,\mathbb{E}[z_t]$$

Summing up for all $t \in [T]$ gives the inequality (EC.4). Hence, we conclude that

$$\mathbb{E}[\mathsf{ALG}] = \mathbb{E}\left[\sum_{t=1}^{T} z_t\right] - \frac{\gamma}{T}\mathbb{E}\left[\sum_{t=1}^{T} b_t\right] \leq \mathbb{E}\left[\sum_{t=1}^{T} z_t\right] - \frac{\gamma}{T}(0.5-\epsilon)\mathbb{E}\left[\sum_{t=1}^{T} z_t\right] \leq 0.5T - 0.25\gamma(0.5-\epsilon).$$

In the last inequality, we used that $0.25T \leq \mathbb{E}[\sum_{t=1}^{T} z_t]$ and the fact that capacity is $0.5T$. □

Step 3: Putting Everything Together.

From Claim EC.1 and Claim EC.2, we conclude that

$$\mathbb{E}[\mathsf{OPT}] - \mathbb{E}[\mathsf{ALG}] \geq 0.25\gamma(0.5-\epsilon) - \Theta(\sqrt{T}).$$

for any online algorithm such that $\mathbb{E}[\sum_{t=1}^{T} z_t] \geq 0.25T$. Hence, whenever $\gamma = \Omega(T)$, the regret must be at least $\Omega(T)$. This completes the proof of Proposition 2.

EC.5. Alternative Interpretation of `CA-DL` (Algorithm 1)

In this appendix, we describe how our design idea of Algorithm 1 is connected to the theory of Lyapunov optimization, in particular drift-plus-penalty method (Neely 2022). To understand the connection of our algorithm to this method, it is helpful to think of two different policies designed for each of the following extremes. First, suppose we just want to control the average backlog without considering the net matching reward. The idea of Lyapunov optimization is that one can design a policy that can upper-bound an objective function by (i) defining a Lyapunov potential function (that is closely related to the original objective function) (ii) choosing a control that minimizes an *upper bound* of the expected drift (conditional on the history) of the potential function — often called Lyapunov drift. In our context, the potential function is the sum of squared backlogs, i.e., $\psi(\mathbf{b}_t) = \frac{1}{2}\|\mathbf{b}_t\|_2^2$ (as commonly used). Due to our drift lemma (Lemma 3), we know that $\psi(\mathbf{b}_t) - \psi(\mathbf{b}_{t-1}) \leq \mathbf{b}_{t-1} \cdot (\mathbf{z}_t - \mathbf{s}_t) + \mathcal{O}(1)$. Because the service rate of each affiliate i is $\rho_i + \epsilon$, the Lyapunov drift of $\psi(\cdot)$ is then upper bounded by

$$\mathbb{E}[\psi(\mathbf{b}_t) - \psi(\mathbf{b}_{t-1})|\mathcal{H}_{t-1}] \leq \mathbf{b}_{t-1} \cdot (\mathbb{E}[\mathbf{z}_t|\mathcal{H}_{t-1}] - \boldsymbol{\rho} - \epsilon\mathbf{1}) + \mathcal{O}(1). \tag{EC.5}$$

where $\mathbf{1}$ is a vector of ones. Hence, a policy that minimizes the upper bound of the Lyapunov drift is simply assigning a case t to an affiliate i with the *minimum* current backlog $b_{t-1,i}$. Note that this policy is equivalent to using an adjusted score $-b_{t-1,i}$ and matching a case to the affiliate with the maximum adjusted score. In fact, if we replace line 5 of Algorithm 1 with such a policy, one can show that the expected average backlog of such policy is $\mathcal{O}(1/\epsilon)$.

Now consider the other extreme where the goal is to maximize the net matching reward (without considering the average backlog). The literature of online resource allocation (see Section 1.3 for a review), and in particular the scoring techniques therein, suggests that there exists a time-invariant dual variable $\boldsymbol{\theta}^*$ (for the over-allocation cost) and $\boldsymbol{\lambda}^*$ (for the capacity constraint) where we can obtain the optimal net matching reward by assigning case t to affiliate i with the *maximum* adjusted score $(w_{t,i} - \theta_i^* - \lambda_i^*)$. Furthermore, under the i.i.d. arrival model, one can learn the dual variables by employing adversarial online learning. Importantly, the expected loss (regret) stemming from such learning is $\mathcal{O}(\sqrt{T})$ (Agrawal and Devanur 2014, Balseiro et al. 2023).

A natural way of interpolating both extremes is to employ a scoring policy with the adjusted score $\mathbf{w}_t - \boldsymbol{\theta}^* - \boldsymbol{\lambda}^* - \zeta\mathbf{b}_{t-1}$ where ζ is a parameter that encodes the "weight" on the objective of minimizing the average backlog compared to the other. A keen reader would note that this is reminiscent of the *drift-plus-penalty* method, a general technical framework designed to minimize the cumulative penalty function (in our context, a negative of the total net matching reward) of a queueing network while stabilizing the queues (see Neely (2022) and references therein for a more detailed review). The method, though developed in different contexts, similarly chooses a control action that greedily minimizes a linear combination of the penalty function and the Lyapunov drift of the queues.

The main novelty of our algorithm is that, because we do not know $(\boldsymbol{\theta}^*, \boldsymbol{\lambda}^*)$ a priori, we employ adversarial online learning techniques to adaptively learn (update) them. In this sense, our algorithm can be viewed as a novel combination of the drift-plus-penalty method with the adversarial online learning method. We remark that, while the proof involves many intricacies, the regret bound in Theorem 1 takes a natural combination of $\mathcal{O}(1/\epsilon)$ on the average backlog and $\mathcal{O}(\sqrt{T})$ on the loss of the net matching reward — which is consistent with each of the two aforementioned extreme objective functions.

EC.6. Missing Proofs of Section 3

Before going forward, we first present some technical preliminaries that we will frequently use. The following claim is the standard result of online mirror descent.

Claim EC.3 **(Online Mirror Descent (Bubeck 2015))**. *Let $\{f_t : \mathcal{D} \to \mathbb{R}\}$ be a sequence of convex functions and $\boldsymbol{\nu}_t$ be a sequence of iterates such that*

$$\begin{aligned} \nabla h(\tilde{\boldsymbol{\nu}}_{t+1}) &= \nabla h(\boldsymbol{\nu}_t) - \eta_t \hat{\mathbf{g}}_t \\ \boldsymbol{\nu}_{t+1} &= \arg\min_{\mathbf{x}\in\mathcal{D}} V_h(\mathbf{x}, \tilde{\boldsymbol{\nu}}_{t+1}) \end{aligned} \tag{EC.6}$$

where (1) $\mathcal{D}$ is a bounded convex set, (2) $\mathbb{E}[\hat{\mathbf{g}}_t|\mathcal{H}_{t-1}] = \nabla f_t(\boldsymbol{\nu}_t)$, and (3) h is a σ-strongly convex function with respect to $\|\cdot\|$ norm on set $\mathcal{D}$,. For any $\boldsymbol{\nu} \in \mathcal{D}$ and positive integers (k, s), we have

$$\sum_{t=k}^{s}(f_t(\boldsymbol{\nu}_t) - f_t(\boldsymbol{\nu})) \le \sum_{t=k}^{s}\frac{\eta_t\|\hat{\mathbf{g}}_t\|_*^2}{2\sigma} + \frac{1}{\eta_k}V_h(\boldsymbol{\nu}, \boldsymbol{\nu}_k) + \sum_{t=k+1}^{s}\left(\frac{1}{\eta_t} - \frac{1}{\eta_{t-1}}\right)V_h(\boldsymbol{\nu}, \boldsymbol{\nu}_t) + \sum_{t=k}^{s}\hat{\mathbf{u}}_t\cdot(\boldsymbol{\nu}_t - \boldsymbol{\nu}) \tag{EC.7}$$

where $\hat{\mathbf{u}}_t := \mathbb{E}[\hat{\mathbf{g}}_t|H_{t-1}] - \hat{\mathbf{g}}_t$ (i.e., noise of the gradient) and $V_h(\mathbf{x}, \mathbf{y}) = h(\mathbf{x}) - h(\mathbf{y}) - \nabla h(\mathbf{y})\cdot(\mathbf{x} - \mathbf{y})$ is the Bregman divergence with respect to h.

The stated theorem is slightly different from the one from Bubeck (2015) because we allow for the step size to be time-varying in Section 4. For the sake of completeness, we reproduce the proof of Claim EC.3 in Appendix EC.7.3.

The following claim shows that we can always "linearize" the total over-allocation cost, which will be crucial throughout the proof.

Claim EC.4 **(Dual Representation of Over-allocation Cost)**. *There exists $\boldsymbol{\theta}^\star \in [0, \alpha]^m$ for which*

$$\alpha\sum_{i=1}^{m}\left(\sum_{t=1}^{T} z_{t,i} - c_i\right)_+ = \sum_{t=1}^{T}\boldsymbol{\theta}^\star\cdot(\mathbf{z}_t - \boldsymbol{\rho}) \tag{EC.8}$$

Proof of Claim EC.4. The proof simply follows from $\alpha\cdot(x)_+ = \max_{\theta\in[0,\alpha]}\theta x$. □

EC.6.1. Proof of Lemma 3

We prove both the lower and upper bound on the drift separately.

Lower-bound on drift: To show the lower-bound on the drift $\phi_i(b_{t,i}) - \phi_i(b_{t-1,i})$, note that the potential function $\psi(x) = \frac{1}{2}\|\mathbf{x}\|_2^2 = \sum_i \phi_i(x)$, where each $\phi_i(x) \triangleq \frac{1}{2}x_i^2$ is convex. Now, either $b_{t-1,i} = 0$ and in that case $\phi_i(b_{t,i}) - \phi_i(b_{t-1,i}) \geq 0 = b_{t-1,i}(z_{t,i} - s_{t,i})$, or $b_{t-1,i} \neq 0$ and therefore $b_{t,i} = b_{t-1,i} + z_{t,i} - s_{t,i}$.[38] In this case, as $\phi_i(x)$ is convex, we have:

$$\phi_i(b_{t,i}) - \phi_i(b_{t-1,i}) \geq \phi_i'(b_{t-1,i})(b_{t,i} - b_{t-1,i}) = b_{t-1,i}(z_{t,i} - s_{t,i}) \ .$$

Putting two cases together and summing over all i, we conclude that

$$\psi(\mathbf{b}_t) - \psi(\mathbf{b}_{t-1}) \geq \mathbf{b}_{t-1} \cdot (\mathbf{z}_t - \mathbf{s}_t)$$

Upper-bound on drift: To show the upper-bound on the drift $\phi_i(b_{t,i}) - \phi_i(b_{t-1,i})$, we first note that $b_{t,i} = (b_{t-1,i} + z_{t,i} - s_{t,i})_+ \leq |b_{t-1,i} + z_{t,i} - s_{t,i}|$. Hence, we always have $b_{t,i}^2 \leq (b_{t-1,i} + z_{t,i} - s_{t,i})^2$. Rearranging this inequality, we have

$$\frac{b_{t,i}^2 - b_{t-1,i}^2}{2} \leq b_{t-1,i}(z_{t,i} - s_{t,i}) + \frac{(z_{t,i} - s_{t,i})^2}{2}.$$

Summing the inequality over $i \in [m]$, we have

$$\psi(\mathbf{b}_t) - \psi(\mathbf{b}_{t-1}) \leq \mathbf{b}_{t-1} \cdot (\mathbf{z}_t - \mathbf{s}_t) + \frac{\|\mathbf{z}_t - \mathbf{s}_t\|_2^2}{2}.$$

We highlight that while we proved the above inequality from first principles, we could derive it directly using the fact that $\frac{1}{2}\|x\|_2^2$ is a 1-Lipschitz smooth function. Finally, the squared norm $\|\mathbf{z}_t - \mathbf{s}_t\|_2^2$ is trivially upper-bounded by $1 + m = \mathcal{O}(1)$. This completes the proof.

EC.6.2. Proof of Lemma 4

Let $\boldsymbol{\beta}_t = \zeta \mathbf{b}_{t-1}$ (see equation (8) in Section 3.2 and related discussions), which is the implicit dual variables for dynamic resources. We recall from line (15) that

$$\begin{aligned} K_t &:= \mathbf{w}_t \cdot \mathbf{z}_t + \boldsymbol{\theta}_t \cdot (\boldsymbol{\rho} - \mathbf{z}_t) + \boldsymbol{\lambda}_t \cdot (\boldsymbol{\rho} - \mathbf{z}_t) - \zeta D_t \\ &\geq (\mathbf{w}_t - \boldsymbol{\theta}_t - \boldsymbol{\lambda}_t - \boldsymbol{\beta}_t) \cdot \mathbf{z}_t + \boldsymbol{\theta}_t \cdot \boldsymbol{\rho} + \boldsymbol{\lambda}_t \cdot \boldsymbol{\rho} + \boldsymbol{\beta}_t \cdot \mathbf{s}_t - \mathcal{O}(\zeta) \end{aligned} \tag{EC.9}$$

where the last inequality follows from Lemma 3. Furthermore, we note that for all $t \leq T_A$, we recall from equation (17) that

$$\mathbf{z}_t \in \arg\max_{\mathbf{z} \in \mathcal{Z}(i_t^\dagger)} (\mathbf{w}_t - \boldsymbol{\theta}_t - \boldsymbol{\lambda}_t - \boldsymbol{\beta}_t) \cdot \mathbf{z} \tag{EC.10}$$

In the similar spirit of Talluri and Van Ryzin (1998), the following claim shows that there always exists a static control whose value (roughly) upper-bounds the net matching reward of the offline benchmark (Definition 2) (part (c)) with a marginal matching rate at most $\boldsymbol{\rho}$ in expectation (part (b)).

[38] The equation is always valid as long as $z_{t,i} \in \{0, 1\}$, which is true for Algorithm 1.

CLAIM EC.5 **(Static Control).** *Consider the following static control problem:*

$$P^\star_{\mathcal{Z}} := \max_{\substack{\mathbf{z}(\mathbf{A})\in\mathcal{Z}(i^\dagger)\\ \forall \mathbf{A}\in\mathcal{A}}} \mathbb{E}[\mathbf{w}\cdot\mathbf{z}(\mathbf{A})] \quad s.t.\ \mathbb{E}\left[\mathbf{z}(\mathbf{A})\right] \le \boldsymbol{\rho} \tag{EC.11}$$

where the expectation is with respect to $\mathbf{A}=(\mathbf{w},i^\dagger)\sim\mathcal{F}$. *Let* $\bar{\mathbf{z}}:\mathcal{A}\to\Delta_m$ *be the optimal static control that solves the above program. Then we have*

(a) $\bar{\mathbf{z}}(\mathbf{A})\in\mathcal{Z}(i^\dagger)$ *for any* $\mathbf{A}=(\mathbf{w},i^\dagger)$

(b) $\mathbb{E}_{\mathbf{A}}[\bar{\mathbf{z}}(\mathbf{A})]\le\boldsymbol{\rho}$

(c) $P^\star_{\mathcal{Z}}=\mathbb{E}_{\mathbf{A}}[\mathbf{w}\cdot\bar{\mathbf{z}}(\mathbf{A})]\ge \dfrac{\mathbb{E}[\mathsf{NMR}(\{\hat{\mathbf{z}}_t\}_{t=1}^T;\alpha)]}{T}-\mathcal{O}(1/T^2)$ *for and any matching profile* $\{\hat{\mathbf{z}}_t\}_{t=1}^T$ *that satisfies*

$$\hat{\mathbf{z}}_t\in\mathcal{Z}(i_t^\dagger)\quad\forall t\in[T],\quad \sum_{t=1}^T \mathbb{1}[i_t^\dagger=0]z_{t,i}\le\left(c_i-\sum_{t=1}^T\mathbb{1}[i_t^\dagger=i]z_{t,i}\right)_+\quad\forall i\in[m]. \tag{EC.12}$$

Note that, due to our assumption that $\rho_i\ge\mathbb{P}[i^\dagger=i]$, the program is always feasible and therefore $\mathsf{P}^\star_{\mathcal{Z}}$ is well-defined. We defer the proof of Claim EC.5 to the end of this section, but first we complete the proof of Lemma 4 building on this claim. Based on the static control $\bar{\mathbf{z}}(\cdot)$ defined in Claim EC.5, we define an alternative matching decision as

$$\bar{\mathbf{z}}_t := \bar{\mathbf{z}}(\mathbf{A}_t). \tag{EC.13}$$

By line (EC.9) and the optimality criterion (EC.10), for $t\le T_A$, we have

$$K_t\ge(\mathbf{w}_t-\boldsymbol{\theta}_t-\boldsymbol{\lambda}_t-\beta_t)\cdot\bar{\mathbf{z}}_t+\boldsymbol{\theta}_t\cdot\boldsymbol{\rho}+\boldsymbol{\lambda}_t\cdot\boldsymbol{\rho}+\beta_t\cdot\mathbf{s}_t-\mathcal{O}(\zeta)$$

for every sample path. Taking the expectation conditional on history $\mathcal{H}_{t-1}=\{\mathbf{A}_\tau,\mathbf{s}_\tau\}_{\tau=1}^{t-1}$, we obtain the following: for any matching profile $\{\hat{\mathbf{z}}_t\}_{t=1}^T$ that satisfies (EC.12),

$$\begin{aligned}
\mathbb{E}[K_t|\mathcal{H}_{t-1}] &\ge \mathbb{E}[(\mathbf{w}_t-\boldsymbol{\theta}_t-\boldsymbol{\lambda}_t-\beta_t)\cdot\bar{\mathbf{z}}_t+\boldsymbol{\theta}_t\cdot\boldsymbol{\rho}+\boldsymbol{\lambda}_t\cdot\boldsymbol{\rho}+\beta_t\cdot\mathbf{s}_t-\mathcal{O}(\zeta)|\mathcal{H}_{t-1}]\\
&= \mathbb{E}[\mathbf{w}_t\cdot\bar{\mathbf{z}}_t+\boldsymbol{\theta}_t\cdot(\boldsymbol{\rho}-\bar{\mathbf{z}}_t)+\boldsymbol{\lambda}_t\cdot(\boldsymbol{\rho}-\bar{\mathbf{z}}_t)+\beta_t\cdot(\mathbf{s}_t-\bar{\mathbf{z}}_t)-\mathcal{O}(\zeta)|\mathcal{H}_{t-1}]\\
&= \mathbb{E}[\mathbf{w}_t\cdot\bar{\mathbf{z}}_t|\mathcal{H}_{t-1}]+\boldsymbol{\theta}_t\cdot\mathbb{E}[(\boldsymbol{\rho}-\bar{\mathbf{z}}_t)|\mathcal{H}_{t-1}]+\boldsymbol{\lambda}_t\cdot\mathbb{E}[(\boldsymbol{\rho}-\bar{\mathbf{z}}_t)|\mathcal{H}_{t-1}]+\\
&\quad\ \beta_t\cdot\mathbb{E}[(\mathbf{s}_t-\bar{\mathbf{z}}_t)|\mathcal{H}_{t-1}]-\mathcal{O}(\zeta) \qquad\qquad (\text{EC.14})\\
&\ge \frac{\mathbb{E}[\mathsf{NMR}(\{\hat{\mathbf{z}}_t\}_{t=1}^T;\alpha)]}{T}-\mathcal{O}\Big(\frac{1}{T^2}\Big)+\zeta\epsilon\|\mathbf{b}_{t-1}\|_1-\mathcal{O}(\zeta) \qquad\qquad (\text{EC.15})
\end{aligned}$$

Line (EC.14) is because $(\boldsymbol{\theta}_t,\boldsymbol{\lambda}_t,\beta_t)$ are $\mathcal{H}_{t-1}$-measurable (that is, knowing the exact realizations of $\mathcal{H}_{t-1}$ determines the value of the dual variables). In line (EC.15), we used the property of the static control $\bar{\mathbf{z}}(\cdot)$ stated in Claim EC.5. To be precise, we first explain how we relate $\mathbb{E}[\mathbf{w}_t\cdot\bar{\mathbf{z}}_t|\mathcal{H}_{t-1}]$ of line (EC.14) to $\frac{\mathbb{E}[\mathsf{NMR}(\{\hat{\mathbf{z}}_t\}_{t=1}^T;\alpha)]}{T}$ in line (EC.15). Note that the static control $\bar{\mathbf{z}}(\cdot)$ only depends on the arrival distribution and the current arrival type. Hence, combined with the i.i.d. nature of the

arrival process, we have $\mathbb{E}[\mathbf{w}_t \cdot \bar{\mathbf{z}}_t | \mathcal{H}_{t-1}] = \mathbb{E}[\mathbf{w}_t \cdot \bar{\mathbf{z}}_t] = \mathsf{P}^\star_{\mathcal{Z}}$. Furthermore, from Claim EC.5-(c), we have $\mathbb{E}[\mathbf{w}_t \cdot \bar{\mathbf{z}}_t] \geq \frac{\mathbb{E}[\mathsf{NMR}(\{\hat{\mathbf{z}}_t\}_{t=1}^T;\alpha)]}{T} - \mathcal{O}(1/T^2)$. From the same line of reasoning, we can apply Claim EC.5-(b) to show that $\mathbb{E}[(\boldsymbol{\rho} - \bar{\mathbf{z}}_t)|\mathcal{H}_{t-1}] \geq \mathbf{0}$. Lastly, Claim EC.5-(b) again implies $\mathbb{E}[(\mathbf{s}_t - \bar{\mathbf{z}}_t)|\mathcal{H}_{t-1}] \geq \epsilon \mathbf{1}$ where $\mathbf{1}$ is the vector of ones. Combining these lower-bounds with $\boldsymbol{\theta}_t \geq \mathbf{0}$, $\boldsymbol{\lambda}_t \geq \mathbf{0}$ and $\boldsymbol{\beta}_t = \zeta \mathbf{b}_{t-1} \geq \mathbf{0}$, we obtain the final line (EC.15).

We now sum line (EC.15) for $t \in [T_A]$ to deduce that, for any feasible matching $\{\hat{\mathbf{z}}_t\}_{t=1}^T$ and filtration $\{\mathcal{H}_t\}_{t=1}^T$:

$$\begin{aligned}
\sum_{t=1}^{T_A} \mathbb{E}[K_t|\mathcal{H}_{t-1}] &\geq \frac{T_A \mathbb{E}[\mathsf{NMR}(\{\hat{\mathbf{z}}_t\}_{t=1}^T;\alpha)]}{T} - \mathcal{O}\Big(\frac{T_A}{T^2}\Big) + \zeta\epsilon \sum_{t=1}^{T_A-1} \|\mathbf{b}_t\|_1 - \mathcal{O}(T_A\zeta) \\
&= \mathbb{E}[\mathsf{NMR}(\{\hat{\mathbf{z}}_t\}_{t=1}^T;\alpha)] - \frac{T-T_A}{T}\mathbb{E}[\mathsf{NMR}(\{\hat{\mathbf{z}}_t\}_{t=1}^T;\alpha)] - \mathcal{O}\Big(\frac{T_A}{T^2}\Big) + \zeta\epsilon \sum_{t=1}^{T_A-1} \|\mathbf{b}_t\|_1 - \mathcal{O}(T_A\zeta) \\
&\geq \mathbb{E}[\mathsf{NMR}(\{\hat{\mathbf{z}}_t\}_{t=1}^T;\alpha)] - (T-T_A) + \zeta\epsilon \sum_{t=1}^{T_A-1} \|\mathbf{b}_t\|_1 - \mathcal{O}(T\zeta).
\end{aligned}$$

In the last line, we used the fact that (i) $\mathsf{NMR}(\{\hat{\mathbf{z}}_t\}_{t=1}^T;\alpha) \leq T$ (because the per-period reward is at most one) and (ii) $T_A \leq T$ for every sample path. (note that $\mathcal{O}(T_A/T^2) = \mathcal{O}(1/T)$, which is absorbed by $\mathcal{O}(T\zeta)$). We finally take the outer expectation of the preceding inequality over the entire history to obtain

$$\mathbb{E}\left[\sum_{t=1}^{T_A} \mathbb{E}[K_t|\mathcal{H}_{t-1}]\right] \geq \mathbb{E}[\mathsf{NMR}(\{\hat{\mathbf{z}}_t\}_{t=1}^T;\alpha)] - \mathbb{E}\left[(T-T_A)\right] + \zeta\epsilon\mathbb{E}\left[\sum_{t=1}^{T_A-1} \|\mathbf{b}_t\|_1\right] - \mathcal{O}(T\zeta). \tag{EC.16}$$

It only remains to prove that the left-hand side is equivalent to

$$\mathbb{E}\left[\sum_{t=1}^{T_A} \mathbb{E}[K_t|\mathcal{H}_{t-1}]\right] = \mathbb{E}\left[\sum_{t=1}^{T_A} K_t\right]. \tag{EC.17}$$

To prove this, define a stochastic process $Y_t = K_t - \mathbb{E}[K_t|\mathcal{H}_{t-1}]$ and $X_t = \sum_{\tau=1}^t Y_t$ with $X_0 := 0$. We observe that Y_t is $\mathcal{H}_t$-measurable and $\mathbb{E}[Y_t|\mathcal{H}_{t-1}] = 0$. Hence, $\{Y_t\}$ is a Martingale difference sequence with respect to filtration $\{\mathcal{H}_t\}$. Furthermore, we note that T_A is a bounded stopping time with respect to the filtration $\{\mathcal{H}_t\}$. Hence, the optional stopping theorem implies that $\mathbb{E}[X_{T_A}] = X_0 = 0$, which implies line (EC.17). This completes the proof of Lemma 4.

Proof of Claim EC.5. For ease of reference, we rewrite the definition of the static control defined in the claim.

$$\mathsf{P}^\star_{\mathcal{Z}} := \max_{\substack{\mathbf{z}(\mathbf{A}) \in \mathcal{Z}(i^\dagger) \\ \forall \mathbf{A} \in \mathcal{A}}} \mathbb{E}[\mathbf{w} \cdot \mathbf{z}(\mathbf{A})] \quad \text{s.t. } \mathbb{E}\left[\mathbf{z}(\mathbf{A})\right] \leq \boldsymbol{\rho} \tag{EC.18}$$

where the expectation is with respect to $\mathbf{A} = (\mathbf{w}, i^\dagger) \sim \mathcal{F}$. This optimization problem can be viewed as the stochastic program where the decision is to choose a function (control) $\mathbf{z} : \mathcal{A} \to \Delta_m$ subject to $\mathbf{z}(\mathbf{A}) \in \mathcal{Z}(i^\dagger)$. That is, for each arrival type $\mathbf{A} = (\mathbf{w}, i^\dagger)$, we choose the matching rate $\mathbf{z}(\mathbf{A}) \in \Delta_m$ if $i^\dagger = 0$ (free case) or we set it as $\mathbf{z}(\mathbf{A}) = \mathbf{e}_{i^\dagger}$ otherwise. The objective is to maximize the functional

$\mathbb{E}[\mathbf{w}\cdot\mathbf{z}(\mathbf{A})]$, which is the expected per-period matching reward given the control $\mathbf{z}(\cdot)$, subject to the (ex-ante) capacity constraint on the matching rate. Define $\bar{\mathbf{z}}(\cdot)$ as the optimal mapping that solves the problem in (EC.18). In the following, we show that $\bar{\mathbf{z}}(\cdot)$ satisfies the condition (a)-(c) stated in Claim EC.5.

Proof of Claim EC.5 (a) and (b). This trivially follows from the feasibility condition of $\bar{\mathbf{z}}(\cdot)$ in line (EC.18).

Proof of Claim EC.5 (c). Let $\{\hat{\mathbf{z}}_t\}_{t=1}^T$ be an arbitrary (potentially random) sequence that satisfies

$$\hat{\mathbf{z}}_t \in \mathcal{Z}(i_t^\dagger) \quad \forall t\in[T], \quad \sum_{t=1}^T \mathbb{1}[i_t^\dagger=0] z_{t,i} \le \left(c_i - \sum_{t=1}^T \mathbb{1}[i_t^\dagger=i] z_{t,i}\right)_+ \quad \forall i\in[m]. \tag{EC.19}$$

It suffices to prove

$$\mathsf{P}_{\mathcal{Z}}^\star \ge \mathbb{E}\Big[\frac{\mathsf{NMR}(\{\hat{\mathbf{z}}_t\}_{t=1}^T;\alpha)}{T}\Big] - \mathcal{O}(1/T^2).$$

To prove this, we now consider the dual problem of $\mathsf{P}_{\mathcal{Z}}^\star$, defined as:

$$\mathsf{D}_{\mathcal{Z}}(\boldsymbol{\phi}) := \mathbb{E}\left[\max_{\mathbf{z}(\mathbf{A})\in\mathcal{Z}(i^\dagger)} (\mathbf{w}-\boldsymbol{\phi})\cdot\mathbf{z}(\mathbf{A}) + \boldsymbol{\rho}\cdot\boldsymbol{\phi}\right], \quad \mathsf{D}_{\mathcal{Z}}^\star := \min_{\boldsymbol{\phi}\ge\mathbf{0}} \mathsf{D}(\boldsymbol{\phi}). \tag{EC.20}$$

By the compactness and convexity of $\mathcal{Z}(i^\dagger)$ for all $i^\dagger\in\{0\}\cup[m]$, a straightforward application of Sion's minmax theorem (Sion 1958) implies that $\mathsf{D}_{\mathcal{Z}}^\star = \mathsf{P}_{\mathcal{Z}}^\star$. Furthermore, as we show in Claim EC.31 (see Section EC.8.2), we can restrict the domain to $\|\boldsymbol{\phi}\|_\infty \le 1$ without loss of optimality. In summary, we have

$$\mathsf{P}_{\mathcal{Z}}^\star = \mathsf{D}_{\mathcal{Z}}^\star = \min_{\boldsymbol{\phi}\ge\mathbf{0}} \mathsf{D}_{\mathcal{Z}}(\boldsymbol{\phi}) = \min_{\boldsymbol{\phi}\ge\mathbf{0},\|\boldsymbol{\phi}\|_\infty\le 1} \mathsf{D}_{\mathcal{Z}}(\boldsymbol{\phi}) \tag{EC.21}$$

Let $N_{T,i} := \sum_{t=1}^T \mathbb{1}[i^\dagger = i]$ denote the total number of tied cases at affiliate i over the horizon. Define event G_T as

$$G_T := \{N_{T,i} \le c_i,\ \forall i\in[m]\}. \tag{EC.22}$$

Note that, due to our assumption that $\rho - \mathbb{P}[i^\dagger = i] = \Omega(1)$, a straightforward application of Hoeffding's inequality and union bound implies that $\mathbb{P}[G_T] \ge 1-\mathcal{O}(\exp(-T)) \ge 1-\mathcal{O}(1/T^2)$. Given these ingredients, we show that $\mathsf{P}_{\mathcal{Z}}^\star \ge \mathbb{E}[\frac{\mathsf{NMR}(\{\hat{\mathbf{z}}_t\}_{t=1}^T;\alpha)}{T}] - \mathcal{O}(1/T^2)$ in the following.

$$\begin{aligned}
&\mathbb{E}[\mathsf{NMR}(\{\hat{\mathbf{z}}_t\}_{t=1}^T;\alpha)] \\
&\le \mathbb{E}\left[\max_{\substack{\mathbf{z}_t\in\mathcal{Z}(i_t^\dagger),\\ t\in[T]}} \sum_{t=1}^T \mathbf{w}_t\cdot\mathbf{z}_t - \alpha\sum_{i=1}^m\left(\sum_{t=1}^T z_{t,i} - c_i\right)_+ \text{ s.t. } \sum_{t=1}^T \mathbb{1}[i_t^\dagger=0] z_{t,i} \le \left(c_i - \sum_{t=1}^T \mathbb{1}[i_t^\dagger=i] z_{t,i}\right)_+, \forall i\in[m]\right] \qquad \text{(EC.23)} \\
&\le \mathbb{E}\left[\left\{\max_{\substack{\mathbf{z}_t\in\mathcal{Z}(i_t^\dagger),\\ t\in[T]}} \sum_{t=1}^T \mathbf{w}_t\cdot\mathbf{z}_t - \alpha\sum_{i=1}^m\left(\sum_{t=1}^T z_{t,i} - c_i\right)_+ \right.\right. \\
&\qquad \left.\left. \text{s.t. } \sum_{t=1}^T \mathbb{1}[i_t^\dagger=0] z_{t,i} \le \left(c_i - \sum_{t=1}^T \mathbb{1}[i_t^\dagger=i] z_{t,i}\right)_+, \forall i\in[m]\right\}\cdot\mathbb{1}[G_T]\right] + \mathcal{O}(1/T) \qquad \text{(EC.24)}
\end{aligned}$$

$$= \mathbb{E}\left[\left(\max_{\mathbf{z}_t \in \mathcal{Z}(i_t^\dagger), t\in[T]} \sum_{t=1}^{T} \mathbf{w}_t \cdot \mathbf{z}_t \quad \text{s.t.} \quad \sum_{t=1}^{T} z_{t,i} \le c_i, \forall i \in [m]\right) \cdot \mathbb{1}[G_T]\right] + \mathcal{O}(1/T) \tag{EC.25}$$

$$\le \mathbb{E}\left[\left(\max_{\mathbf{z}_t \in \mathcal{Z}(i_t^\dagger), t\in[T]} \sum_{t=1}^{T} \mathbf{w}_t \cdot \mathbf{z}_t - \boldsymbol{\phi} \cdot \left(\sum_{t=1}^{T} \mathbf{z}_t - \mathbf{c}\right)\right) \cdot \mathbb{1}[G_T]\right] + \mathcal{O}(1/T) \quad \forall \boldsymbol{\phi} \ge \mathbf{0},\ \|\boldsymbol{\phi}\|_\infty \le 1 \tag{EC.26}$$

$$= \sum_{t=1}^{T} \mathbb{E}\left[\max_{\mathbf{z}_t \in \mathcal{Z}(i_t^\dagger)} (\mathbf{w}_t - \boldsymbol{\phi}) \cdot \mathbf{z}_t + \boldsymbol{\phi} \cdot \boldsymbol{\rho} \middle| G_T\right] \mathbb{P}[G_T] + \mathcal{O}(1/T) \quad \forall \boldsymbol{\phi} \ge \mathbf{0},\ \|\boldsymbol{\phi}\|_\infty \le 1 \tag{EC.27}$$

$$\le \sum_{t=1}^{T} \{\mathsf{D}_{\mathcal{Z}}(\boldsymbol{\phi}) + \mathcal{O}(1/T^2)\} + \mathcal{O}(1/T) \quad \forall \boldsymbol{\phi} \ge \mathbf{0},\ \|\boldsymbol{\phi}\|_\infty \le 1 \tag{EC.28}$$

$$= T\mathsf{D}_{\mathcal{Z}}(\boldsymbol{\phi}) + \mathcal{O}(1/T) \quad \forall \boldsymbol{\phi} \ge \mathbf{0},\ \|\boldsymbol{\phi}\|_\infty \le 1 \tag{EC.29}$$

$$\le T\mathsf{D}^\star_{\mathcal{Z}} + \mathcal{O}(1/T) \tag{EC.30}$$

We elaborate on each line in the following. Line (EC.23) follows from maximizing the net matching reward subject to line (EC.19). In line (EC.24), we bounded the expected total net matching reward under the event G_T^c by using that that $\mathbb{P}[G_T] \ge 1 - \mathcal{O}(1/T^2)$ and the total net matching reward is $\mathcal{O}(T)$. Line (EC.25) is because the capacity constraint in line (EC.24) reduces to $\sum_{t=1}^{T} z_{t,i} \le c_i$ under event G_T. Lines (EC.26) follows from weak-duality.

To understand line (EC.28), let $\hat{D}_{\mathcal{Z}}(\boldsymbol{\phi}; \mathbf{A}_t) := \max_{\mathbf{z}_t \in \mathcal{Z}(i_t^\dagger)} (\mathbf{w}_t - \boldsymbol{\phi}) \cdot \mathbf{z}_t + \boldsymbol{\phi} \cdot \boldsymbol{\rho}$. For any $\boldsymbol{\phi}$ such that $\|\boldsymbol{\phi}\|_\infty \le 1$, arrival $\mathbf{A}_t = (\mathbf{w}_t, i_t^\dagger)$, and $\mathbf{z}_t \in \mathcal{Z}(i_t^\dagger)$,

$$|(\mathbf{w}_t - \boldsymbol{\phi}) \cdot \mathbf{z}_t + \boldsymbol{\phi} \cdot \boldsymbol{\rho}| \le \|\mathbf{w}_t\|_\infty \|\mathbf{z}_t\|_1 + \|\boldsymbol{\phi}\|_\infty \|\mathbf{z}_t\|_1 + \|\boldsymbol{\phi}\|_\infty \|\boldsymbol{\rho}\|_1 \le 3. \tag{EC.31}$$

The first inequality is due to Cauchy-Schwartz inequality, and the last inequality is because of our assumption that the reward is at most one and $\sum_{i=1}^{m} \rho_i \le 1$ (which was without loss of generality). Hence, we must have $|\hat{D}_{\mathcal{Z}}(\boldsymbol{\phi}; \mathbf{A}_t)| \le 3$ for any arrival $\mathbf{A}_t$ and $\|\boldsymbol{\phi}\|_\infty \le 1$. We now use this fact to derive (EC.28) as follows:

$$\mathsf{D}_{\mathcal{Z}}(\boldsymbol{\phi}) = \mathbb{E}[\hat{D}_{\mathcal{Z}}(\boldsymbol{\phi}; \mathbf{A}_t)] \tag{EC.32}$$

$$= \mathbb{E}[\hat{D}_{\mathcal{Z}}(\boldsymbol{\phi}; \mathbf{A}_t) | G_T] \mathbb{P}[G_T] + \mathbb{E}[\hat{D}_{\mathcal{Z}}(\boldsymbol{\phi}; \mathbf{A}_t) | G_T^c] \mathbb{P}[G_T^c] \tag{EC.33}$$

$$\ge \mathbb{E}[\hat{D}_{\mathcal{Z}}(\boldsymbol{\phi}; \mathbf{A}_t) | G_T] \mathbb{P}[G_T] - \mathcal{O}(1/T^2). \tag{EC.34}$$

The first line is by definition of $\mathsf{D}_{\mathcal{Z}}(\cdot)$ (see line (EC.20)) and the arrival is i.i.d. The second line is just re-writing the expectation. The last line is because $|\hat{D}_{\mathcal{Z}}(\boldsymbol{\phi}; \mathbf{A}_t)| \le 3 = \mathcal{O}(1)$ for all arrival $\mathbf{A}_t$ and $\|\boldsymbol{\phi}\|_\infty \le 1$, along with $\mathbb{P}[G_T^c] \le \mathcal{O}(1/T^2)$. Hence, from line (EC.34), we directly deduce line (EC.28).

Finally, line (EC.30) follows from taking minimum over $\boldsymbol{\phi}$ of the inequality (EC.29), along with line (EC.21). The proof is complete because $\mathsf{D}^\star_{\mathcal{Z}} = \mathsf{P}^\star_{\mathcal{Z}}$ by line (EC.21). □

EC.6.3. Proof of Lemma 5

By the definition of the pseudo-rewards, we have the following:

$$\begin{aligned}
\sum_{t=1}^{T_A} K_t &= \sum_{t=1}^{T_A} \{\mathbf{w}_t \cdot \mathbf{z}_t + \boldsymbol{\theta}_t \cdot (\boldsymbol{\rho} - \mathbf{z}_t) + \boldsymbol{\lambda}_t \cdot (\boldsymbol{\rho} - \mathbf{z}_t) - \zeta D_t\} \\
&= \sum_{t=1}^{T_A} \mathbf{w}_t \cdot \mathbf{z}_t + \sum_{t=1}^{T_A} \boldsymbol{\theta}_t \cdot (\boldsymbol{\rho} - \mathbf{z}_t) + \sum_{t=1}^{T_A} \boldsymbol{\lambda}_t \cdot (\boldsymbol{\rho} - \mathbf{z}_t) - \zeta \psi(\mathbf{b}_{T_A}) && \text{(EC.35)} \\
&= \sum_{t=1}^{T_A} \mathbf{w}_t \cdot \mathbf{z}_t - \alpha \sum_{i=1}^{m} \left(\sum_{t=1}^{T} z_{t,i} - c_i \right)_+ + \sum_{t=1}^{T_A} \boldsymbol{\theta}_t \cdot (\boldsymbol{\rho} - \mathbf{z}_t) - \sum_{t=1}^{T} \boldsymbol{\theta}^\star \cdot (\boldsymbol{\rho} - \mathbf{z}_t) + \\
&\quad \sum_{t=1}^{T_A} \boldsymbol{\lambda}_t \cdot (\boldsymbol{\rho} - \mathbf{z}_t) - \zeta \psi(\mathbf{b}_{T_A}) && \text{(EC.36)} \\
&\le \underbrace{\sum_{t=1}^{T} \mathbf{w}_t \cdot \mathbf{z}_t - \alpha \sum_{i=1}^{m} \left(\sum_{t=1}^{T} z_{t,i} - c_i \right)_+}_{\mathsf{NMR}(\{\mathbf{z}_t\}_{t=1}^T)} + \underbrace{\sum_{t=1}^{T} \boldsymbol{\theta}_t \cdot (\boldsymbol{\rho} - \mathbf{z}_t) - \sum_{t=1}^{T} \boldsymbol{\theta}^\star \cdot (\boldsymbol{\rho} - \mathbf{z}_t)}_{R_\theta} + \\
&\quad \underbrace{\sum_{t=1}^{T_A} \boldsymbol{\lambda}_t \cdot (\boldsymbol{\rho} - \mathbf{z}_t)}_{R_\lambda} + 2\alpha(T - T_A) - \zeta \psi(\mathbf{b}_{T_A}) && \text{(EC.37)}
\end{aligned}$$

Line (EC.35) is because of the telescoping sum of D_t and $\mathbf{b}_0 = \mathbf{0}$. In line (EC.36), we added and subtracted the over-allocation cost and used Claim EC.4. In line (EC.37), we used the Cauchy-Schwartz inequality to bound $\sum_{t=1}^{T_A} \boldsymbol{\theta}_t \cdot (\boldsymbol{\rho} - \mathbf{z}_t) \le \sum_{t=1}^{T} \boldsymbol{\theta}_t \cdot (\boldsymbol{\rho} - \mathbf{z}_t) + 2\alpha(T - T_A)$ by using that (i) $\boldsymbol{\theta} \in [0, \alpha]^m$, (ii) $\|\boldsymbol{\rho} - \mathbf{z}_t\|_1 \le \|\boldsymbol{\rho}\|_1 + \|\mathbf{z}_t\|_1 \le 2$ (by definition of $\mathbf{z}_t$ and our assumption that $\|\boldsymbol{\rho}\|_1 \le 1$).

In the following claims, we show that $R_\theta \le \mathcal{O}(\sqrt{T})$ and $R_\lambda \le \mathcal{O}(\sqrt{T}) - (1 + 2\alpha)(T - T_A)$. Plugging the bound of R_θ and R_λ into inequality (EC.37) completes the proof of Lemma 5.

Claim EC.6. *$R_\theta \le \mathcal{O}(\sqrt{T})$ for every sample path.*

Proof of Claim EC.6. We first describe how we can view $\{\boldsymbol{\theta}_t\}_{t=1}^T$ as a sequence of the online mirror descent iterates for properly defined primitives stated in Claim EC.3. Let $f_t(\boldsymbol{\theta}) := \boldsymbol{\theta} \cdot (\boldsymbol{\rho} - \mathbf{z}_t)$, $\mathcal{D} := [0, \alpha]^m$, $\hat{\mathbf{g}}_t := \boldsymbol{\rho} - \mathbf{z}_t$, and $h(\boldsymbol{\theta}) := \sum_{i=1}^m \theta_i \log(\theta_i)$. Note that the gradient has no noise (i.e., $\hat{\mathbf{u}}_t = \mathbf{0}$). With these primitives, it is straightforward to see that the update rule (EC.6) is reduced to that of line (7) (for dual variables $\boldsymbol{\theta}$) in Algorithm 1. Furthermore, $\|\hat{\mathbf{g}}\|_1 \le \|\mathbf{z}_t\|_1 + \|\boldsymbol{\rho}\|_1 \le 2$ and $h(\cdot)$ is $(1/\alpha)$-strongly convex with respect to $\|\cdot\|_\infty$ in $\mathcal{D}$. Hence, we invoke Claim EC.3 with the fixed step size $\eta_t = \eta$:

$$R_\theta = \sum_{t=1}^{T} (f_t(\boldsymbol{\theta}_t) - f_t(\boldsymbol{\theta}^\star)) \le 2\alpha\eta T + \frac{1}{\eta} V_h(\boldsymbol{\theta}^\star, \boldsymbol{\theta}_1) \qquad \text{(EC.38)}$$

The bregman distance $D(\boldsymbol{\theta}^\star, \boldsymbol{\theta}_1)$ is bounded because $\boldsymbol{\theta}^\star$ and $\boldsymbol{\theta}_1$ are all in $\mathcal{D}$ by definition. Hence, with $\eta = \Theta(1/\sqrt{T})$, the right-hand side of (EC.38) is $\mathcal{O}(\sqrt{T})$. □

Claim EC.7. *$R_\lambda \le \mathcal{O}(\sqrt{T}) - (1 + 2\alpha)(T - T_A)$ for every sample path.*

Proof of Claim EC.7. Let us first denote the upper-bound of λ as $\bar{\lambda} := \frac{1+2\alpha}{\underline{\rho}}$. Similar to the proof of Claim EC.6, we define $f_t(\boldsymbol{\lambda}) := \boldsymbol{\lambda} \cdot (\boldsymbol{\rho} - \mathbf{z}_t)$, $\mathcal{D} := [0, \bar{\lambda}]^m$, $\hat{\mathbf{g}}_t := \boldsymbol{\rho} - \mathbf{z}_t$, and $h(\boldsymbol{\lambda}) := \sum_{i=1}^m \lambda_i \log(\lambda_i)$. It is straightforward to see that the update rule (EC.6) is reduced to that of line (7) (for dual variable $\boldsymbol{\lambda}$) in Algorithm 1. Furthermore, $\|\hat{\mathbf{g}}\|_1 \le 2$ and $h(\cdot)$ is $1/\bar{\lambda}$-strongly convex with respect to $\|\cdot\|_\infty$ in $\mathcal{D}$. Hence, we invoke Claim EC.3 with the fixed step size $\eta_t = \eta$ to obtain that, for any $\boldsymbol{\lambda}^\star \in [0, \bar{\lambda}]^m$,

$$R_\lambda \le \underbrace{2\bar{\lambda}\eta T + \frac{1}{\eta} V_h(\boldsymbol{\lambda}^\star, \boldsymbol{\lambda}_1)}_{\blacklozenge} + \underbrace{\sum_{t=1}^{T_A} \boldsymbol{\lambda}^\star \cdot (\boldsymbol{\rho} - \mathbf{z}_t)}_{\diamondsuit}$$

Since both $\boldsymbol{\lambda}^\star$ and $\boldsymbol{\lambda}_1$ are bounded, the term $\blacklozenge$ is $\mathcal{O}(\sqrt{T})$ with $\eta = \Theta(1/\sqrt{T})$. We now show that, by properly choosing $\boldsymbol{\lambda}^\star$, the term $\diamondsuit$ is upper bounded by $(1+2\alpha)(T_A - T)$. We consider two cases. First, if $T_A = T$, then we take $\boldsymbol{\lambda}^\star = \mathbf{0}$ and the bound is trivial. Otherwise, $T_A < T$ and there exists i such that $\sum_{t=1}^{T_A} z_{t,i} \ge \rho_i T$ (see line (16) in Section 3.3). Take that coordinate i and let $\boldsymbol{\lambda}^\star = \frac{1+2\alpha}{\rho_i}\mathbf{e}_i$. Note that $\boldsymbol{\lambda}^\star \in [0, \bar{\lambda}]^m$ by definition of $\bar{\lambda}$. The term $\diamondsuit$ is then given by

$$\begin{aligned}
\diamondsuit &= \frac{1+2\alpha}{\rho_i} \sum_{t=1}^{T_A} (\rho_i - z_{t,i}) \\
&\le \frac{1+2\alpha}{\rho_i} ((T_A - T)\rho_i) \\
&= (1+2\alpha)(T_A - T)
\end{aligned}$$

The second line is by the definition of the coordinate i we have chosen. The last line simply follows from $\underline{\rho} \le \rho_i$. Hence, taking the worst case over $\boldsymbol{\lambda}^* \in \{\mathbf{0}, \frac{1+2\alpha}{\rho_1}\mathbf{e}_1, \frac{1+2\alpha}{\rho_2}\mathbf{e}_2, ..., \frac{1+2\alpha}{\rho_1}\mathbf{e}_m\}$, the term $\diamondsuit$ is always at most $(1+2\alpha)(T_A - T)$. This completes the proof. □

EC.6.4. Missing Details for Step 3: Finishing the Proof of Lemmas 1 and 2

With inequality (20) in place to prove Lemma 1, we set $\hat{\mathbf{z}}_t = \mathbf{z}_t^*$ for all $t \in [T]$ (i.e., the optimal offline allocation in Definition 2) and observe that $(\mathsf{B}') \ge 0$ because the backlog is non-negative. The proof is complete by plugging $\zeta = \Theta(1/\sqrt{T})$ as stated in Theorem 1.

To prove Lemma 2, we set $\{\hat{\mathbf{z}}_t\}_{t=1}^T$ to be the matching profile that maximizes the net matching reward subject to $\hat{\mathbf{z}}_t \in \mathcal{Z}(i_t^\dagger)$ for all $t \in [T]$ and (Capacity Feasibility-T). By definition of such $\{\hat{\mathbf{z}}_t\}_{t=1}^T$, term (A) is nonnegative and hence, term (B') is upper bounded by $\mathcal{O}(\sqrt{T} + \zeta T)$. Hence, the average backlog *up to* the stopping time is $\mathcal{O}(1/\epsilon)$. The remaining step is to show that the average backlog *after* the stopping time is not dominant. Observe that (i) by construction of `CA-DL`, only tied cases can arrive after T_A and (ii) the arrival rate of the tied case is strictly smaller than $\boldsymbol{\rho}$. Therefore, the backlog process is a random walk with a negative drift. In the following, we combine these observations with a drift analysis of the potential function $\psi(\cdot)$ to show that the average backlog accrued after the stopping time is $\mathcal{O}(1)$.

We begin by decomposing the average backlog for the entire horizon as

$$\frac{1}{T}\mathbb{E}\left[\sum_{t=1}^{T}\|\mathbf{b}_t\|_1\right]=\frac{1}{T}\mathbb{E}\left[\sum_{t=1}^{T_A-1}\|\mathbf{b}_t\|_1\right]+\frac{1}{T}\mathbb{E}\left[\sum_{t=T_A}^{T}\|\mathbf{b}_t\|_1\right] \tag{EC.39}$$

The first (second, resp.) term is the contribution of the backlog until (after, resp.) the stopping time. We first bound the first term. Recall from the inequality (20) that

$$\zeta\epsilon\mathbb{E}\left[\sum_{t=1}^{T_A-1}\|\mathbf{b}_t\|_1\right]+\zeta\mathbb{E}[\psi(\mathbf{b}_{T_A})]\le\mathcal{O}(\sqrt{T}+\zeta T). \tag{EC.40}$$

Noting that $\psi(\cdot)$ is always non-negative, we straightforwardly obtain

$$\frac{1}{T}\mathbb{E}\left[\sum_{t=1}^{T_A-1}\|\mathbf{b}_t\|_1\right]\le\mathcal{O}\Big(\frac{1}{\epsilon}(1+\frac{1}{\zeta\sqrt{T}})\Big)\le\mathcal{O}\Big(\frac{1}{\epsilon}\Big). \tag{EC.41}$$

where the last inequality is because we set $\zeta=\Theta(1/\sqrt{T})$ in the stable regime.

We now turn our attention to bounding the contribution of the backlog after $t\ge T_A$. Toward this goal, we first obtain a bound of the backlog at the stopping time.

Claim EC.8. $\mathbb{E}[\|\mathbf{b}_{T_A}\|_1^2]\le\mathcal{O}(T)$

Proof of Claim EC.8. The proof is the direct consequence of line (EC.40). To be precise, because the backlog is non-negative, the line (EC.40) implies that

$$\mathbb{E}[\psi(\mathbf{b}_{T_A})]\le\mathcal{O}\left(T+\frac{\sqrt{T}}{\zeta}\right)=\mathcal{O}(T)$$

where the last inequality follows since $\zeta=\Theta(1/\sqrt{T})$. We finally recall the definition of $\psi(\mathbf{b}_{T_A})=\frac{1}{2}\|\mathbf{b}_{T_A}\|_2^2$. By Cauchy-Schwarz inequality, we have $\|\mathbf{b}_{T_A}\|_1^2\le m\|\mathbf{b}_{T_A}\|_2^2$ (for every sample path). Hence, $\mathcal{O}(T)$ bound on $\mathbb{E}[\psi(\mathbf{b}_{T_A})]$ is equivalent of showing the same order of bound on $\mathbb{E}[\|\mathbf{b}_{T_A}\|_1^2]$.[39] This completes the proof. □

Building on this claim and the fact that only the tied cases can arrive at each affiliate after $t\ge T_A$, we now prove that the average backlog after the stopping time is $\mathcal{O}(1)$, which completes the proof of Lemma 2.

Claim EC.9. $\frac{1}{T}\mathbb{E}\left[\sum_{t=T_A}^{T}\|\mathbf{b}_t\|_1\right]\le\mathcal{O}(1)$

Proof of Claim EC.9. For each affiliate i and $t\ge T_A+1$, we have

$$\mathbb{E}[b_{t,i}^2-b_{t-1,i}^2|\mathcal{H}_{t-1}]\le b_{t-1,i}(\mathbb{E}[z_{t,i}-s_{t,i}|\mathcal{H}_{t-1,i}])+\mathcal{O}(1) \tag{EC.42}$$

$$=b_{t-1,i}(\mathbb{P}[i_t^\dagger=i]-\rho_i-\epsilon)+\mathcal{O}(1) \tag{EC.43}$$

$$\le-(d+\epsilon)b_{t-1,i}+\mathcal{O}(1) \tag{EC.44}$$

[39] Because of our assumption that the capacity grows linearly with T, the number of affiliates must be $O(1)$.

The first line follows from (one-dimensional version of) Lemma 3. For the second line, we used the fact that only tied cases can arrive at each affiliate after the stopping time, along with the i.i.d. nature of the arrival and service process. For the last line, we used our assumption that there exists constant $d>0$ such that $\rho_i - \mathbb{P}[i^\dagger = i] \geq d$. Summing up the previous inequalities over $T_A+1 \leq t \leq T$ (and using the similar Martingale argument used to justify line (EC.17)), we deduce that

$$\mathbb{E}[b_{T,i}^2 - b_{T_A,i}^2] \leq -(d+\epsilon)\mathbb{E}\Big[\sum_{t=T_A}^{T-1} b_{t,i}\Big] + O(\mathbb{E}[T-T_A]). \tag{EC.45}$$

Combined with Claim EC.8, the above inequality directly implies that $\mathbb{E}\Big[\sum_{t=T_A}^{T-1} b_{t,i}\Big] \leq \mathcal{O}(T)$ and $\mathbb{E}[b_{T,i}] \leq \mathcal{O}(\sqrt{T})$. Summing these two bounds across all $i \in [m]$, we conclude that

$$\frac{1}{T}\mathbb{E}\left[\sum_{t=T_A}^{T} \|\mathbf{b}_t\|_1\right] \leq \mathcal{O}(1). \tag{EC.46}$$

This completes the proof. □

EC.6.5. Some Remarks on Algorithm 1

EC.6.5.1. Bound on the Expected Stopping Time

For technical reasons, our analysis relied on taking inaction for free cases after the first time an affiliate reaches its capacity (i.e. the stopping time T_A defined in line (16)). In the following, under a mild assumption, we show that the stopping time is near the end of the horizon in expectation. To introduce this assumption, we recall the static control introduced in Claim EC.5.

$$\mathsf{P}^\star_{\mathcal{Z}} := \max_{\substack{\mathbf{z}(\mathbf{A}) \in \mathcal{Z}(i^\dagger) \\ \forall \mathbf{A} \in \mathcal{A}}} \mathbb{E}[\mathbf{w} \cdot \mathbf{z}(\mathbf{A})] \quad \text{s.t. } \mathbb{E}[\mathbf{z}(\mathbf{A})] \leq \boldsymbol{\rho} \tag{EC.47}$$

We make a mild assumption that the value of the static control is a strictly positive constant as well as the penalty cost parameter α (for over-allocation)

Assumption EC.10.

(a) The penalty cost parameter for over-allocation is $\alpha > 1$ (that is, greater than the maximum reward of each case).

(b) The arrival distribution $\mathcal{F}$ is such that $\mathsf{P}^\star_{\mathcal{Z}} = \Omega(1)$.[40]

The following proposition shows that, under the above assumption, the stopping time is very near the end of the horizon in expectation.

Proposition EC.11. *Under the stable regime (Definition 1) and Assumption EC.10, we have* $\mathbb{E}[T_A] \geq T - \mathcal{O}(\sqrt{T})$.

[40] A sufficient condition for this assumption is, for example, that the reward distribution satisfies $\mathbb{E}[\mathbf{w} \cdot \boldsymbol{\rho}] = \Omega(1)$.

Proof of Proposition EC.11. The proof follows similar steps of that for Lemma 4. First, we establish a lower bound on the expected pseudo-reward as follows.

LEMMA EC.12. $\mathbb{E}\Big[\sum_{t=1}^{T_A} K_t\Big] \geq \mathsf{P}^\star_{\mathcal{Z}} T - \mathbb{E}[T - T_A] - \mathcal{O}(\zeta T)$.

Proof of Lemma EC.12. The proof only requires a minor modification of the steps taken in Section EC.6.2. Specifically, we have

$$\begin{aligned}\mathbb{E}[K_t|\mathcal{H}_{t-1}] &\geq \mathbb{E}[\mathbf{w}_t \cdot \bar{\mathbf{z}}_t|\mathcal{H}_{t-1}] + \boldsymbol{\theta}_t \cdot \mathbb{E}[(\boldsymbol{\rho} - \bar{\mathbf{z}}_t)|\mathcal{H}_{t-1}] + \boldsymbol{\lambda}_t \cdot \mathbb{E}[(\boldsymbol{\rho} - \bar{\mathbf{z}}_t)|\mathcal{H}_{t-1}] + \boldsymbol{\beta}_t \cdot \mathbb{E}[(\mathbf{s}_t - \bar{\mathbf{z}}_t)|\mathcal{H}_{t-1}] - \mathcal{O}(\zeta) \\ &\geq \mathsf{P}^\star_{\mathcal{Z}} - \mathcal{O}(\zeta)\end{aligned}$$

where the first line follows from line (EC.14) and the second line is because of Claim EC.5. Summing the inequality over $t \in [T_A]$ and taking expectation, we obtain:

$$\mathbb{E}\Big[\sum_{t=1}^{T_A} K_t\Big] \geq \mathsf{P}^\star_{\mathcal{Z}} \mathbb{E}[T_A] - \mathcal{O}(\zeta T) = \mathsf{P}^\star_{\mathcal{Z}} T - \mathsf{P}^\star_{\mathcal{Z}} \mathbb{E}[T - T_A] - \mathcal{O}(\zeta T) \geq \mathsf{P}^\star_{\mathcal{Z}} T - \mathbb{E}[T - T_A] - \mathcal{O}(\zeta T) \quad \text{(EC.48)}$$

where the last equality is because the per-period reward is at most one. This completes the proof. □

We now establish an upper bound on the expected pseudo-rewards in terms of $\mathbb{E}[T_A]$.

LEMMA EC.13. $\mathbb{E}\Big[\sum_{t=1}^{T_A} K_t\Big] \leq \mathbb{E}[T_A] \mathsf{P}^\star_{\mathcal{Z}} - \mathbb{E}[T - T_A] + \mathcal{O}(\sqrt{T})$

Proof of Lemma EC.13. The proof is a modification of the steps taken in Section EC.6.3. Recall the dual program of the static control problem defined in Claim EC.5:

$$\mathsf{D}_{\mathcal{Z}}(\boldsymbol{\phi}) := \mathbb{E}\left[\max_{\mathbf{z}(\mathbf{A}) \in \mathcal{Z}(i^\dagger)} (\mathbf{w} - \boldsymbol{\phi}) \cdot \mathbf{z}(\mathbf{A}) + \boldsymbol{\rho} \cdot \boldsymbol{\phi}\right], \quad \mathsf{D}^\star_{\mathcal{Z}} := \min_{\boldsymbol{\phi} \geq \mathbf{0}} \mathsf{D}_{\mathcal{Z}}(\boldsymbol{\phi}).$$

By the compactness and convexity of $\mathcal{Z}(i^\dagger)$ for all $i^\dagger \in \{0\} \cup [m]$, a straightforward application of Sion's minmax theorem (Sion 1958) implies that $\mathsf{D}^\star_{\mathcal{Z}} = \mathsf{P}^\star_{\mathcal{Z}}$.

Equipped with the definition of $\mathsf{D}^\star_{\mathcal{Z}}$, we now obtain the following:

$$\begin{aligned}\sum_{t=1}^{T_A} K_t &= \sum_{t=1}^{T_A} \{\mathbf{w}_t \cdot \mathbf{z}_t + \boldsymbol{\theta}_t \cdot (\boldsymbol{\rho} - \mathbf{z}_t) + \boldsymbol{\lambda}_t \cdot (\boldsymbol{\rho} - \mathbf{z}_t) - \zeta D_t\} \\ &= \sum_{t=1}^{T_A} \mathbf{w}_t \cdot \mathbf{z}_t + \sum_{t=1}^{T_A} \boldsymbol{\theta}_t \cdot (\boldsymbol{\rho} - \mathbf{z}_t) + \sum_{t=1}^{T_A} \boldsymbol{\lambda}_t \cdot (\boldsymbol{\rho} - \mathbf{z}_t) - \zeta \psi(\mathbf{b}_{T_A}) \\ &\leq \sum_{t=1}^{T_A} \mathbf{w}_t \cdot \mathbf{z}_t + \underbrace{\sum_{t=1}^{T_A} \boldsymbol{\theta}_t \cdot (\boldsymbol{\rho} - \mathbf{z}_t)}_{R_\theta} + \underbrace{\sum_{t=1}^{T_A} \boldsymbol{\lambda}_t \cdot (\boldsymbol{\rho} - \mathbf{z}_t)}_{R_\lambda}\end{aligned} \quad \text{(EC.49)}$$

The second line is due to the telescoping sum of D_t, along with $\mathbf{b}_0 = \mathbf{0}$. The last line follows from the non-negativity of the backlog. We now bound R_θ and R_λ as follows. For the former, define $\boldsymbol{\phi}^* \geq \mathbf{0}$ such that $\mathsf{D}_{\mathcal{Z}}(\boldsymbol{\phi}^*) = \mathsf{D}^\star_{\mathcal{Z}}$. In Claim EC.31-(a) (see Section EC.8.2), we establish that $\|\boldsymbol{\phi}^*\|_\infty \leq 1$ and

therefore $\|\boldsymbol{\phi}^*\|_\infty \le \alpha$ (by Assumption EC.10). Hence, by mirroring the arguments in Claim EC.6, we invoke Claim EC.3 to obtain

$$R_\theta \le \sum_{t=1}^{T_A} \boldsymbol{\phi}^* \cdot (\boldsymbol{\rho} - \mathbf{z}_t) + \mathcal{O}(\sqrt{T}). \tag{EC.50}$$

We now bound R_λ. Let $\bar{\lambda} = \frac{1+2\alpha}{\underline{\rho}}$. By mirroring the arguments for the proof of Claim EC.7, we invoke Claim EC.3 to obtain that, for any $\boldsymbol{\lambda}^\star \in [0, \bar{\lambda}]^m$, $R_\lambda \le \mathcal{O}(\sqrt{T}) + \sum_{t=1}^{T_A} \boldsymbol{\lambda}^\star \cdot (\boldsymbol{\rho} - \mathbf{z}_t)$. We now show that, by properly choosing $\boldsymbol{\lambda}^\star$, the second term $\sum_{t=1}^{T_A} \boldsymbol{\lambda}^\star \cdot (\boldsymbol{\rho} - \mathbf{z}_t)$ is upper bounded by $(T_A - T)$. We consider two cases. First, if $T_A = T$, then we take $\boldsymbol{\lambda}^\star = \mathbf{0}$ and the bound is trivial. Otherwise, $T_A < T$ and there exists i such that $\sum_{t=1}^{T_A} z_{t,i} \ge \rho_i T$ (see line (16) in Section 3.3). Take that coordinate i and let $\boldsymbol{\lambda}^\star = \frac{1}{\rho_i}\mathbf{e}_i$. Note that $\boldsymbol{\lambda}^\star \in [0, \bar{\lambda}]^m$ by definition of $\bar{\lambda}$. Hence, the term $\diamondsuit$ is then given by

$$\frac{1}{\rho_i}\sum_{t=1}^{T_A}(\rho_i - z_{t,i}) = \frac{1}{\rho_i}\sum_{t=1}^{T_A}(\rho_i - z_{t,i}) \le \frac{1}{\rho_i}((T_A - T)\rho_i) = T_A - T.$$

The first inequality is by the definition of the coordinate i we have chosen. Taking the worst case over $\boldsymbol{\lambda}^* \in \{\mathbf{0}, \frac{1}{\rho_1}\mathbf{e}_1, \frac{1}{\rho_2}\mathbf{e}_2, ..., \frac{1}{\rho_1}\mathbf{e}_m\}$ of the above inequality, the term $\diamondsuit$ is always at most $T_A - T$. Hence, we have deduced that

$$R_\lambda \le \mathcal{O}(\sqrt{T}) + T_A - T \tag{EC.51}$$

Plugging the bounds (EC.50) and (EC.51) into line (EC.49) and taking expectation, we have

$$\begin{aligned}
\mathbb{E}\Big[\sum_{t=}^{T_A} K_t\Big] &\le \Big[\sum_{t=1}^{T_A}(\mathbf{w}_t - \boldsymbol{\phi}^*)\cdot\mathbf{z}_t + \boldsymbol{\rho}\cdot\boldsymbol{\phi}^*\Big] - \mathbb{E}[T - T_A] + \mathcal{O}(\sqrt{T}) \\
&\le \Big[\sum_{t=1}^{T_A}\max_{\mathbf{z}_t \in \mathcal{Z}(i_t^\dagger)}(\mathbf{w}_t - \boldsymbol{\phi}^*)\cdot\mathbf{z}_t + \boldsymbol{\rho}\cdot\boldsymbol{\phi}^*\Big] - \mathbb{E}[T - T_A] + \mathcal{O}(\sqrt{T}) \\
&= \mathbb{E}[T_A]\mathsf{D}^\star_{\mathcal{Z}} - \mathbb{E}[T - T_A] + \mathcal{O}(\sqrt{T})
\end{aligned}$$

In the last line, we used Wald's equation and the i.i.d. nature of the arrival process. The proof is complete by noting that $\mathsf{D}^\star_{\mathcal{Z}} = \mathsf{P}^\star_{\mathcal{Z}}$. □

To complete the proof of Proposition EC.11, we combine Lemmas EC.12 and EC.13 to obtain

$$\mathsf{P}^\star_{\mathcal{Z}}\mathbb{E}[T - T_A] \le \mathcal{O}(\sqrt{T} + \zeta T) \le \mathcal{O}(\sqrt{T}) \tag{EC.52}$$

where the last inequality is because $\zeta = \mathcal{O}(1/\sqrt{T})$ under the stable regime. The proof is complete by dividing both sides of the inequality by $\mathsf{P}^\star_{\mathcal{Z}}$, which is justified by Assumption EC.10. □

EC.6.5.2. Removing Dummy Affiliate

In the main body of our paper, we have allowed inaction for free cases throughout the entire horizon. In this appendix, we partially relax this assumption. Specifically, we consider a variant of Algorithm 1

which cannot take inaction for free cases before the first time an affiliate reaches its capacity. Formally, for a given target affiliate $i^\dagger$, define a type-feasibility set (without any inaction) as

$$\mathcal{X}(i_t^\dagger) = \begin{cases} \{\mathbf{z} \in \mathbb{R}_+ : \sum_{i=1}^m z_i = 1\} & \text{if } i_t^\dagger = 0 \\ \{\mathbf{e}_{i^\dagger}\} & \text{otherwise.} \end{cases} \tag{EC.53}$$

For $t \le T_A$ (see line (16) for its definition), we now modify 5 of Algorithm 1 as

$$\mathbf{z}_t \in \arg\max_{\mathbf{z} \in \mathcal{X}(i_t^\dagger)} (\mathbf{w}_t - \boldsymbol{\theta}_t - \boldsymbol{\lambda}_t - \zeta \mathbf{b}_{t-1}) \cdot \mathbf{z}. \tag{EC.54}$$

The rest of the algorithm remains unchanged (including line 6 that leaves free cases unmatched once any of the affiliates exhaust its capacity). We call this algorithm $\overline{\texttt{CA-DL}}$. In the following, we show that $\overline{\texttt{CA-DL}}$ has the same theoretical guarantee as CA-DL under the stable regime and an additional assumption that $\sum_{i=1}^m \rho_i = 1$.

Corollary EC.14. *Let $\eta = \Theta(1/\sqrt{T})$ and $\zeta = \Theta(1/\sqrt{T})$. Under the stable regime (Definition 1) and $\sum_{i=1}^m \rho_i = 1$, the regret (Definition 3) of $\overline{\texttt{CA-DL}}$ is $\mathcal{O}(\sqrt{T} + \frac{\gamma}{\epsilon})$.*

Proof of Corollary EC.14. The proof mirrors that of Theorem 1 except for a minor change of Claim EC.5 (See Section EC.6.2). Specifically, we need to modify the claim by defining a static control problem *without any inaction*, and prove that the value of the optimal static control is an upper bound on the net matching value of the offline benchmark (Definition 2).

Claim EC.15 **(Static Control: Modification of Claim EC.5)**. *Consider the following static control problem:*

$$P_{\mathcal{X}}^\star := \max_{\substack{\mathbf{z}(\mathbf{A}) \in \mathcal{X}(i^\dagger) \\ \forall \mathbf{A} \in \mathcal{A}}} \mathbb{E}[\mathbf{w} \cdot \mathbf{z}(\mathbf{A})] \quad s.t.\ \mathbb{E}[\mathbf{z}(\mathbf{A})] \le \boldsymbol{\rho} \tag{EC.55}$$

where the expectation is with respect to $\mathbf{A} = (\mathbf{w}, i^\dagger) \sim \mathcal{F}$. Let $\tilde{\mathbf{z}} : \mathcal{A} \to \Delta_m$ be the optimal static control that solves the above program. Then we have

(a) $\tilde{\mathbf{z}}(\mathbf{A}) \in \mathcal{X}(i^\dagger)$ *for any* $\mathbf{A} = (\mathbf{w}, i^\dagger)$

(b) $\mathbb{E}_{\mathbf{A}}[\tilde{\mathbf{z}}(\mathbf{A})] \le \boldsymbol{\rho}$

(c) $P_{\mathcal{X}}^\star = \mathbb{E}_{\mathbf{A}}[\mathbf{w} \cdot \tilde{\mathbf{z}}(\mathbf{A})] \ge \dfrac{\mathbb{E}[\mathsf{NMR}(\{\hat{\mathbf{z}}_t\}_{t=1}^T; \alpha)]}{T}$ *for and any matching profile $\{\hat{\mathbf{z}}_t\}_{t=1}^T$ that satisfies*

$$\hat{\mathbf{z}}_t \in \mathcal{Z}(i_t^\dagger) \quad \forall t \in [T], \quad \sum_{t=1}^T \mathbb{1}[i_t^\dagger = 0] z_{t,i} \le \left(c_i - \sum_{t=1}^T \mathbb{1}[i_t^\dagger = i] z_{t,i} \right)_+ \quad \forall i \in [m].$$

Proof of Claim EC.15. Parts (a) and (b) are trivial. The only minor modification of the proof is for part (c). We first recall the definition of $\mathsf{P}_{\mathcal{Z}}^\star$ from line (EC.18) (which allowed for inaction only for free cases). Because of our assumption that $\sum_{i=1}^m \rho_i = 1$, the optimal static control $\mathsf{P}_{\mathcal{Z}}^\star$ matches every case to the actual affiliates, and hence we have $\mathsf{P}_{\mathcal{Z}}^\star = \mathsf{P}_{\mathcal{X}}^\star$. The rest of the proof follows the identical steps taken in the proof of Claim EC.5-(c). □

Using this claim, one can follow the steps from line (EC.13) onward to prove the desired result. Specifically, we replace $\bar{\mathbf{z}}_t$ in line (EC.13) with $\tilde{\mathbf{z}}_t := \tilde{\mathbf{z}}(\mathbf{A}_t)$. and follow the same steps from from line (EC.13) onward. For brevity, this part of the proof is omitted. □

Finally, we also show that, the stopping time of $\overline{\texttt{CA-DL}}$ is near the end of the horizon in expectation.

COROLLARY EC.16. *Under Assumption EC.10 and* $\sum_{i=1}^m \rho_i = 1$, *the stopping time* T_A *(see line (16) for definition) of* $\overline{\texttt{CA-DL}}$ *satisfies* $\mathbb{E}[T_A] \geq T - \mathcal{O}(\sqrt{T})$ *under the stable regime (Definition 1).*

Proof of Corollary EC.16. The proof again follows the identical steps taken for the proof of Proposition EC.11. We only point out a minor change in the proof of Lemma EC.13. We define a static dual problem with type-feasibility set $\mathcal{X}$ as

$$\mathsf{D}_{\mathcal{X}}(\boldsymbol{\phi}) := \mathbb{E}\left[\max_{\mathbf{z}(\mathbf{A})\in\mathcal{X}(i^\dagger)} (\mathbf{w}-\boldsymbol{\phi})\cdot\mathbf{z}(\mathbf{A}) + \boldsymbol{\rho}\cdot\boldsymbol{\phi}\right], \quad \mathsf{D}^\star_{\mathcal{X}} := \min_{\boldsymbol{\phi}\geq\mathbf{0}} \mathsf{D}_{\mathcal{X}}(\boldsymbol{\phi}). \tag{EC.56}$$

By strong duality, we have $\mathsf{P}^\star_{\mathcal{X}} = \mathsf{D}^\star_{\mathcal{X}}$. To completely extend the proof of Lemma EC.13 to $\overline{\texttt{CA-DL}}$, we need to show that there exists $\boldsymbol{\phi}^* \in \arg\min_{\boldsymbol{\phi}\geq\mathbf{0}} \mathsf{D}_{\mathcal{X}}$ such that $\|\boldsymbol{\phi}^*\|_\infty \leq \alpha$ (and the rest of the proof follows the identical steps from line (EC.49) and onwards). We claim that such $\boldsymbol{\phi}^*$ indeed exists. To see this, we first recall the static dual problem under the type-feasibility set $\mathcal{Z}(\cdot)$ from line (EC.20):

$$\mathsf{D}_{\mathcal{Z}}(\boldsymbol{\phi}) := \mathbb{E}\left[\max_{\mathbf{z}(\mathbf{A})\in\mathcal{Z}(i^\dagger)} (\mathbf{w}-\boldsymbol{\phi})\cdot\mathbf{z}(\mathbf{A}) + \boldsymbol{\rho}\cdot\boldsymbol{\phi}\right], \quad \mathsf{D}^\star_{\mathcal{Z}} := \min_{\boldsymbol{\phi}\geq\mathbf{0}} \mathsf{D}_{\mathcal{Z}}(\boldsymbol{\phi}).$$

Let $\boldsymbol{\phi}^* \in \arg\min_{\boldsymbol{\phi}\geq\mathbf{0}} \mathsf{D}_{\mathcal{Z}}$. As we prove in Claim EC.31-(a) (see Section EC.8.2), we have $\|\boldsymbol{\phi}^*\|_\infty \leq \alpha$ under Assumption EC.10. We now observe that

$$\mathsf{D}_{\mathcal{X}}(\boldsymbol{\phi}^*) \overset{\text{(a)}}{\leq} \mathsf{D}_{\mathcal{Z}}(\boldsymbol{\phi}^*) = \mathsf{P}^\star_{\mathcal{Z}} \overset{\text{(b)}}{=} \mathsf{P}^\star_{\mathcal{X}} = \mathsf{D}^\star_{\mathcal{X}} \tag{EC.57}$$

Inequality (a) is because $\mathcal{X}(\cdot) \subseteq \mathcal{Z}(\cdot)$. In equation (b), we used our assumption that $\sum_{i=1}^m \rho_i = 1$ (see the proof of claim EC.15). Because $\mathsf{D}_{\mathcal{X}}(\boldsymbol{\phi}^*) \geq \mathsf{D}^\star_{\mathcal{X}}$ by definition, the above inequality implies that $\mathsf{D}_{\mathcal{X}}(\boldsymbol{\phi}^*) = \mathsf{D}^\star_{\mathcal{X}}$, or $\boldsymbol{\phi}^* \in \arg\min_{\boldsymbol{\phi}\geq\mathbf{0}} \mathsf{D}_{\mathcal{X}}$ equivalently. This proves the desired claim. □

EC.6.6. Proof of Corollary 1

Our first step is to show that whenever the backlog is above $1/\zeta$, we have a constant negative drift on the expected backlog.

CLAIM EC.17. *Let* $t \leq T_A$. *Whenever* $b_{t-1,i} \geq 1/\zeta$, *we have* $\mathbb{E}[b_{t,i} - b_{t-1,i} | \mathcal{H}_{t-1}] \leq -(\rho_i - \mathbb{P}[i_t^\dagger = i])$

Proof of Claim EC.17. From Proposition 2, we assume that $\gamma = o(T)$ without loss. Because $1/\zeta = \sqrt{T/\gamma}$ is super-constant under this assumption, we also have $1/\zeta > 1$ without loss. Then, $b_{t,i} - b_{t-1,i} = z_{t,i} - s_{t,i}$ whenever $b_{t-1,i} \geq 1/\zeta$. Furthermore, we can write $z_{t,i} = \mathbb{1}[i_t^\dagger = i]$ whenever $b_{t-1,i} \geq 1/\zeta$. To see this, whenever $i_t^\dagger = 0$ (a free case) and $b_{t-1,i} \geq 1/\zeta$, we observe that $w_{t,i} - \theta_{t,i} - \lambda_{t,i} - \zeta b_{t-1,i} < 0$. Combining this with the fact that the adjusted score of the dummy affiliate is always zero (by

definition), we have $z_{t,i}=0$ whenever $b_{t-1,i}>1/\zeta$. On the other hand, if the arrival t is a tied case to location $j\neq i$, we again have $z_{t,i}=0$. Combining, we have

$$\mathbb{E}[b_{t,i}-b_{t-1,i}|H_{t-1}]=\mathbb{E}[z_{t,i}-s_{t,i}|H_{t-1}]=\mathbb{P}[i_t^\dagger=i]-\rho_i-\epsilon.$$

The proof is complete because $\epsilon\geq 0$. □

We now prove the following lemma, which establishes $\mathcal{O}\left(\frac{1}{\zeta}+1\right)$ bound on the average backlog (in expectation) for any $\epsilon\geq 0$.

LEMMA EC.18 **(Upper Bound on Backlog for Algorithm 1).** *For any given $\zeta\geq 0$, the expected average backlog of Algorithm 1 is*

$$\frac{1}{T}\mathbb{E}\Big[\sum_{t=1}^{T}\|\mathbf{b}_t\|_1\Big]\leq\mathcal{O}\Big(\frac{1}{\zeta}+1\Big).$$

Proof of Lemma EC.18. We apply the following result extracted from Wei et al. (2023) and Gupta (2022).

CLAIM EC.19 **(Lemma 5 of Wei et al. (2023)).** *Let $\Psi(t)$ be an $\{\mathcal{H}_t\}$-adapted stochastic process satisfying:*

- *Bounded variation:* $|\Psi(t+1)-\Psi(t)|\leq K$
- *Negative drift:* $\mathbb{E}[\Psi(t+1)-\Psi(t)|\mathcal{H}_t]\leq -d$ *whenever* $\Psi(t)\geq D$
- $\Psi(0)\leq K+D$

Then, we have

$$\mathbb{E}[\Psi(t)]\leq K\left(1+\lceil\frac{D}{K}\rceil\right)+K\left(\frac{K-d}{2d}\right).$$

To apply Claim EC.19 to our setting, let $\Psi(t):=b_{t,i}$ let $d_i=\rho_i-\mathbb{P}[i_t^\dagger=i]$. We recall that $d_i>0$ is a positive constant by our assumption (see Section 2). Furthermore, we have $b_{0,i}=0$ and $|b_{t,i}-b_{t-1,i}|\leq 1$. From Claim EC.17, we have $\mathbb{E}[b_{t,i}-b_{t-1,i}|H_t]\leq -d_i$ whenever $b_{t-1,i}\geq 1/\zeta$. Applying Claim EC.19 with, $K=1$, and $D=1/\zeta$, we have

$$\mathbb{E}[b_{t,i}]\leq 1+\lceil\frac{1}{\zeta}\rceil+\frac{1-d_i}{2d_i}\leq\frac{1}{\zeta}+\frac{1+3d_i}{2d_i}\tag{EC.58}$$

Hence, we have $\mathbb{E}[\|\mathbf{b}_t\|_1]\leq\mathcal{O}\left(\frac{1}{\zeta}+1\right)$ for all $t\leq T_A$. Finally, the average backlog is bounded by

$$\begin{aligned}\frac{1}{T}\mathbb{E}\left[\sum_{t=1}^{T}\|\mathbf{b}_t\|_1\right]&=\frac{1}{T}\mathbb{E}\left[\sum_{t=1}^{T_A}\|\mathbf{b}_t\|_1\right]+\frac{1}{T}\mathbb{E}\left[\sum_{t=T_A+1}^{T}\|\mathbf{b}_t\|_1\right]\\&\leq\mathcal{O}\left(\frac{1}{\zeta}+1\right)\end{aligned}$$

The last inequality is because we have $\mathbb{E}[\|\mathbf{b}_t\|_1]\leq\mathcal{O}\left(\frac{1}{\zeta}+1\right)$ for all $t\leq T_A$ from inequality (EC.58), along with Claim EC.9 (See Appendix EC.6.4). This completes the proof of Lemma EC.18. □

Proof of Corollary 1. We now complete the proof of Corollary 1. For ease of reference, we recall the inequality (20) (see Section 3.3) in the following: for *any* matching profile $\{\hat{\mathbf{z}}_t\}_{t=1}^T$that satisfies $\hat{\mathbf{z}}_t \in \mathcal{Z}(i_t^\dagger)$ for all $t \in [T]$ and (Capacity Feasibility-T) (see Definition 2),

$$\underbrace{\mathbb{E}[\mathsf{NMR}(\{\hat{\mathbf{z}}_t\}_{t=1}^T;\alpha)] - \mathbb{E}[\mathsf{NMR}(\{\mathbf{z}_t\}_{t=1}^T;\alpha)]}_{(\mathsf{A})} + \zeta\epsilon\mathbb{E}\left[\sum_{t=1}^{T_A-1}\|\mathbf{b}_t\|_1\right] + \zeta\mathbb{E}[\psi(\mathbf{b}_{T_A})] \le \mathcal{O}(\sqrt{T}+\zeta T)$$

Note that the above inequality holds for *any* $\epsilon \ge 0$, including the near-critical regime. This implies that the term (A) is still $\mathcal{O}(\sqrt{T}+\zeta T)$. Let $\{\mathbf{z}_t^*\}_{t=1}^T$ denote the the optimal offline solution (Definition 2). By utilizing the same decomposition as line (10), we bound the regret by

$$\begin{aligned}
\mathbb{E}[\mathsf{OPT}(\alpha,\gamma)] - \mathbb{E}[\mathsf{ALG}(\alpha,\gamma)] &\le \underbrace{\mathbb{E}[\mathsf{NMR}(\{\mathbf{z}_t^*\}_{t=1}^T;\alpha)] - \mathbb{E}[\mathsf{NMR}(\{\mathbf{z}_t\}_{t=1}^T;\alpha)]}_{(\mathsf{A})} + \underbrace{\frac{\gamma}{T}\mathbb{E}\left[\sum_{t=1}^T\|\mathbf{b}_t\|_1\right]}_{\text{Lemma EC.18}} \\
&\le \mathcal{O}(\zeta T+\sqrt{T}) + \mathcal{O}\left(\frac{\gamma}{\zeta}+\gamma\right) \\
&\le \mathcal{O}(\sqrt{\gamma T})
\end{aligned}$$

where the last line follows from $\zeta = \Theta(\sqrt{\frac{\gamma}{T}})$ under the near-critical regime.

EC.7. Matching Lower Bounds

We begin by clarifying an important aspect of our upper bound in Theorem 1. While our formal definition of regret compares the online algorithms objective to that of the benchmark $\mathsf{OPT}(\alpha,\gamma)$, the proof in Section 3.3 in fact establishes a stronger result: our regret bound holds even when benchmarked against $\mathsf{OPT}(\alpha,0)$. This is because we explicitly separate the analysis of net matching reward (Lemma 1) and backlog penalty (Lemma 2). In particular, the the net matching reward of `CA-DL` is lower-bounded relative to *any feasible* offline solutionincluding $\mathsf{OPT}(\alpha,0)$, which incurs no backlog cost (Lemma 4). As a result, the bound $O(\sqrt{T}+\gamma/\varepsilon)$ holds against the stronger benchmark $\mathsf{OPT}(\alpha,0)$.

In this section, we show that the our regret upper-bound is asymptotically optimal with respect to $\mathsf{OPT}(\alpha,0)$ by establishing a tight lower bound for any online algorithm's regret. In Section EC.7.1, we show that the regret bound of Algorithm 1 in the stable regime (Theorem 1) is asymptotically optimal with respect to both benchmarks $\mathsf{OPT}(\alpha,0)$ and $\mathsf{OPT}(\alpha,\gamma)$. In Section EC.7.2, we show that the regret bound of Algorithm 1 in the near-critical regime where $\epsilon = O(1/\sqrt{T})$ (Corollary 1) is asymptotically optimal with respect to $\mathsf{OPT}(\alpha,0)$ in the spacial case of $\epsilon = 0$.

Throughout, we let $\mathcal{I}$ denote an instance of our problem (see the paragraph preceding Section 2), and use $\mathsf{ALG}_{\mathcal{I}}(\alpha,\gamma)$ to denote the objective achieved by any online algorithm under instance $\mathcal{I}$, and $\mathsf{OPT}_{\mathcal{I}}(\alpha,\gamma)$ the corresponding offline optimum (Definition 2).

EC.7.1. Lower Bounds for Stable Regime

We first provide a lower bound for regret of any online algorithm against $\mathsf{OPT}(\alpha, 0)$, showing our upper-bound in Theorem 1 is asymptotically optimal in the stable regime. Specifically, the following result demonstrates that both the $\sqrt{T}$ and γ/ε terms are necessary to capture the worst-case regret against $\mathsf{OPT}(\alpha, 0)$.

PROPOSITION EC.20 **(Lower Bound on Regret against $\mathsf{OPT}(\alpha, 0)$).** *In the stable regime (Definition 1), there exist instances for which the regret of any online algorithm with respect to* $\mathsf{OPT}(\alpha, 0)$ *is* $\Omega\left(\sqrt{T} + \frac{\gamma}{\varepsilon}\right)$.

Proof of Proposition EC.20. In the following, we construct two instances, $\mathcal{I}_1$ and $\mathcal{I}_2$, and show that for instance $\mathcal{I}_1$ (resp. $\mathcal{I}_2$), any online algorithm incurs regret of at least $\Omega(\gamma/\varepsilon)$ (resp. $\Omega(\sqrt{T})$). We then establish the claimed lower bound by taking the worst case over the two instances.

(ii) Instance $\mathcal{I}_1$. We consider an instance with one actual affiliate ($m = 1$). The capacity is given by $c = 0.5T$ (i.e., $\rho = 0.5$), and the service rate is $r = 0.5 + \varepsilon$ for positive constant $\epsilon \in (0, 0.5)$. At each time t, an arrival is a tied (resp. free) case with probability $0.5 - \varepsilon$ (resp. $0.5 + \varepsilon$). The reward of all arrivals is one. Note that we have $\mathsf{OPT}_{\mathcal{I}_1}(\alpha, 0) = 0.5T$. This is because the offline algorithm accepts all tied cases (as required) and fills the remaining capacity using free caseswithout incurring any congestion penalty.[41]

We now upper bound the objective of any online algorithm. First, observe that any online algorithm must accept all tied cases, which arrive at rate $0.5 - \varepsilon$. Since the service rate is $0.5 + \varepsilon$, this creates a discrete-time single-server queue with arrival probability $\lambda = 0.5 - \varepsilon$ and service probability $\mu = 0.5 + \varepsilon$. The resulting backlog process induced by tied cases evolves as a reflected random walk on the non-negative integers. This Markov chain is ergodic (i.e., irreducible and aperiodic) and admits a well-defined stationary distribution. In steady state, one can show that the expected backlog is $\Theta(1/\varepsilon)$. By the ergodicity of the chain, the time-average backlog converges to this stationary expectation (Gross et al. 2011), implying that the expected backlog over a finite horizon T is $\Theta(1/\varepsilon) + o(1)$. Because accepting free cases can only increase the backlog, it follows that the expected average backlog of any online algorithm is at least $\Theta(1/\varepsilon)$. Combining this with the upper bound on achievable reward $(0.5T)$, we conclude $\mathbb{E}[\mathsf{ALG}_{\mathcal{I}_1}(\alpha, \gamma)] \le 0.5T - \Theta\left(\frac{\gamma}{\varepsilon}\right)$, which implies

$$\mathbb{E}[\mathsf{OPT}_{\mathcal{I}_1}(\alpha, 0) - \mathsf{ALG}_{\mathcal{I}_1}(\alpha, \gamma)] \ge \Omega(\gamma/\varepsilon). \tag{EC.59}$$

(ii) Instance $\mathcal{I}_2$. This instance is adapted from Freund and Banerjee (2019), with a single affiliate ($m = 1$) and all arrivals being free cases ($i_t^\dagger = 0$ for all $t \in [T]$). The total capacity is set to $c = 0.5T$.

[41] Because $\epsilon = \Omega(1)$ in the stable regime and the arrival rate of tied cases is $\rho - \epsilon$ in this instance, there is no over-allocation cost with high probability.

At each time t, an arrival type is given by $\mathbf{A}_t = (w_t, 0)$ with the reward $w_t \in \{1/3, 2/3, 1\}$ being drawn independently with probabilities:

$$\Pr[w_t = 1] = \frac{1}{2} - \frac{1}{\sqrt{T}}, \quad \Pr[w_t = 2/3] = \frac{1}{\sqrt{T}}, \quad \Pr[w_t = 1/3] = \frac{1}{2}. \tag{EC.60}$$

Freund and Banerjee (2019) analyzes a setting without congestion cost and show that, in this instance, no online algorithm can achieve reward within $\mathcal{O}(\sqrt{T})$ of $\mathsf{OPT}_{\mathcal{I}_2}(\alpha, 0)$. That is, when the objective consists solely of cumulative reward (i.e., $\gamma = 0$), we have the following:

LEMMA EC.21 **(Proposition 4 of Freund and Banerjee (2019)).** *In instance $\mathcal{I}_2$, no online algorithm can achieve $\mathcal{O}(\sqrt{T})$ regret relative to $\mathsf{OPT}_{\mathcal{I}_2}(\alpha, 0)$.*[42] *In particular, for any online algorithm,*

$$\mathbb{E}[\mathsf{OPT}_{\mathcal{I}_2}(\alpha, 0) - \mathsf{ALG}_{\mathcal{I}_2}(\alpha, 0)] \geq \Omega(\sqrt{T}). \tag{EC.61}$$

Since $\mathsf{ALG}_{\mathcal{I}_2}(\alpha, \gamma) \leq \mathsf{ALG}_{\mathcal{I}_2}(\alpha, 0)$ for any $\gamma \geq 0$ (because congestion cost only reduces the objective), it follows that the same lower bound holds when the algorithms objective includes a congestion penalty. That is,

$$\mathbb{E}[\mathsf{OPT}_{\mathcal{I}_2}(\alpha, 0) - \mathsf{ALG}_{\mathcal{I}_2}(\alpha, \gamma)] \geq \Omega(\sqrt{T}). \tag{EC.62}$$

Thus, combining (EC.59) and (EC.62), the worst-case regret of any online algorithm against $\mathsf{OPT}(\alpha, 0)$ is $\Omega\left(\max\left\{\sqrt{T}, \frac{\gamma}{\varepsilon}\right\}\right) = \Omega\left(\sqrt{T} + \frac{\gamma}{\varepsilon}\right)$. □

We now turn to our original benchmark $\mathsf{OPT}(\alpha, \gamma)$ (that penalizes the objective by the congestion cost). The following result provides a lower bound that matches the regret upper bound with respect to $\mathsf{OPT}(\alpha, \gamma)$ in the stable regime ($\epsilon = \Omega(1)$)(Theorem 1).

PROPOSITION EC.22 **(Lower Bound on Regret against $\mathsf{OPT}(\alpha, \gamma)$).** *In the stable regime (Definition 1), there exist instances for which the regret (Definition 3) of any online algorithm with respect to $\mathsf{OPT}(\alpha, \gamma)$ is $\Omega\left(\sqrt{T} + \gamma\right)$.*

Proof of Proposition EC.22. We construct two instances, $\mathcal{I}_2$ and $\mathcal{I}_3$, and show that for instance $\mathcal{I}_2$ (resp. $\mathcal{I}_3$), any online algorithm incurs regret against $\mathsf{OPT}(\alpha, \gamma)$ of at least $\Omega\left(\sqrt{T} - \gamma\right)$ (resp. $\Omega(\gamma)$). Combining these two bounds yields the desired lower bound on the regret.

(i) Instance $\mathcal{I}_2$. We revisit instance $\mathcal{I}_2$ from the proof of Proposition EC.20. We begin by upper-bounding the gap between $\mathsf{OPT}_{\mathcal{I}_2}(\alpha, \gamma)$ and $\mathsf{OPT}_{\mathcal{I}_2}(\alpha, 0)$. Consider the optimal solution to $\mathsf{OPT}_{\mathcal{I}_2}(\alpha, 0)$, which maximizes total reward without penalizing backlog. With a slight abuse of notation, we use $\mathsf{OPT}_{\mathcal{I}_2}(\alpha, 0)$ denote the total reward earned by this solution. We note that the optimal solution to $\mathsf{OPT}_{\mathcal{I}_2}(\alpha, 0)$ is feasible for $\mathsf{OPT}_{\mathcal{I}_2}(\alpha, \gamma)$. We now lower bound the expected time-average backlog incurred by the solution to $\mathsf{OPT}_{\mathcal{I}_2}(\alpha, 0)$. Let $T_o := \sup\{t \in [T] : \sum_{\tau=1}^{t} \mathbb{1}[w_\tau = 1] \leq 0.5T\}$ be the first

[42] Although this instance does not involve tied arrivalsand hence α plays no meaningful rolewe retain the notation $\mathsf{OPT}(\alpha, \gamma)$ for consistency.

time at which $0.5T$ reward-1 arrivals have occurred. By construction, $T_o \geq 0.5T$ on every sample path. The solution to $\mathsf{OPT}(\alpha, 0)$ maximizes total reward by accepting reward-1 arrivals whenever it can, and thus it accepts all reward-1 arrivals that occur before time T_o. Furthermore, for large enough $T \geq \Omega(1/\epsilon^2)$, the arrival rate of reward-1 case is at least $0.5-\epsilon$. Consequently, during time $[1, T_o]$, the backlog process $\{b_t\}$ induced by this solution evolves as a reflected birthdeath process with arrival rate at least $0.5-\epsilon$ and service rate $0.5+\epsilon$. Following the similar argument of the analysis of instance $\mathcal{I}_1$, the expected time-average backlog up to T_o satisfies

$$\mathbb{E}\left[\frac{1}{T_o}\sum_{t=1}^{T_o} b_t\right] \geq \Theta(1/\epsilon).$$

Since $T_o \in [0.5T, T]$ for all sample paths, we have

$$\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T} b_t\right] \geq \mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T_o} b_t\right] = \mathbb{E}\left[\frac{T_o}{T}\cdot\left(\frac{1}{T_o}\sum_{t=1}^{T_o} b_t\right)\right] \geq \frac{1}{2}\cdot\mathbb{E}\left[\frac{1}{T_o}\sum_{t=1}^{T_o} b_t\right] \geq \Theta(1/\epsilon),$$

which gives a valid lower bound on the expected time-average backlog incurred by $\mathsf{OPT}_{\mathcal{I}_2}(\alpha, 0)$ over the entire horizon. It follows that, when evaluated under the objective with congestion penalty γ, we have:

$$\mathbb{E}[\mathsf{OPT}_{\mathcal{I}_2}(\alpha,\gamma)] \geq \mathbb{E}[\mathsf{OPT}_{\mathcal{I}_2}(\alpha,0)] - \Theta\left(\frac{\gamma}{\varepsilon}\right). \tag{EC.63}$$

Combining the lower bound in (EC.62) with the inequality (EC.63), we obtain:

$$\begin{aligned}\mathbb{E}[\mathsf{OPT}_{\mathcal{I}_2}(\alpha,\gamma) - \mathsf{ALG}_{\mathcal{I}_2}(\alpha,\gamma)] &= \mathbb{E}[\mathsf{OPT}_{\mathcal{I}_2}(\alpha,0) - \mathsf{ALG}_{\mathcal{I}_2}(\alpha,\gamma)] - \mathbb{E}[\mathsf{OPT}_{\mathcal{I}_2}(\alpha,0) - \mathsf{OPT}_{\mathcal{I}_2}(\alpha,\gamma)] \\ &\geq \Omega\left(\sqrt{T} - \frac{\gamma}{\varepsilon}\right).\end{aligned} \tag{EC.64}$$

(iii) Instance $\mathcal{I}_3$. We revisit the instance considered in Proposition 2, where there is a single affiliate ($m = 1$) with capacity $c = 0.5T$, and all arrivals are free cases with deterministic reward of one. Following an identical argument to that in Proposition 2, one can show that, under the stable regime, any online algorithm must satisfy:

$$\mathbb{E}[\mathsf{OPT}_{\mathcal{I}_3}(\alpha,\gamma) - \mathsf{ALG}_{\mathcal{I}_3}(\alpha,\gamma)] \geq \Omega(\gamma). \tag{EC.65}$$

Taking the maximum of the lower bounds in (EC.64) and (EC.65) and assuming $\epsilon = \Omega(1)$, the worst-case regret of any online algorithm against $\mathsf{OPT}(\alpha,\gamma)$ is at least $\Omega\left(\max\left\{\sqrt{T}-\gamma, \gamma\right\}\right)$. Consider two cases. If $\gamma = o(\sqrt{T})$, then the bound reduces to $\Omega(\sqrt{T})$. Otherwise, if $\gamma = \Omega(\sqrt{T})$, it becomes $\Omega(\gamma)$. Thus, in combining cases, the regret against $\mathsf{OPT}(\alpha,\gamma)$ is $\Omega\left(\max\{\sqrt{T}, \gamma\}\right) = \Omega(\sqrt{T}+\gamma)$. This completes the proof. □

EC.7.2. Lower Bounds for Near-Critical Regime

In the near-critical regime where $\epsilon = \mathcal{O}(1/\sqrt{T})$, the regret upper bound of Algorithm 1 is given by $\mathcal{O}(\sqrt{\gamma T})$ (see Corollary 1), even when compared against the stronger benchmark $\mathsf{OPT}(\alpha, 0)$. In the following proposition, we show that this regret bound is tight with respect to $\mathsf{OPT}(\alpha, 0)$ in the special case where $\epsilon = 0$.

PROPOSITION EC.23. *For $\epsilon = 0$, there exists an instance for which the regret of any online algorithm with respect to $\mathsf{OPT}(\alpha, 0)$ is $\Omega(\sqrt{\gamma T})$.*

The rest of this section is devoted to proving Proposition EC.23. For the near-critical regime, establishing the lower bound even in this special case of $\epsilon = 0$ turns out to be technically challenging due to the difficulty of characterizing a (non-stationary) finite-horizon optimal online algorithm. Consequently, our proof introduces several novel techniques: (i) characterizing the optimal policy of an auxiliary infinite-horizon Markov decision process tailored to our instance, and (ii) establishing the convergence rate of the finite-horizon optimal policy to its infinite-horizon counterpart.

Proof of Proposition EC.23. We analyze the following instance $\mathcal{I}_4$.

Instance $\mathcal{I}_4$. We consider a single-affiliate setting where all arrivals are free cases, and the reward $w_t \in \{0, 1\}$ of each arrival is independently drawn from a Bernoulli distribution with $\Pr(w_t = 1) = 0.5$. We often write $\mathsf{Bernoulli}(0.5)$ to denote this distribution. We set the total capacity to $c = 0.5T$. The service rate is exactly $r = 0.5$ (thus $\epsilon = 0$). Because there are no tied cases, the parameter α plays no role in the current instance. Hence, we simplify notation by writing $\mathsf{OPT}(0)$ to denote $\mathsf{OPT}(\alpha, 0)$i.e., the offline benchmark that maximizes total reward subject to capacity, with no congestion penalty.

In this instance, note that $\mathsf{OPT}(0)$ accepts all of reward-1 arrivals (and reject the others) to fill the capacity. Thus, the value of $\mathsf{OPT}(0)$ on instance $\mathcal{I}_4$ is:

$$\mathbb{E}[\mathsf{OPT}_{\mathcal{I}_4}(0)] = \mathbb{E}\left[\min\left(0.5T, \sum_{t=1}^{T} \mathbb{1}[w_t = 1]\right)\right] = 0.5T - \Theta(\sqrt{T}). \tag{EC.66}$$

To analyze the regret of any online algorithm, we examine the optimal online policy $\mathsf{DP}_T(\gamma)$, which maximizes (Objective)the cumulative reward minus the congestion costover a horizon of T periods, given a congestion penalty γ. This policy can be formulated as a finite-horizon Markov Decision Process (MDP), which we define formally in Section EC.7.2.1. The following lemma provides an upper bound on the expected value of $\mathsf{DP}_T(\gamma)$.

LEMMA EC.24. $\mathbb{E}[\mathsf{DP}_T(\gamma)] \le 0.5T - \Theta(\sqrt{\gamma T})$,

Combined with Equation (EC.66), Lemma EC.24 directly implies Proposition EC.23. The proof of Lemma EC.24 is fairly intricate; here, we provide a heuristic argument and defer the full proof to the next subsection. Suppose, for the sake of developing intuition, that $\mathsf{DP}_T(\gamma)$ can be well approximated by the following stationary threshold policy: accept a reward-1 arrival if and only if the current backlog

$b_{t-1} \le K-1$, for some fixed threshold $K > 0$, which we will optimize later. Under this policy, the backlog evolves similar to an $M/M/1/K$ queue with both arrival and service rates equal to 0.5. The stationary distribution of backlog of such $M/M/1/K$ queue is uniform over $\{0,\dots,K\}$, leading to a rejection probability of $\Theta(1/K)$ and an average backlog of $\Theta(K)$. The expected objective of this threshold policy can then be roughly given by:

$$0.5T \cdot \left(1 - \Theta\left(\frac{1}{K}\right)\right) - \gamma \cdot \Theta(K).$$

where the first (resp. second) term is the matching rewards (resp. congestion cost). Optimizing over K yields the choice $K^* = \Theta(\sqrt{T/\gamma})$. With this choice of K^*, the resulting value of the optimum online is $\mathbb{E}[\mathsf{DP}_T(\gamma)] \approx 0.5T - \Theta(\sqrt{\gamma T})$, as in Lemma EC.24.

In Section EC.7.2.1, we formalize the above intuition by showing that (i) the horizon-optimal online policy $\mathsf{DP}_T(\gamma)$ can indeed be well approximated by a threshold-form stationary policy with an additive gap of $O(\sqrt{T})$, and (ii) the value of such a stationary policy is $0.5T - \Theta(\sqrt{\gamma T})$.

EC.7.2.1. Proof of Lemma EC.24

For the proof, we first formally define the optimal online policy $\mathsf{DP}_T(\gamma)$, which provides a valid upper bound on the objective value (Objective) for any online algorithm. Analyzing the value of $\mathsf{DP}_T(\gamma)$ directly is challenging, as the finite-horizon optimal policy is generally non-stationary. To overcome this, we define an auxiliary infinite-horizon problem closely related to $\mathsf{DP}_T(\gamma)$. We then characterize the optimal stationary policy and its value for the infinite-horizon problem, and connect this value back to that of $\mathsf{DP}_T(\gamma)$. We elaborate each step in the following.

Finite-Horizon Optimal Online Algorithm. We define $\mathsf{DP}_T(\gamma)$ as a dynamic program that maximizes the total objective over T periods (reward minus congestion cost), *ignoring the capacity constraint*. Note that the objective value of $\mathsf{DP}_T(\gamma)$ is a valid upper bound on the objective value of any feasible online algorithm (which must respect the capacity constraint). At each time t, the state of $\mathsf{DP}_T(\gamma)$ is the current backlog b_{t-1}. We denote the horizon-optimal policy by

$$\hat{\pi}_T = (\pi_1, \pi_2, \dots, \pi_T),$$

where $\pi_t(b) \in \{0,1\}$ represents the decision to match an arrival at time t when the backlog is b.

At each time t, let $w_t \sim \mathsf{Bernoulli}(0.5)$ denote the reward of the arriving case. Technically, the matching decision depends on both the backlog and the realized reward w_t. However, it is without loss of optimality to reject any arrival with $w_t = 0$, since matching it incurs congestion cost without increasing the matching reward. To simplify notation, we thus adopt the convention that

$$\pi_t(b_{t-1}) := 0 \quad \text{whenever } w_t = 0.$$

That is, we implicitly restrict attention to reward-1 arrivals, and interpret $\pi_t(b_{t-1})$ as the decision to match such an arrival when the backlog is b_{t-1}. Under this convention, we define the *per-period payoff* function as follows:

$$r(\pi_t(b_{t-1}), b_{t-1}) := \pi_t(b_{t-1}) - \frac{\gamma}{T} b_{t-1}. \tag{EC.67}$$

The total expected value of the finite-horizon dynamic program when starting from backlog $b_0 = 0$ is:

$$\mathbb{E}[\mathsf{DP}_T(\gamma)] := \mathbb{E}\left[\sum_{t=1}^{T} r(\pi_t(b_{t-1}), b_{t-1}) \,\middle|\, b_0 = 0\right], \tag{EC.68}$$

where the expectation is taken over the sequence of arrivals $\{w_t\}_{t=1}^T$ and backlog states $(b_t)_{t=1}^T$ induced by the policy $\hat{\pi}_T$.

Auxiliary Infinite-Horizon MDP. We now define a parallel infinite-horizon average-reward MDP as an auxiliary problem. This MDP ignores capacity constraints and uses the same per-period congestion penalty γ/T as in the finite-horizon problem. Importantly, both γ and T are fixed in this formulation; we consider running the MDP for a growing horizon $H \to \infty$ while keeping γ/T constant. This will allow us to analyze the long-run behavior of stationary policies under the same trade-off between matching reward and congestion penalty as in the original finite-horizon setting.

Let $\pi^*\colon \mathbb{N} \to \{0, 1\}$ denote a stationary policy that maps the current backlog b to a matching decision. As before, we assume without loss of generality that the policy only acts on arrivals with $w_t = 1$, and we suppress the dependence on w_t in notation. Using the payoff function from (EC.67), we define an infinite-horizon optimal stationary policy as

$$\pi^* \in \arg\max_{\pi} \limsup_{H\to\infty} \frac{1}{H} \mathbb{E}\left[\sum_{t=1}^{H} r(\pi(b_{t-1}), b_{t-1})\right],$$

and its long-run average payoff is denoted by:

$$\rho^* := \lim_{H\to\infty} \frac{1}{H} \mathbb{E}\left[\sum_{t=1}^{H} r(\pi^*(b_{t-1}), b_{t-1})\right],$$

where the expectation is taken over the stationary distribution of the backlog b_{t-1} induced by π^* and the i.i.d. arrival rewards $w_t \sim \mathsf{Bernoulli}(0.5)$.

We now state two auxiliary results from infinite-horizon average-reward MDP theory (Puterman 1994) that are instrumental in our analysis.

First, we invoke an optimality condition for infinite-horizon problems, which characterizes the optimal stationary policy via its long-run average payoff and associated bias function. This result will help bridge the finite-horizon optimal value $\mathsf{DP}_T(\gamma)$ with the performance of a stationary policy in the associated infinite-horizon problem.

DEFINITION EC.25 **(Equation (8.2.3) in Puterman (1994), Bias function).** *Let π^* be the infinite-horizon optimal stationary policy with long-run average payoff ρ^*. The bias function $h\colon \mathbb{N} \to \mathbb{R}$ is defined by:*

$$h(b) := \sum_{t=1}^{\infty} \mathbb{E}\left[r(\pi^*(b_{t-1}), b_{t-1}) - \rho^* \,\middle|\, b_0 = b \right],$$

where $r(\cdot,\cdot)$ is the per-period payoff function defined in (EC.67). The expectation is taken over the backlog trajectory induced by π^*, starting from backlog b, and an i.i.d. arrival sequence $\{w_t\}_{t=1}^{\infty}$.

LEMMA EC.26 **(Equation (8.4.2) in Puterman (1994), Bias-gain optimality condition).** *Let π^* be the optimal stationary policy for the infinite-horizon problem with long-run payoff ρ^* and bias function $h(\cdot)$. Then π^* satisfies the following optimality condition:*

$$\rho^* + h(b) = \max_{z \in \{0,1\}} \{ r(z, b) + \mathbb{E}\left[h(b')\right] \},$$

for all non-negative integers b. Here, $b' = (b + \pi^*(b) - s)_+$ denotes the next backlog starting from b, where $s \sim \mathsf{Bernoulli}(0.5)$ represents the random service. The expectation $\mathbb{E}[h(b')]$ is taken with respect to the random arrival $w \sim \mathsf{Bernoulli}(0.5)$ and service $s \sim \mathsf{Bernoulli}(0.5)$, and the resulting next state $b' = (b + \pi^*(b) - s)_+$ (induced by applying policy π^* at state b when $w = 1$).

Furthermore, the following lemma shows that optimal infinite-horizon policy admits a simple threshold structure.

LEMMA EC.27 **(Yechiali (1971)).** *The infinite-horizon optimal policy π^* is of threshold type: there exists threshold K^* such that $\pi^*(b) = 1$ if and only if $b < K^*$.*

Proof of Lemma EC.24. We are now ready to prove Lemma EC.24 in four steps. The proof of all auxiliary claims is deferred to the end of this section.

Step 1: Upper bound via bias decomposition.

We begin by upper bounding the finite-horizon value $\mathsf{DP}_T(\gamma)$ using the bias function $h(\cdot)$ and long-run average payoff ρ^* from the infinite-horizon MDP. Recall $\hat{\pi}_T = (\pi_1, \ldots, \pi_T)$ be the finite-horizon optimal policy. Let $\hat{b}_0 = 0$ be the initial backlog. Let $\hat{b}_1, \ldots, \hat{b}_T$ denote the resulting backlog process under $\hat{\pi}_T$.

CLAIM EC.28. *For all T and γ, we have:*

$$\mathbb{E}[\mathsf{DP}_T(\gamma)] \le T\rho^* + h(0) - \mathbb{E}[h(\hat{b}_T)]. \tag{EC.69}$$

The proof follows by applying the biasgain optimality condition (Lemma EC.26) recursively along the backlog trajectory induced by the finite-horizon optimal policy $\hat{\pi}_T$.

Step 2: Asymptotic expression for ρ^*. We now analyze the long-run average payoff of the infinite-horizon MDP under the threshold policy π^*.

Claim EC.29. *The long-run average payoff under the infinite-horizon optimal policy satisfies:*

$$\rho^* = 0.5 - \Theta\left(\sqrt{\frac{\gamma}{T}}\right),$$

and the optimal stationary policy uses threshold $K^* = \Theta\left(\sqrt{T/\gamma}\right)$.

To prove Claim EC.29, we crucially rely on the threshold structure of the infinite-horizon optimal policy (Lemma EC.27) and optimize the backlog threshold K by analyzing the stationary distribution induced by the corresponding threshold policy.

Step 3: Bounding the Bias Magnitude. We now upper bound the bias term $h(0) - \mathbb{E}[h(\hat{b}_T)]$ appearing in (EC.69).

Claim EC.30. *For any fixed* γ *and* T*, the bias gap satisfies:*

$$|\mathbb{E}[h(0) - h(\hat{b}_T)]| \le \mathcal{O}\left(\sqrt{T}\right).$$

The proof of Claim EC.30 involves several technical steps. In particular, we leverage the fact that the backlog process induced by the infinite-horizon optimal policy (characterized in Claim EC.29) mixes "fast" to bound the bias term $\mathbb{E}[h(\hat{b}_T)]$ in terms of $\mathbb{E}[\hat{b}_T]$, and analyze the worst-case expected backlog using a result on the running maximum of a random walk (Comtet and Majumdar 2005).

Step 4: Putting everything together. From Claim EC.28, $\mathbb{E}[\mathsf{DP}_T(\gamma)] \le T \cdot \rho^* + h(0) - \mathbb{E}[h(\hat{b}_{T+1})]$. Claim EC.29 gives $\rho^* = 0.5 - \Theta(\sqrt{\gamma/T})$, and Claim EC.30 shows $|h(0) - \mathbb{E}[h(\hat{b}_{T+1})]| \le \mathcal{O}(\sqrt{T})$. Substituting,

$$\mathbb{E}[DP_T(\gamma)] \le T\left(0.5 - \Theta(\sqrt{\gamma/T})\right) + \mathcal{O}(\sqrt{T}) = 0.5T - \Theta(\sqrt{\gamma T}).$$

Thus $\mathbb{E}[DP_T(\gamma)] \le 0.5T - \Theta(\sqrt{\gamma T})$, as desired. □

Proof of Claim EC.28. By the bias-gain optimality condition (Lemma EC.26), we have, for all $t = 1, \ldots, T$,

$$r(\pi_t(\hat{b}_{t-1}), \hat{b}_{t-1}) \le \rho^* + h(\hat{b}_{t-1}) - \mathbb{E}[h(\hat{b}_t)].$$

Summing both sides over $t = 1$ to T, we obtain:

$$\sum_{t=1}^{T} r(\pi_t(\hat{b}_{t-1}), \hat{b}_{t-1}) \le T\rho^* + h(\hat{b}_0) - \mathbb{E}[h(\hat{b}_T)].$$

Taking expectations and using $\hat{b}_0 = 0$, we conclude:

$$\mathbb{E}\left[\sum_{t=1}^{T} r(\pi_t(\hat{b}_{t-1}), \hat{b}_{t-1})\right] \le T\rho^* + \mathbb{E}[h(0) - h(\hat{b}_T)],$$

Because the left-hand side is equal to $\mathbb{E}[\mathsf{DP}_T(\gamma)]$ by equation (EC.68), this completes the proof. □

Proof of Claim EC.29. We identify the threshold K^* used by the optimal stationary policy π^*, and compute the corresponding average payoff ρ^*. From Lemma EC.27, policy π^* accepts an arrival

if and only if the backlog $b < K$, for some threshold $K \in \mathbb{N}$. Under such a policy, the backlog process forms a lazy random walk on $\{0, 1, \ldots, K\}$.[43] Let b^* denote the stationary backlog under this policy. The stationary distribution (and resulting expected value) of this random walk is given by (Levin and Peres 2017):

$$\mathbb{P}(b^* = K) = \Theta\left(\frac{1}{K}\right), \qquad \mathbb{E}[b^*] = \Theta(K).$$

The resulting long-run average payoff is:

$$\rho^* = \underbrace{0.5 \cdot \mathbb{P}(b^* < K)}_{\text{match reward}} - \underbrace{\frac{\gamma}{T} \cdot \mathbb{E}[b^*]}_{\text{congestion penalty}}.$$

Since $\mathbb{P}(b^* < K) = 1 - \mathbb{P}(b^* = K) = 1 - \Theta(1/K)$, we have:

$$\rho^* = 0.5 - \Theta\left(\frac{1}{K}\right) - \Theta\left(\frac{\gamma}{T} \cdot K\right).$$

Optimizing over K, we conclude that the optimal the optimal threshold K^* must be given by $K^* = \Theta\left(\sqrt{\frac{T}{\gamma}}\right)$. Substituting this into the expression for ρ^*, we get:

$$\rho^* = 0.5 - \Theta\left(\sqrt{\frac{\gamma}{T}}\right),$$

as claimed. □

Proof of Claim EC.30. By triangular inequality and Jensen's inequality, we have

$$|\mathbb{E}[h(0) - h(\hat{b}_T)]| \le \mathbb{E}[|h(0)|] + \mathbb{E}[|h(\hat{b}_T)|]. \tag{EC.70}$$

In the following, we focus on showing that

$$\mathbb{E}[|h(\hat{b}_T)|] \le \mathcal{O}(\sqrt{T}). \tag{EC.71}$$

The argument for $\mathbb{E}[|h(0)|]$ follows the identical steps and is omitted for brevity.

To prove inequality (EC.72), we first note that, as shown in Claim EC.29, the Markov chain $\{b_t^*\}$ induced by the stationary policy π^* is a lazy reflected random walk on the path $\{0, 1, \ldots, K^*\}$. This chain is irreducible and aperiodic, and hence admits a unique stationary distribution μ. By Theorem 4.9 of Levin and Peres (2017), the chain satisfies geometric convergence: there exists a rate parameter $\alpha \in (0, 1)$ such that for any initial state b and any $t \ge 0$, the total variation distance satisfies

$$\|P^t(b, \cdot) - \mu(\cdot)\|_{\text{TV}} \le \mathcal{O}(\alpha^t), \tag{EC.72}$$

[43] Under the threshold policy π^*, the backlog process evolves as $b_t = (b_{t-1} + \pi^*(b_{t-1}) \cdot w_t - s_t)_+$, where $w_t, s_t \sim$ Bernoulli(0.5) are independent reward-1 arrival and service indicators. For internal states $1 \le b < K$, the transition probabilities satisfy:

$$P(b \to b+1) = 0.25, \quad P(b \to b-1) = 0.25, \quad P(b \to b) = 0.5.$$

At the boundaries: $P(0 \to 0) = 0.75, P(0 \to 1) = 0.25$, and $P(K \to K) = 0.5, P(K \to K-1) = 0.5$. This defines a lazy, reflected random walk on $\{0, 1, \ldots, K\}$.

where $P^t(b,\cdot)$ denotes the distribution of b_t^* conditional on $b_0^* = b$, and $\|\cdot\|_{\text{TV}}$ denotes total variation distance.[44] Furthermore, it is known that the convergence rate α is determined by the spectral gap κ of the chain, with $\alpha = 1-\kappa$ (see, for example, chapter 12 of Levin and Peres (2017)). Since $\{b_t^*\}$ is a lazy random walk on a path of length K^*, it follows from Exercise 13.2 in Levin and Peres (2017) that $\kappa = \Theta((K^*)^{-2})$, and hence $\alpha = 1-\Theta((K^*)^{-2})$.

Building on the above result, we now establish an upper bound on $|h(\hat{b}_T)|$. Recall from Definition EC.25 that the bias function is given by

$$h(\hat{b}_T) := \sum_{t=1}^{\infty} \left(\mathbb{E}[r(\pi^*(b_{t-1}^*), b_{t-1}^*) \mid b_0^* = \hat{b}_T] - \rho^* \right).$$

Define $\tilde{r}(k) := \pi^*(k) - \frac{\gamma}{T}\cdot k$, so that

$$\mathbb{E}[r(\pi^*(b_{t-1}^*), b_{t-1}^*) \mid b_0^* = \hat{b}_T] = \sum_{k=0}^{\max(K^*,\hat{b}_T)} \tilde{r}(k)\cdot P^{t-1}(\hat{b}_T, k), \tag{EC.73}$$

$$\rho^* = \sum_{k=0}^{K^*} \tilde{r}(k)\cdot\mu(k) \le \sum_{k=1}^{\max(K^*,\hat{b}_T)} \tilde{r}(k)\cdot\mu(k). \tag{EC.74}$$

To see why equation (EC.73) holds, we recall that the infinite-horizon optimal policy π^* accepts a reward-1 arrival if and only if the backlog is strictly less than K^*. Thus, if the initial state is $\hat{b}_T > K^*$, the backlog under π^* can only decrease and will eventually enter and remain in $\{0,\dots,K^*\}$. Hence, the payoff at time t only depend on the values $k \le \max(K^*, \hat{b}_T)$.

From equations (EC.73) and (EC.74), we have:

$$\left|\mathbb{E}[r(\pi^*(b_{t-1}^*), b_{t-1}^*) \mid b_0^* = \hat{b}_T] - \rho^*\right| \le \sum_{k=0}^{\max(K^*,\hat{b}_T)} |\tilde{r}(k)|\cdot|P^{t-1}(b,k) - \mu(k)| \tag{EC.75}$$

$$\le \left(\max_{0\le k\le \max(K^*,\hat{b}_T)} |\tilde{r}(k)|\right)\cdot 2\cdot \|P^{t-1}(b,\cdot) - \mu\|_{\text{TV}}. \tag{EC.76}$$

Note that $|\tilde{r}(k)| \le 1 + \frac{\gamma}{T}k = \mathcal{O}\left(\frac{\gamma}{T}k\right)$. Combining this with (EC.72) and (EC.76), we obtain:

$$\left|\mathbb{E}[r(\pi^*(b_{t-1}^*) \mid b_0^* = \hat{b}_T] - \rho^*\right| \le \mathcal{O}\left(\frac{\gamma}{T}\cdot\max(\hat{b}_T, K^*)\cdot\alpha^{t-1}\right),$$

and thus

$$|h(\hat{b}_T)| \le \mathcal{O}\left(\frac{\gamma}{T}\cdot\max(\hat{b}_T,K^*)\cdot\sum_{t=1}^{\infty}\alpha^{t-1}\right) = \mathcal{O}\left(\frac{\gamma}{T}\cdot\max(\hat{b}_T,K^*)\cdot\frac{1}{1-\alpha}\right) \tag{EC.77}$$

Recall that $\alpha = 1 - O((K^*)^{-2})$. Furthermore, from Claim EC.29, we have $K^* = \sqrt{T/\gamma}$. Plugging this into the above bound yields:

$$|h(\hat{b}_T)| \le \mathcal{O}(\max(\hat{b}_T, K^*)) \le \mathcal{O}(\hat{b}_T + K^*). \tag{EC.78}$$

[44] For probability measures ν, μ on a state space Ω, the total variation distance is defined as:

$$\|\nu - \mu\|_{\text{TV}} := \frac{1}{2}\sum_{x\in\Omega}|\nu(x) - \mu(x)|.$$

Taking expectation over $\hat{b}_T$, we conclude:

$$\mathbb{E}[|h(\hat{b}_T)|] \leq \mathcal{O}\left(\mathbb{E}[\hat{b}_T] + K^*\right). \tag{EC.79}$$

We now show that $\mathbb{E}[\hat{b}_T] \leq \mathcal{O}(\sqrt{T})$. Plugging this into (EC.79) and using the fact that $K^* = \mathcal{O}(\sqrt{T/\gamma}) \leq \mathcal{O}(\sqrt{T})$ (from Claim EC.29), we conclude:

$$\mathbb{E}[|h(\hat{b}_T)|] \leq \mathcal{O}(\sqrt{T}),$$

as desired.

To prove $\mathbb{E}[\hat{b}_T] \leq \mathcal{O}(\sqrt{T})$, Consider a (trivial) stationary policy that accepts every arrival with reward 1 (and reject the others). Let $\tilde{b}_t$ denote the backlog under this policy. Then, for any sample path,

$$\hat{b}_T \leq \tilde{b}_T.$$

Thus, it suffices to bound $\mathbb{E}[\tilde{b}_T]$. Note that $\tilde{b}_t$ evolves as a lazy reflected random walk:

$$\tilde{b}_t = \max(\tilde{b}_{t-1} + \delta_t, 0),$$

where $\delta_t \in \{-1, 0, 1\}$ is an i.i.d. random variable with:

$$\mathbb{P}[\delta_t = -1] = \frac{1}{4}, \quad \mathbb{P}[\delta_t = 0] = \frac{1}{2}, \quad \mathbb{P}[\delta_t = 1] = \frac{1}{4}.$$

Let $S_t = \sum_{\tau=1}^{t} \delta_\tau$ be the associated (unreflected) lazy random walk. By induction, one can show that:

$$\tilde{b}_t = \max_{1 \leq \tau \leq t} S_\tau - S_t,$$

for any sample path. Since $\mathbb{E}[\delta_t] = 0$, it follows that:

$$\mathbb{E}[\tilde{b}_t] = \mathbb{E}\left[\max_{1 \leq \tau \leq t} S_\tau\right],$$

i.e., the expected running maximum of a symmetric lazy random walk. This is known to be $\Theta(\sqrt{t})$; see, for example, Comtet and Majumdar (2005). Therefore:

$$\mathbb{E}[\hat{b}_T] \leq \mathbb{E}[\tilde{b}_T] = \mathcal{O}(\sqrt{T}),$$

as claimed. This completes the proof. □

EC.7.3. Proof of Claim EC.3

For any given $\boldsymbol{\nu} \in \mathcal{D}$ and $\mathbf{g}_t = \nabla f_t(\boldsymbol{\nu}_t)$,

$$
\begin{aligned}
f_t(\boldsymbol{\nu}_t) - f_t(\boldsymbol{\nu}) &\le \mathbf{g}_t \cdot (\boldsymbol{\nu}_t - \boldsymbol{\nu}) \\
&= \hat{\mathbf{g}}_t \cdot (\boldsymbol{\nu}_t - \boldsymbol{\nu}) + (\mathbf{g}_t - \hat{\mathbf{g}}_t) \cdot (\boldsymbol{\nu}_t - \boldsymbol{\nu}) \\
&= \frac{1}{\eta_t}(\nabla h(\boldsymbol{\nu}_t) - \nabla h(\tilde{\boldsymbol{\nu}}_{t+1})) \cdot (\boldsymbol{\nu}_t - \boldsymbol{\nu}) + (\mathbf{g}_t - \hat{\mathbf{g}}_t) \cdot (\boldsymbol{\nu}_t - \boldsymbol{\nu}) && \text{(EC.80)} \\
&= \frac{1}{\eta_t}(V_h(\boldsymbol{\nu}, \boldsymbol{\nu}_t) + V_h(\boldsymbol{\nu}_t, \tilde{\boldsymbol{\nu}}_{t+1}) - V_h(\boldsymbol{\nu}, \tilde{\boldsymbol{\nu}}_{t+1})) + (\mathbf{g}_t - \hat{\mathbf{g}}_t) \cdot (\boldsymbol{\nu}_t - \boldsymbol{\nu}) && \text{(EC.81)} \\
&\le \frac{1}{\eta_t}(V_h(\boldsymbol{\nu}, \boldsymbol{\nu}_t) + V_h(\boldsymbol{\nu}_t, \tilde{\boldsymbol{\nu}}_{t+1}) - V_h(\boldsymbol{\nu}, \boldsymbol{\nu}_{t+1}) - V_h(\boldsymbol{\nu}_{t+1}, \tilde{\boldsymbol{\nu}}_{t+1})) + (\mathbf{g}_t - \hat{\mathbf{g}}_t) \cdot (\boldsymbol{\nu}_t - \boldsymbol{\nu}) && \text{(EC.82)} \\
&= \frac{1}{\eta_t}(V_h(\boldsymbol{\nu}, \boldsymbol{\nu}_t) - V_h(\boldsymbol{\nu}, \boldsymbol{\nu}_{t+1})) + \frac{1}{\eta_t}\underbrace{(V_h(\boldsymbol{\nu}_t, \tilde{\boldsymbol{\nu}}_{t+1}) - V_h(\boldsymbol{\nu}_{t+1}, \tilde{\boldsymbol{\nu}}_{t+1}))}_{(\star)} + (\mathbf{g}_t - \hat{\mathbf{g}}_t) \cdot (\boldsymbol{\nu}_t - \boldsymbol{\nu}) && \text{(EC.83)}
\end{aligned}
$$

The first line is because of the definition of the subgradient. Line (EC.80) is from the update rule (EC.6). Line (EC.81) is from the three-point equality property of V_h (Lemma 5.2 of Bubeck (2011)). Line (EC.82) is due to Generalized Pythagorean Theorem (Lemma 5.3 of Bubeck (2011)). We now bound the term $(\star)$ as

$$
\begin{aligned}
(\star) &= h(\boldsymbol{\nu}_t) - h(\boldsymbol{\nu}_{t+1}) - \nabla h(\tilde{\boldsymbol{\nu}}_{t+1}) \cdot (\boldsymbol{\nu}_t - \boldsymbol{\nu}_{t+1}) && \text{(EC.84)} \\
&\le (\nabla h(\boldsymbol{\nu}_t) - \nabla h(\tilde{\boldsymbol{\nu}}_{t+1})) \cdot (\boldsymbol{\nu}_t - \boldsymbol{\nu}_{t+1}) - \frac{\sigma}{2}\|\boldsymbol{\nu}_t - \boldsymbol{\nu}_{t+1}\|^2 && \text{(EC.85)} \\
&= \eta_t \hat{\mathbf{g}}_t \cdot (\boldsymbol{\nu}_t - \boldsymbol{\nu}_{t+1}) - \frac{\sigma}{2}\|\boldsymbol{\nu}_t - \boldsymbol{\nu}_{t+1}\|^2 && \text{(EC.86)} \\
&\le \eta_t \|\hat{\mathbf{g}}_t\|_* \|\boldsymbol{\nu}_t - \boldsymbol{\nu}_{t+1}\| - \frac{\sigma}{2}\|\boldsymbol{\nu}_t - \boldsymbol{\nu}_{t+1}\|^2 && \text{(EC.87)} \\
&\le \frac{\eta_t^2 \|\hat{\mathbf{g}}_t\|_*^2}{2\sigma} && \text{(EC.88)}
\end{aligned}
$$

Line (EC.84) is by the definition of V_h. Line (EC.85) is because $h(\cdot)$ is σ-strongly convex with respect to $\|\cdot\|$. Line (EC.86) is again due to the update rule (EC.6). In the final two lines, we used the generalized Cauchy-Schartz inequality and the fact that $az - bz^2 \le \frac{a^2}{4b}$ for any $a > 0, b > 0$ and $z \in \mathbb{R}$.

Plugging the bound of $(\star)$ in (EC.88) into (EC.83), we have

$$
f_t(\boldsymbol{\nu}_t) - f_t(\boldsymbol{\nu}) \le \frac{1}{\eta_t}(V_h(\boldsymbol{\nu}, \boldsymbol{\nu}_t) - V_h(\boldsymbol{\nu}, \boldsymbol{\nu}_{t+1})) + \frac{\eta_t \|\hat{\mathbf{g}}_t\|_*^2}{2\sigma} + (\mathbf{g}_t - \hat{\mathbf{g}}_t) \cdot (\boldsymbol{\nu}_t - \boldsymbol{\nu}) \qquad \text{(EC.89)}
$$

We sum up the preceding inequality for $k \le t \le s$ to obtain

$$
\begin{aligned}
\sum_{t=k}^{s}(f_t(\boldsymbol{\nu}_t) - f_t(\boldsymbol{\nu})) &\le \sum_{t=k}^{s}\frac{1}{\eta_t}(V_h(\boldsymbol{\nu}, \boldsymbol{\nu}_t) - V_h(\boldsymbol{\nu}, \boldsymbol{\nu}_{t+1})) + \sum_{t=k}^{s}\frac{\eta_t \|\hat{\mathbf{g}}_t\|_*^2}{2\sigma} + \sum_{t=k}^{s}(\mathbf{g}_t - \hat{\mathbf{g}}_t) \cdot (\boldsymbol{\nu}_t - \boldsymbol{\nu}) \\
&= \sum_{t=k+1}^{s}\left(\frac{1}{\eta_t} - \frac{1}{\eta_{t-1}}\right)V_h(\boldsymbol{\nu}, \boldsymbol{\nu}_t) + \frac{1}{\eta_k}V_h(\boldsymbol{\nu}, \boldsymbol{\nu}_k) - \frac{1}{\eta_s}V_h(\boldsymbol{\nu}, \boldsymbol{\nu}_{s+1}) + \sum_{t=k}^{s}\frac{\eta_t \|\hat{\mathbf{g}}_t\|_*^2}{2\sigma} + \\
&\quad \sum_{t=k}^{s}(\mathbf{g}_t - \hat{\mathbf{g}}_t) \cdot (\boldsymbol{\nu}_t - \boldsymbol{\nu})
\end{aligned}
$$

$$\leq \sum_{t=k+1}^{s}\left(\frac{1}{\eta_t}-\frac{1}{\eta_{t-1}}\right)V_h(\boldsymbol{\nu},\boldsymbol{\nu}_t)+\frac{1}{\eta_k}V_h(\boldsymbol{\nu},\boldsymbol{\nu}_k)+\sum_{t=k}^{s}\frac{\eta_t\|\hat{\mathbf{g}}_t\|_*^2}{2\sigma}+\sum_{t=k}^{s}(\mathbf{g}_t-\hat{\mathbf{g}}_t)\cdot(\boldsymbol{\nu}_t-\boldsymbol{\nu})$$

where the last inequality follows from the non-negativity of V_h. This completes the proof.

EC.8. Missing Details and Proofs of Section 4

EC.8.1. Formal description of CO-DL

We provide the formal description of the congestion-oblivious algorithm in Algorithm 2.

Algorithm 2 Congestion-Oblivious Dual-Learning (CO-DL) Algorithm

Input: T, $\boldsymbol{\rho}$, constant $k>0$, and $\eta_t \leftarrow \frac{k}{\sqrt{t}}$ for all t.
Initialize $\theta_{1,i} \leftarrow \exp(-1)$, $\lambda_{1,i} \leftarrow \exp(-1)$, and $c_{0,i} \leftarrow \rho_i T$ for all $i \in [m]$.
for each arrival $\{\mathbf{A}_t\}_{t=1}^{T}$ **do**
  if $i_t^\dagger \in [m]$ **then**
    Set $\mathbf{z}_t = \mathbf{e}_{i_t^\dagger}$
  else
    if $\min_{i\in[m]} c_{t-1,i} > 0$ **then**
      Set $\mathbf{z}_t \in \arg\max_{\mathbf{z}\in\Delta_m\cap\{0,1\}^m}(\mathbf{w}_t-\boldsymbol{\theta}_t-\boldsymbol{\lambda}_t)\cdot\mathbf{z}$
    else
      Set $\mathbf{z}_t = \mathbf{0}$
    end if
  end if
  Update the remaining capacity: $c_{t,i} \leftarrow c_{t-1,i} - z_{t,i} \quad \forall i \in [m]$
  Update the dual variables for all $i \in [m]$:

$$\theta_{t+1,i} \leftarrow \min\{\theta_{t,i}\exp(\eta_t(z_{t,i}-\rho_i)),\alpha\} \tag{EC.90}$$

$$\lambda_{t+1,i} \leftarrow \min\left\{\lambda_{t,i}\exp(\eta_t(z_{t,i}-\rho_i)),\frac{1+2\alpha}{\underline{\rho}}\right\} \tag{EC.91}$$

end for

EC.8.2. Preliminaries: Properties of the Static Dual Problem

The purpose of this section is to study the static dual problem defined in Section 4.2.1. We prove some properties of the static dual problem which will be useful throughout the proof of Theorem 2 (Section EC.8.4).

Recall that we use $\boldsymbol{\nu}=(\boldsymbol{\theta},\boldsymbol{\lambda})\in\mathbb{R}_+^{2m}$ to collectively denote the two dual variables. For ease of reference, we recall that the static dual problem (Definition 6 in Section 4.2.1) was defined as

$$\begin{aligned} D(\boldsymbol{\nu}) &:= \mathbb{E}\left[\max_{\mathbf{z}\in\mathcal{Z}(i^\dagger)}(\mathbf{w}-\boldsymbol{\theta}-\boldsymbol{\lambda})\cdot\mathbf{z}+\boldsymbol{\rho}\cdot\boldsymbol{\theta}+\boldsymbol{\rho}\cdot\boldsymbol{\lambda}\right] \\ D(\boldsymbol{\nu}^*) &:= \min_{\boldsymbol{\nu}\in\mathcal{V}} D(\boldsymbol{\nu}) \end{aligned} \tag{EC.92}$$

where we recall that $\mathcal{V} := [0,\alpha]^m \times [0, \frac{1+2\alpha}{\underline{\rho}}]$. We further defined the dual-based primal decision $\tilde{\mathbf{z}}(\boldsymbol{\nu}, \mathbf{A})$ and *matching rate* $\boldsymbol{\mu}(\boldsymbol{\nu})$ as

$$\tilde{\mathbf{z}}(\boldsymbol{\nu}, \mathbf{A}) := \arg\max_{\mathbf{z}\in\mathcal{Z}(i^\dagger)} (\mathbf{w} - \boldsymbol{\theta} - \boldsymbol{\lambda}) \cdot \mathbf{z}$$
$$\boldsymbol{\mu}(\boldsymbol{\nu}) := \mathbb{E}_{\mathbf{A}}[\tilde{\mathbf{z}}(\boldsymbol{\nu}, \mathbf{A})].$$

Note that, under Assumption 1, the dual-based primal decision is unique almost surely for each pair of arrival type and dual variable, and hence the dual-based matching rate is well-defined.

In the following, we show some useful properties of the optimal solution $\boldsymbol{\nu}^*$ for the static dual problem.

Claim EC.31 **(Properties of Optimal Solution for Static Dual Problem).** *The optimal static dual variable $\boldsymbol{\nu}^*$ defined in (EC.92) satisfies the following properties:*

(a) $\|\boldsymbol{\theta}^* + \boldsymbol{\lambda}^*\|_\infty \le 1$.[45]

(b) Under Assumption 1, $\mu_i(\boldsymbol{\nu}^*) \le \rho_i$ *with for all* $i \in [m]$ *with equality if* $\theta_i^* > 0$.

(c) Under Assumption 1, $D(\boldsymbol{\nu}^*) = \mathbb{E}_{\mathbf{A}}[\mathbf{w} \cdot \tilde{\mathbf{z}}(\boldsymbol{\nu}^*, \mathbf{A})]$

Proof of Claim EC.31. We first prove part (a). Suppose for a contradiction that $\theta_i^* + \lambda_i^* > 1$ for some $i \in [m]$. To derive contradiction, we will construct an alternative solution $\tilde{\boldsymbol{\nu}}$ that can strictly decrease the dual objective function. Note that we can write $D(\boldsymbol{\nu})$ as

$$D(\boldsymbol{\nu}) = \sum_{j=1}^m \mathbb{E}[(w_j - \theta_j - \lambda_j) \cdot \mathbb{1}[i^\dagger = j]] + \mathbb{E}\left[\max_{j\in[m]} (w_j - \theta_j - \lambda_j)_+ \cdot \mathbb{1}[i^\dagger = 0]\right] + \sum_{j=1}^m \rho_j(\theta_j + \lambda_j).$$

We now define $\tilde{\boldsymbol{\theta}} := \boldsymbol{\theta}^* - \delta \mathbf{e}_i$ and $\tilde{\boldsymbol{\lambda}} := \boldsymbol{\lambda}^* - \delta \mathbf{e}_i$ with infinitesimally small $\delta > 0$. We first observe that

$$\max_{j\in[m]} (w_j - \tilde{\theta}_j - \tilde{\lambda}_j)_+ = \max_{j\in[m]} (w_j - \theta_j^* - \lambda_j^*)_+ \tag{EC.93}$$

for every sample path. To see this, recall that $w_i \le 1$ (by our assumption in Section 2). Because $\theta_i^* + \lambda_i^* > 1$, we have $w_i - \theta_i^* - \lambda_i^* < 0$ and therefore $w_i - \tilde{\theta}_i - \tilde{\lambda}_i < 0$ by choosing sufficiently small $\delta > 0$. Furthermore, $(w_j - \theta_j^* - \lambda_j^*)_+ = (w_j - \tilde{\theta}_j - \tilde{\lambda}_j)_+$ for all $j \ne i$ by definition of $\tilde{\boldsymbol{\theta}}$ and $\tilde{\boldsymbol{\lambda}}$. Combining the two facts leads to line (EC.93). Consequently, we have:

$$\begin{aligned} D(\tilde{\boldsymbol{\nu}}) - D(\boldsymbol{\nu}^*) &= 2\delta\mathbb{E}[\mathbb{1}[i^\dagger = i]] + \mathbb{E}\left[\left\{\max_{j\in[m]} (w_j - \tilde{\theta}_j - \tilde{\lambda}_j)_+ - \max_{j\in[m]} (w_j - \theta_j^* - \lambda_j^*)_+\right\} \mathbb{1}[i^\dagger = 0]\right] - 2\delta\rho_i \\ &= 2\delta(\mathbb{P}[i^\dagger = i] - \rho_i) \\ &< 0. \end{aligned}$$

The second line follows from (EC.93). The last line is due to our assumption $\mathbb{P}[i^\dagger = i] < \rho_i$ for all $i \in [m]$ (see Section 2). Hence, we have proved Claim EC.31-(a).

[45] It is worth pointing out that we can re-parameterize this dual problem by defining $\boldsymbol{\phi} = \boldsymbol{\theta} + \boldsymbol{\lambda}$. However, for our theoretical analysis, it is helpful to define $\boldsymbol{\theta}$ and $\boldsymbol{\lambda}$ separately as they play different roles. In simple terms, $\boldsymbol{\theta}$ and $\boldsymbol{\lambda}$ of our algorithms control the over-allocation cost and the stopping time T_A (see eq. (16) for the definition), respectively see, for example, Claim EC.6 and EC.7 in Section EC.6.3.

We now prove parts (b) and (c). Let $\bar{\lambda} = \frac{1+2\alpha}{\underline{\rho}}$. By the Karush-Kuhn-Tucker condition for $\boldsymbol{\nu}^*$, there exists $\underline{u}_i \geq 0$ and $\overline{u}_i \geq 0$ such that

$$\frac{\partial D(\boldsymbol{\nu})}{\partial \lambda_i}\Big|_{\nu=\boldsymbol{\nu}^*} - \underline{u}_i + \overline{u}_i = 0 \tag{EC.94}$$

$$\underline{u}_i \lambda_i^* = 0 \tag{EC.95}$$

$$\overline{u}_i(\lambda_i^* - \bar{\lambda}) = 0. \tag{EC.96}$$

Because $\bar{\lambda} > 1$ (due to $\rho_i \leq 1$ for all $i \in [m]$) and $\lambda_i^* \leq 1$ by part (a), we must have $\overline{u}_i = 0$. Furthermore, Assumption 1 implies that each $\tilde{\mathbf{z}}(\boldsymbol{\nu}, \mathbf{A})$ is unique almost surely. Therefore, by Theorem 7.44 of (Shapiro et al. 2021), the function $D(\cdot)$ is differentiable and the partial derivative with respect to λ_i is

$$\frac{\partial D(\boldsymbol{\nu})}{\partial \lambda_i}\Big|_{\boldsymbol{\nu}=\boldsymbol{\nu}^*} = \rho_i - \mathbb{E}[\tilde{z}_i(\boldsymbol{\nu}^*, \mathbf{A})]. \tag{EC.97}$$

Plugging this into the preceding Karush-Khun-Tucker conditions (along with $\overline{u}_i = 0$), we have:

$$\rho_i - \mathbb{E}[\tilde{z}_i(\boldsymbol{\nu}^*, \mathbf{A})] \geq 0 \tag{EC.98}$$

$$(\rho_i - \mathbb{E}[\tilde{z}_i(\boldsymbol{\nu}^*, \mathbf{A})])\lambda_i^* = 0. \tag{EC.99}$$

Part (b) is a direct consequence of line (EC.98). For part (c), one can follow the identical steps as before to show the complementary slackness $(\rho_i - \tilde{z}_i(\boldsymbol{\nu}^*, \mathbf{A})])\theta_i^* = 0$ for all $i \in [m]$. Combining this with line (EC.99), the dual objective at optimal dual $\boldsymbol{\nu}^*$ must be $D(\boldsymbol{\nu}^*) = \mathbb{E}[\mathbf{w} \cdot \tilde{\mathbf{z}}(\boldsymbol{\nu}^*, \mathbf{A})]$. This completes the proof of part (c). □

EC.8.3. Proof of Proposition 3

General Setup. We begin by describing a general setup of online stochastic mirror descent (OSMD). Let $D : \mathcal{V} \to \mathbb{R}$ be a convex function and (with a slight abuse of notation) $\boldsymbol{\nu}_t$ be a sequence of iterates such that

$$\nabla h(\mathbf{y}_b) = \nabla h(\boldsymbol{\nu}_{b-1}) - \eta_{b-1}\hat{\mathbf{g}}_{b-1}$$

$$\boldsymbol{\nu}_b = \arg\min_{\boldsymbol{\nu} \in \mathcal{V}} V_h(\boldsymbol{\nu}, \mathbf{y}_b)$$

where

(i) (Bounded domain): $\mathcal{V}$ is a bounded convex set

(ii) (Strongly convex mirror map): $h(\cdot) : \mathcal{V} \to \mathbb{R}$ is a σ-strongly convex on domain $\mathcal{V}$ with respect to $\|\cdot\|_1$.

(iii) (Unbiased and bounded Gradient): $\mathbb{E}[\hat{\mathbf{g}}_t | \mathcal{H}_{t-1}] = \nabla D(\boldsymbol{\nu}_t)$ and $\|\hat{\mathbf{g}}_t\|_\infty \leq G$ almost surely for a constant G.

(iv) (Bounded Noise): The noise $\hat{\mathbf{u}}_t := \mathbb{E}[\hat{\mathbf{g}}_t | \mathcal{H}_{t-1}] - \hat{\mathbf{g}}_t$ satisfies $\|\hat{\mathbf{u}}\|_1 \leq U$ almost surely for some constant U.

(v) $\eta_t = \frac{k}{\sqrt{t}}$ for some constant $k > 0$.

The following is the main result we will establish.

THEOREM 32 **(Extending Theorem 3.2 of Harvey et al. (2019) to OSMD).** *Define* $\boldsymbol{\nu}^*$ *such that*

$$\boldsymbol{\nu}^* := \arg\min_{\boldsymbol{\nu}\in\mathcal{V}} D(\boldsymbol{\nu}).$$

Under the conditions ((i))-((v)) *and for any given* s*, we have*

$$\mathbb{P}\left[D(\boldsymbol{\nu}_s) - D(\boldsymbol{\nu}^*) \le \mathcal{O}\left(\frac{\log(s)\log(1/\delta)}{\sqrt{s}}\right)\right] \ge 1-\delta. \tag{EC.100}$$

Connection to Proposition 3. To see the connection of the theorem to Proposition 3, recall that we aimed to study the last-iterate convergence of the dual variables $\{\boldsymbol{\nu}_t\}_{t=1}^{T_A}$ of Algorithm 2. For ease of reference, we recall that the domain of the dual variables was given by $\mathcal{V} := [0,\alpha]^m \times [0,\bar{\lambda}]^m$ where $\bar{\lambda} := \frac{1+2\alpha}{\underline{\rho}}$. The static dual function (Definition 6 in Section 4.2.1) was defined as:

$$D(\boldsymbol{\nu}) := \mathbb{E}\left[\max_{\mathbf{z}\in\mathcal{Z}(i^\dagger)} (\mathbf{w}-\boldsymbol{\theta}-\boldsymbol{\lambda})\cdot\mathbf{z} + \boldsymbol{\rho}\cdot\boldsymbol{\theta} + \boldsymbol{\rho}\cdot\boldsymbol{\lambda}\right].$$
$$D(\boldsymbol{\nu}^*) = \min_{\boldsymbol{\nu}\in\mathcal{V}} D(\boldsymbol{\nu}).$$

The dual variables $\{\boldsymbol{\nu}_t\}_{t=1}^{T_A}$ is the sequence of OMSD (with the negative entropy mirror map), as we discussed in Section 4. Furthermore, the primitives (U,G,σ) for our variant of OSMD is constant. In particular, it is straightforward to see $G=1$, $U=2$, and $\sigma = \frac{1}{2m\max(\bar{\lambda},\alpha)}$ Hence, applying Theorem 2, we directly obtain Proposition 3.

EC.8.3.1. Proof of Theorem 32

The proof closely follows the seminal work of Harvey et al. (2019), which establishes the same result for stochastic gradient descent, a special case of stochastic mirror descent with a mirror map denoted as $h(\boldsymbol{\nu}) = \|\boldsymbol{\nu}\|_2^2$. Given our primary use of the multiplicative update rule, an online mirror descent employing the negative entropy mirror map, we need to extend the result to the general mirror descent setting. Although extending the proof of Harvey et al. (2019) to non-Euclidean geometry is straightforward, the exact result we require is, to the best of our knowledge, not available in the literature. Therefore, for the sake of completeness, we will revisit the main steps of Harvey et al. (2019) and highlight necessary modifications to prove the result for our setting. To avoid unnecessary repetitions, we clearly label any result that can be directly derived from Harvey et al. (2019) and kindly refer readers to the original paper.

Without loss of generality, we assume that the primitives (U,G,σ) are the values corresponding to our variants of OSMD, as given in the last paragraph of Section EC.8.3. We begin with the standard result on the average convergence guarantee, which follows from Claim EC.3.

LEMMA EC.33 **(Average convergence).** *For any* $\boldsymbol{\nu}\in\mathcal{V}$*,* k*, and* s*, we have*

$$\sum_{t=k}^{s}(D(\boldsymbol{\nu}_t) - D(\boldsymbol{\nu})) \le \sum_{t=k}^{s}\frac{\eta_t G^2}{2\sigma} + \frac{1}{\eta_k}V_h(\boldsymbol{\nu},\boldsymbol{\nu_k}) + \sum_{t=k}^{s}\hat{\mathbf{u}}_t\cdot(\boldsymbol{\nu}_t-\boldsymbol{\nu}) + \sum_{t=k+1}^{s}\left(\frac{1}{\eta_t}-\frac{1}{\eta_{t-1}}\right)V_h(\boldsymbol{\nu},\boldsymbol{\nu}_t)$$

The following lemma associates the last-iterate convergence with the average convergence as follows.

LEMMA EC.34 **(Lemma 8.1 of Harvey et al. (2019))**. *For any $s \geq 1$,*

$$D(\boldsymbol{\nu}_s) - D(\boldsymbol{\nu}^*) \leq \underbrace{\frac{1}{s/2+1} \sum_{t=s/2}^{s} [D(\boldsymbol{\nu}_t) - D(\boldsymbol{\nu}^*)]}_{\spadesuit} + \mathcal{O}\left(\frac{\log(s)}{\sqrt{s}}\right) + \underbrace{\sum_{t=s/2}^{s} \boldsymbol{\omega}_t \cdot \hat{\mathbf{u}}_t}_{\clubsuit}$$

where

$$\boldsymbol{\omega}_t = \sum_{j=s/2}^{t} \frac{1}{(s-j)(s-j+1)} (\boldsymbol{\nu}_t - \boldsymbol{\nu}_j). \tag{EC.101}$$

The proof mirrors that of Lemma 8.1 in Harvey et al. (2019) and we omit the proofs for brevity. Lemma EC.34 implies that the last iterate optimality gap is related to that of the suffix average (♠) and the weighted sum of the noise of the gradients (♣). Hence, it suffices to obtain the high-probability bound for each term. We first begin with the term ♠.

Bounding ♠.

LEMMA EC.35 **(Lemma 8.2 of Harvey et al. (2019))**. *With probability at least $1-\delta$,*

$$\frac{1}{s/2+1} \sum_{t=s/2}^{s} [D(\boldsymbol{\nu}_t) - D(\boldsymbol{\nu}^*)] \leq \mathcal{O}\left(\frac{\sqrt{\log(1/\delta)}}{\sqrt{s}}\right)$$

The proof again mirrors Lemma 8.2 of Harvey et al. (2019) and we kindly refer the readers to the original paper. The idea is to observe that $\hat{\mathbf{u}}_t \cdot (\boldsymbol{\nu}_t - \boldsymbol{\nu}^*)$ is a Martingale-difference sequence with respect to the filtration $\{\mathcal{H}_t\}$ because $\mathbb{E}[\hat{\mathbf{u}}_t | \mathcal{H}_{t-1}] = 0$ and $\boldsymbol{\nu}_t$ is $\mathcal{H}_{t-1}$-measurable. Due to the assumption of boundedness of all the primitives (domain, noise, and the norm of gradients), the magnitude of the difference sequence is bounded by a constant. Hence, a direct application of Azuma's inequality leads to the lemma.

Bounding ♣.

The main technical lemma will be:

LEMMA EC.36.

$$\sum_{t=s/2}^{s} \boldsymbol{\omega}_t \cdot \hat{\mathbf{u}}_t \leq \mathcal{O}\left(\frac{\log(s)\log(1/\delta)}{\sqrt{s}}\right)$$

with probability at least $1-\delta$.

Combining the previous two lemmas directly implies Theorem 32. The proof of Lemma EC.36 is fairly intricate. We outline the proof in the following. Similar to the previous lemma, we note that $\boldsymbol{\omega}_t \cdot \hat{\mathbf{u}}_t$ is a Marintagle difference sequence with respect to $\{\mathcal{H}_t\}$. Hence, the standard Azuma-type inequality would require a bound of $\sum_{t=s/2}^{s} \|\boldsymbol{\omega}_t\|^2$ to obtain a high-probability bound of ♣. However, we will see that the bound of $\sum_{t=s/2}^{s} \|\boldsymbol{\omega}_t\|^2$ itself is related to a linear combination of the Martingale difference sequence $\boldsymbol{\omega}_t \cdot \hat{\mathbf{u}}_t$. This non-standard nature requires a more sophisticated concentration inequality than the plain Azuma's inequality. For this purpose, we extract the following concentration result from Harvey et al. (2019).

THEOREM 37 **(Corollary C.5 of Harvey et al. (2019))**. *Let $d_t := \mathbf{a}_t \cdot \mathbf{b}_t$ where $\mathbf{a}_t$ is $\mathcal{H}_t$ measurable and $\mathbf{b}_t$ is $\mathcal{H}_{t-1}$ measurable. Suppose $\mathbb{E}[\mathbf{a}_t|\mathcal{H}_{t-1}] = 0$, $\|\mathbf{a}_t\|_2 \le 1$ almost surely, and there exists positive values $\{\tilde{\alpha}_t\}_{t=1}^s$ and $R > 0$ such that*

(a) $\max_{t\in[s]} \tilde{\alpha}_t \le \mathcal{O}(\sqrt{R})$

(b) $\sum_{t=1}^s \|\mathbf{b}_t\|_2^2 \le \sum_{t=1}^{s-1} \tilde{\alpha}_t d_t + R\sqrt{\log(1/\delta)}$ *with probability at least* $1-\delta$.

Then $\sum_{t=1}^T d_t \le \mathcal{O}(\sqrt{R}\log(1/\delta))$ *with probability at least* $1-\delta$.

To see the connection of Theorem 37 to our task of bounding ♣, let $\mathbf{a}_t = \frac{1}{2}\hat{\mathbf{u}}_t$ and $\mathbf{b}_t = \boldsymbol{\omega}_t$, which are $\mathcal{H}_t$ and $\mathcal{H}_{t-1}$ measurable, respectively. Note that $\mathbb{E}[\hat{\mathbf{u}}_t|\mathcal{H}_{t-1}] = 0$ and $\|\mathbf{a}_t\|_2 \le \|\mathbf{a}_t\|_1 \le 1$ because of $U = 2$ in our variant of OSMD. Hence, if we can obtain a bound in the form of Theorem 37-(a) and (b), the proof is complete. Naturally, by the definition of $\boldsymbol{\omega}_t$ in line (EC.101), this requires studying $\|\boldsymbol{\nu}_t - \boldsymbol{\nu}_j\|_2^2$ for arbitrary pair (j,t). The following claim makes this point explicit.

CLAIM EC.38 **(Claim E.3 of Harvey et al. (2019))**. *For any $t \le s$,*

$$\|\boldsymbol{\omega}_t\|_2^2 \le \frac{1}{s-t+1} \sum_{j=s/2}^{t-1} \alpha_j \|\boldsymbol{\nu}_t - \boldsymbol{\nu}_j\|_2^2$$

where $\alpha_j := \frac{1}{(s-j)(s-j+1)}$.

The proof is again mainly algebraic and can be found in Claim E.3 of Harvey et al. (2019). In light of Theorem 37 and Claim EC.38, we obtain the bound on $\|\boldsymbol{\nu}_t - \boldsymbol{\nu}_j\|_2^2$ for arbitrary pair (j,t) in the following.

LEMMA EC.39 **(Extension of Lemma 7.3 of Harvey et al. (2019) to Mirror Descent)**. *For any $a < b$, we have*

$$\frac{\sigma}{2}\|\boldsymbol{\nu}_a - \boldsymbol{\nu}_b\|_2^2 \le \sum_{i=a}^{b-1} \frac{G^2\eta_i^2}{\sigma} + \sum_{i=a}^{b-1} \eta_i (D(\boldsymbol{\nu}_a) - D(\boldsymbol{\nu}_i)) + \sum_{i=a}^{b-1} \eta_i \hat{\mathbf{u}}_i \cdot (\boldsymbol{\nu}_i - \boldsymbol{\nu}_a)$$

This is the key lemma we extend from Harvey et al. (2019) where the same result was established only for the stochastic gradient descent. The original result of Harvey et al. (2019) relies on the fact that the update rule is performed in the Euclidean space.[46] However, the original the proof cannot be directly applied to a general online mirror descent setting because, for general mirror descent, the dual update is performed through the Bregman distance rather than the Euclidean distance. Fortunately, the Bregman distance shares similar structural properties with the Euclidean distance. Hence, we can establish a recursive relationship between $V_h(\boldsymbol{\nu}_a, \boldsymbol{\nu}_b)$ and $V_h(\boldsymbol{\nu}_a, \boldsymbol{\nu}_{b-1})$ and use the strong convexity of $h(\cdot)$ to translate the result to that of the Euclidean distance. We prove the lemma in the following.

Proof of Lemma EC.39. Throughout the proof, we will use $\|\cdot\|$ to denote $\mathcal{L}_1$-norm as a primal norm. We also use the following standard result.

[46] That is, the form of $\boldsymbol{\nu}_{t+1} = \Pi_{\mathcal{V}}(\boldsymbol{\nu} - \eta_t \hat{\mathbf{g}}_t)$ where $\Pi_{\mathcal{V}}(\cdot)$ the Euclidean projection onto set $\mathcal{V}$.

FACT 1 **(Three point equality: Lemma 5.2 of Bubeck (2011)).** *For any* $\mathbf{a}$, $\mathbf{b}$, *and* $\mathbf{c}$,

$$V_h(\mathbf{a},\mathbf{c}) = V_h(\mathbf{a},\mathbf{b}) + V_h(\mathbf{b},\mathbf{c}) + (\nabla h(\mathbf{b}) - \nabla h(\mathbf{c}))\cdot(\mathbf{a}-\mathbf{b})$$

FACT 2 **(Bregman Projection: Lemma 5.3 of Bubeck (2011)).** *For a convex set* C, *let*

$$\mathbf{x}^* = \arg\min_{\mathbf{x}\in C} V_h(\mathbf{x},\mathbf{x}_0).$$

For any $\mathbf{y}\in C$, *we have*

$$V_h(\mathbf{y},\mathbf{x}_0) \geq V_h(\mathbf{y},\mathbf{x}^*) + V_h(\mathbf{x}^*,\mathbf{x}_0)$$

FACT 3 **(Lemma 5.1 of Bubeck (2011)).** *Let* $h^*(\mathbf{y}) = \sup_{\mathbf{x}\in\mathcal{D}}\{\mathbf{x}\cdot\mathbf{y} - h(\mathbf{x})\}$ *be convex conjugate of a convex function* h. *Then* $\nabla(h^*) = (\nabla(h))^{-1}$. *That is,* $\nabla(h^*)$ *is inverse of* $\nabla(h)$.

FACT 4 **(Theorem 5.26 of Beck (2017)).** *The following is equivalent:*

1. h *is strongly-convex in* $\|\cdot\|$ *with modulus* σ. *That is,* $h(\mathbf{x}) \geq h(\mathbf{y}) + \nabla h(\mathbf{y})\cdot(\mathbf{y}-\mathbf{x}) + \frac{\sigma}{2}\|\mathbf{x}-\mathbf{y}\|^2$
2. h^* *(convex conjugate of* h*) is smooth in* $\|\cdot\|_*$ *with modulus* $1/\sigma$. *That is,* $\|\nabla h^*(\mathbf{x}) - \nabla h^*(\mathbf{y})\| \leq \frac{1}{\sigma}\|\mathbf{x}-\mathbf{y}\|_*$ *where* $\|\cdot\|_*$ *is the dual norm of* $\|\cdot\|$.

Recall that we can write the update rule of the dual variable as

$$\nabla h(\mathbf{y}_b) = \nabla h(\boldsymbol{\nu}_{b-1}) - \eta_{b-1}\hat{\mathbf{g}}_{b-1} \tag{EC.102}$$
$$\boldsymbol{\nu}_b = \arg\min_{\boldsymbol{\nu}\in\mathcal{V}} V_h(\boldsymbol{\nu},\mathbf{y}_b)$$

We first establish the following recursion:

$$\begin{aligned} V_h(\boldsymbol{\nu}_a,\boldsymbol{\nu}_b) &\leq V_h(\boldsymbol{\nu}_a,\mathbf{y}_b) \\ &\leq V_h(\boldsymbol{\nu}_a,\boldsymbol{\nu}_{b-1}) + V_h(\boldsymbol{\nu}_{b-1},\mathbf{y}_b) + (\nabla h(\boldsymbol{\nu}_{b-1}) - \nabla h(\mathbf{y}_b))\cdot(\boldsymbol{\nu}_a - \boldsymbol{\nu}_{b-1}) \qquad \text{(EC.103)} \\ &\leq V_h(\boldsymbol{\nu}_a,\boldsymbol{\nu}_{b-1}) + V_h(\boldsymbol{\nu}_{b-1},\mathbf{y}_b) + \eta_{b-1}\hat{\mathbf{g}}_{b-1}\cdot(\boldsymbol{\nu}_a - \boldsymbol{\nu}_{b-1}) \end{aligned}$$

The first line follows from Fact 2. In the second line, we used Fact 1. The last line follows from Line (EC.102). We now bound the second term in the following.

$$\begin{aligned} V_h(\boldsymbol{\nu}_{b-1},\mathbf{y}_b) &\leq V_h(\boldsymbol{\nu}_{b-1},\mathbf{y}_b) + V_h(\mathbf{y}_b,\boldsymbol{\nu}_{b-1}) \\ &= (\nabla h(\boldsymbol{\nu}_{b-1}) - \nabla h(\mathbf{y}_b))\cdot(\boldsymbol{\nu}_{b-1} - \mathbf{y}_b) && \text{(EC.104)} \\ &= \eta_{b-1}\hat{\mathbf{g}}_{b-1}\cdot(\boldsymbol{\nu}_{b-1} - \mathbf{y}_b) && \text{(EC.105)} \\ &\leq \eta_{b-1}G\|\boldsymbol{\nu}_{b-1} - \mathbf{y}_b\| && \text{(EC.106)} \\ &= \eta_{b-1}G\|\nabla h^*(\nabla h(\boldsymbol{\nu}_{b-1})) - \nabla h^*(\nabla h(\mathbf{y}_b))\| && \text{(EC.107)} \\ &\leq \frac{\eta_{b-1}G}{\sigma}\|\nabla h(\boldsymbol{\nu}_{b-1}) - \nabla h(\mathbf{y}_b)\|_* && \text{(EC.108)} \\ &\leq \frac{\eta_{b-1}^2 G^2}{\sigma} && \text{(EC.109)} \end{aligned}$$

The first line is due to the non-negativity of Bregman distance. Line (EC.104) follows from Fact 1.[47] Line (EC.105) comes from the update rule (EC.102). In line (EC.106), we used Cauchy Schwartz and the bounded gradient assumption ($\|\hat{\mathbf{g}}_{b-1}\|_* \le G$). For lines (EC.107) and (EC.108), we applied Facts 3 and 4, respectively. The final line is again due to the update rule (EC.102) and the bounded gradient assumption. Hence, from the last line of (EC.103), We have

$$V_h(\boldsymbol{\nu}_a, \boldsymbol{\nu}_b) \le V_h(\boldsymbol{\nu}_a, \boldsymbol{\nu}_{b-1}) + \frac{\eta_{b-1}^2 G^2}{\sigma} + \eta_{b-1}\hat{\mathbf{g}}_{b-1} \cdot (\boldsymbol{\nu}_a - \boldsymbol{\nu}_{b-1}) \tag{EC.110}$$

By induction, we obtain

$$\begin{aligned} V_h(\boldsymbol{\nu}_a, \boldsymbol{\nu}_b) &\le \sum_{i=a}^{b-1} \frac{\eta_i^2 G^2}{\sigma} + \sum_{i=a}^{b-1} \eta_i \hat{\mathbf{g}}_i \cdot (\boldsymbol{\nu}_a - \boldsymbol{\nu}_i) \\ &= \sum_{i=a}^{b-1} \frac{\eta_i^2 G^2}{\sigma} + \sum_{i=a}^{b-1} \eta_i \mathbf{g}_i \cdot (\boldsymbol{\nu}_a - \boldsymbol{\nu}_i) + \sum_{i=a}^{b-1} \eta_i \hat{\mathbf{u}}_i \cdot (\boldsymbol{\nu}_i - \boldsymbol{\nu}_a) \\ &\le \sum_{i=a}^{b-1} \frac{\eta_i^2 G^2}{\sigma} + \sum_{i=a}^{b-1} \eta_i (D(\boldsymbol{\nu}_a) - D(\boldsymbol{\nu}_i)) + \sum_{i=a}^{b-1} \eta_i \hat{\mathbf{u}}_i \cdot (\boldsymbol{\nu}_i - \boldsymbol{\nu}_a) \end{aligned}$$

The last line is because $\mathbf{g}_i$ is subgradient of D at $\boldsymbol{\nu} = \boldsymbol{\nu}_i$. The proof is complete by noting that (i) $V_h(\boldsymbol{\nu}_a, \boldsymbol{\nu}_b) \ge \frac{\sigma}{2}\|\boldsymbol{\nu}_a - \boldsymbol{\nu}_b\|_1^2$ by strong convexity of h with modulus σ (recall that the primal norm is $\mathcal{L}_1$ norm) with respect to $\|\cdot\|_1$ and (ii) $\|\cdot\|_1^2 \ge \|\cdot\|_2^2$. □

Now that we have successfully extended the above lemma to the general mirror descent setting, we can apply Theorem 37 in the following form.

LEMMA EC.40 **(Claims E.4-E.6 of Harvey et al. (2019)).** *Let* $d_t = \frac{1}{2}\hat{\mathbf{u}}_t \cdot \boldsymbol{\omega}_t$. *There exists* $R = \mathcal{O}(\log^2(s)/s)$ *and positive values* $\tilde{\alpha}_t > 0$ *such that*

1. $\max_{s/2 \le t \le s} \tilde{\alpha}_t \le \mathcal{O}(\sqrt{R})$
2. $\sum_{t=s/2}^{s} \|\boldsymbol{\omega}_t\|_2^2 \le \sum_{t=s/2}^{s-1} \tilde{\alpha}_t d_t + R\sqrt{\log(1/\delta)}$ *with probability at least* $1-\delta$.

The proof is mainly algebraic and can be found in Lemmas E.4-E.6 of Harvey et al. (2019). Here we only give a sketch of the proof. From Claim EC.38 and Lemma EC.39, we have $\sum_{t=s/2}^{s} \|\omega_t\|_2^2 \le \Lambda_1 + \Lambda_2 + \Lambda_3$ where

$$\begin{aligned} \Lambda_1 &:= \frac{2G^2}{\sigma^2} \sum_{t=s/2}^{s} \frac{1}{s-t+1} \sum_{j=s/2}^{t-1} \alpha_j \sum_{i=j}^{t-1} \eta_i^2 \\ \Lambda_2 &:= \frac{2}{\sigma} \sum_{t=s/2}^{s} \frac{1}{s-t+1} \sum_{j=s/2}^{t-1} \alpha_j \sum_{i=j}^{t-1} \eta_i (D(\boldsymbol{\nu}_j) - D(\boldsymbol{\nu}_i)) \\ \Lambda_3 &:= \frac{2}{\sigma} \sum_{t=s/2}^{s} \frac{1}{s-t+1} \sum_{j=s/2}^{t-1} \alpha_j \sum_{i=j}^{t-1} \eta_i \hat{\mathbf{u}}_i (\boldsymbol{\nu}_i - \boldsymbol{\nu}_j). \end{aligned}$$

With some delicate algebras, we can show that $\Lambda_1 \le O\left(\frac{\log^2(s)}{s}\right)$ (see Claim E.4 of Harvey et al. (2019)) and $\Lambda_2 \le O\left(\frac{\log^2(s)\sqrt{\log(1/\delta)}}{s}\right)$ with proabability $1-\delta$ (see Claim E.5 of Harvey et al. (2019)).

[47] That is, $0 = V_h(\boldsymbol{\nu}_{b-1}, \boldsymbol{\nu}_{b-1}) = V_h(\boldsymbol{\nu}_{b-1}, \mathbf{y}_b) + V_h(\mathbf{y}_b, \boldsymbol{\nu}_{b-1}) + (\nabla h(\mathbf{y}_b) - \nabla h(\boldsymbol{\nu}_{b-1})) \cdot (\boldsymbol{\nu}_{b-1} - \mathbf{y}_b)$

Interestingly, to prove the bound for Λ_2, we use the result in Lemma EC.35 again. Finally, note that Λ_3 is essentially the scaled version of term ♣ in Lemma EC.34, which we were aiming to bound. In fact, one can follow the same algebras taken in Claim E.6 of Harvey et al. (2019) to show $\Lambda_3 \leq \frac{2}{\sigma}\sum_{i=s/2}^{s-1}\alpha_i(\hat{\mathbf{u}}_i\cdot\boldsymbol{\omega})$ where $\alpha_i = \frac{4}{\sigma}\eta_i\sum_{j=i+1}^{s}\frac{1}{s-i+1}$ is the extra scaling factor. Note that $\alpha_i \leq \mathcal{O}(\frac{\log(s)}{\sqrt{s}})$ for all $\frac{s}{2} \leq i \leq s$. Combining the bounds for Λ_1, Λ_2, and Λ_3 will give the desired result.

Finally, the direct application of Theorem 37 and Lemma EC.40 leads to Lemma EC.36. This completes the proof of Theorem 32.

EC.8.4. Proof of Theorem 2

We first begin by formally defining the endogenous matching rate of CO-DL given the dual variable:

Definition EC.41 **(Dual-based Matching Rate)**. *For a given $\boldsymbol{\nu} = (\boldsymbol{\theta}, \boldsymbol{\lambda})$, the dual-based matching rate is defined as*

$$\boldsymbol{\mu}(\boldsymbol{\nu}) := \mathbb{E}_{\mathbf{A}}[\tilde{\mathbf{z}}(\boldsymbol{\nu}, \mathbf{A})]$$

where we recall from Definition 6 that $\tilde{\mathbf{z}}(\boldsymbol{\nu}, \mathbf{A}) := \arg\max_{\mathbf{z}\in\mathcal{Z}(i^\dagger)}\{(\mathbf{w}-\boldsymbol{\theta}-\boldsymbol{\lambda})\cdot\mathbf{z}\}$.

Note that, under Assumption 1, the primal-based decision $\tilde{\mathbf{z}}(\boldsymbol{\nu}, \mathbf{A})$ is uniquely determined for each $\boldsymbol{\nu}$ and $\mathbf{A}$ almost surely, and hence the dual-based matching rate is well defined.

Given all the mentioned ingredients, we now outline the main steps to prove Theorem 2. Similar to Section 3.3, we prove Theorem 2 based on the the same regret decomposition as in line (10) and obtain bounds on (A), the loss regarding the expected net matching reward, and (B), the expected average backlog. The following lemma first establishes the bound on (A).

Lemma EC.42 **(Bounding Loss of Net Matching Reward of CO-DL)**. *For any arrival distribution $\mathcal{F}$ and service slack $\epsilon \geq 0$, CO-DL satisfies*

$$\mathbb{E}[\mathsf{NMR}(\{\mathbf{z}_t^*\}_{t=1}^T;\alpha)] - \mathbb{E}[\mathsf{NMR}(\{\mathbf{z}_t\}_{t=1}^T;\alpha)] \leq \mathcal{O}(\sqrt{T})$$

where we recall that $\mathsf{NMR}(\cdot;\alpha)$ is the total net matching reward and $\{\mathbf{z}_t^*\}_{t=1}^T$ is the optimal offline solution (Definition 2).

The proof mirrors that of Lemma 1 and we present the proof in Appendix EC.8.4.1.

In Section 4.2.2, we stated Lemma 6 that bounds (B), the expected average backlog. For ease of reference, we restate the lemma in the following.

Lemma 6 **(Bounding Average Backlog of CO-DL Under Stable Regime)**. *Under Assumption 1 and stable regime (Definition 1), for any arrival distribution $\mathcal{F}$ and service slack $\epsilon > 0$, we have $\mathbb{E}\Big[\frac{1}{T}\sum_{t=1}^T\|\mathbf{b}_t\|_1\Big] \leq \mathcal{O}\Big(\frac{1}{\epsilon}\Big)$.*

The proof consists of three steps. In the first step, we establish that the good event $\mathcal{B}_T$ (see (23) in Section 4.2.2 for its definition) occurs with sufficiently high probability using Proposition 3 and Assumption 1. In the second step, we use the definition of the good event to show an $\mathcal{O}(1/\epsilon)$ bound on the average backlog in expectation up to the stopping time. In the last step, we prove that the average build-up after the stopping time is $\mathcal{O}(1)$. We elaborate each step in the following.

Step 1: Establishing a High-probability Bound of the Good Event.

The following lemma shows that the good event occurs with a high probability.[48]

LEMMA EC.43 **(High-probability Bound of Good Event).** $\Pr[\mathcal{B}_T] \geq 1 - \mathcal{O}(1/T^2)$.

The proof of this lemma is presented in Section EC.8.4.2. For the proof, we translate the high-probability convergence in terms of the dual objective (established in Proposition 3) to that of the matching rate for each $t \in \{\sqrt{T}, T_A\}$ using Assumption 1. We then take the union bound of the preceding high probability bound across $t \in \{\sqrt{T}, T_A\}$ to prove the desired result.

Step 2: Using the Good Event to Bound the Average Backlog up to the Stopping time.

To utilize the good event for our task of bounding the average backlog, we start with the following decomposition of the expected cumulative backlog.

$$\mathbb{E}\Big[\sum_{t=1}^{T}\|\mathbf{b}_t\|_1\Big] = \mathbb{E}\Big[\sum_{t=1}^{\sqrt{T}}\|\mathbf{b}_t\|_1\mathbb{1}[\mathcal{B}_T]\Big] + \underbrace{\mathbb{E}\Big[\sum_{t=\sqrt{T}+1}^{T_A}\|\mathbf{b}_t\|_1\mathbb{1}[\mathcal{B}_T]\Big]}_{\text{(B-1)}} + \underbrace{\mathbb{E}\Big[\sum_{t=T_A+1}^{T}\|\mathbf{b}_t\|_1\mathbb{1}[\mathcal{B}_T]\Big]}_{\text{(B-2)}} + \mathbb{E}\Big[\sum_{t=1}^{T}\|\mathbf{b}_t\|_1\mathbb{1}[\mathcal{B}_T^c]\Big] \tag{Decomposition}$$

The first three terms are the cumulative backlog under the good event. The last term is the one under the complement of the good event. We first note that the first and the last terms can bounded by $\mathcal{O}(T)$ and $\mathcal{O}(1)$, respectively, because (i) $\|\mathbf{b}_t\|_1 \leq \mathcal{O}(t)$ for every sample path and (ii) $\mathbb{P}[\mathcal{B}_T^c] \leq \mathcal{O}(1/T^2)$ from Lemma EC.43. Hence, the remaining task is to obtain an upper bound of the terms (B-1) and (B-2). In the following, we bound (B-1) by $\mathcal{O}(T/\epsilon)$.

LEMMA EC.44 **(Bounding (B-1)).** *The term (B-1) in line* (Decomposition) *is* $\mathcal{O}(T/\epsilon)$.

The proof of Lemma EC.44 can be found in Appendix EC.8.4.3. For the proof, we combine our drift lemma (Lemma 3) with the definition of the good event.

Step 3: Bounding the Average backlog after the Stopping Time.

The last step is to establish that the term (B-2) in (Decomposition), the cumulative backlog after the stopping time, is $\mathcal{O}(T)$. We prove this result in the following lemma:

LEMMA EC.45 **(Bounding (B-2)).** *The term (B-2) in line* (Decomposition) *is* $\mathcal{O}(T)$.

[48] More precisely, there exists a constant $\underline{K}$, which only depends on (i) the service slack parameter $\epsilon > 0$, (ii) the Lipschitz constant $L > 0$ in Assumption 1, and (iii) the input of the algorithm other than T (i.e., $\boldsymbol{\rho}$ and $k > 0$), such that $\mathbb{P}[\mathcal{B}_T] \geq 1 - \frac{1}{T^2}$ for all $T \geq \underline{K}$.

The proof of this lemma follows the identical steps taken in Section EC.6.4. The only modification in the proof is establishing an upper bound on $\mathbb{E}[\|\mathbf{b}_{T_A}\|_1^2]$, which we prove through the following claim.

CLAIM EC.46. *For* `CO-DL` *(Algorithm 2),* $\mathbb{E}[\|\mathbf{b}_{T_A}\|_1^2] \le O(T)$

We prove Claim EC.46 in Appendix EC.8.4.4. The remainder of the proofs proceed identically to those in Section EC.6.4, and we omit them for conciseness.

EC.8.4.1. Proof of Lemma EC.42

The proof mirrors the same step of Lemma 1. Here, we only give an outline of the steps and highlight a main difference compared to the proof of Lemma 1. Define the pseudo-reward $\{K_t\}$ as:

$$K_t := \mathbf{w}_t \cdot \mathbf{z}_t + \boldsymbol{\theta}_t \cdot (\boldsymbol{\rho} - \mathbf{z}_t) + \boldsymbol{\lambda}_t \cdot (\boldsymbol{\rho} - \mathbf{z}_t) \tag{EC.111}$$

We also define the stopping time T_A as line in (16) for `CA-DL` (see Section 3.3). The following lemma lower bounds the expected pseudo-rewards up to the stopping time.

CLAIM EC.47 **(Lower bound on Pseudo-rewards for `CO-DL`).** *For any matching profile* $\{\hat{\mathbf{z}}_t\}_{t=1}^T$ *that satisfies* $\hat{\mathbf{z}}_t \in \mathcal{Z}(i_t^\dagger)$ *for all* $t \in [T]$ *and* (Capacity Feasibility-T) *(see Definition 2), we have*

$$\mathbb{E}\left[\sum_{t=1}^{T_A} K_t\right] \ge \mathbb{E}[\mathsf{NMR}(\{\hat{\mathbf{z}}_t\}_{t=1}^T; \alpha)] - (T - T_A).$$

The proof of this claim follows the identical steps taken in the proof of Lemma 4. We omit the proof for brevity.

The next lemma upper bounds the cumulative pseudo-rewards via the total net matching reward of `CO-DL`.

CLAIM EC.48 **(Upper bound on Pseudo-rewards for `CO-DL`).** *For every sample path* $\{\mathbf{A}_t, \mathbf{s}_t\}_{t=1}^T$, *we have*

$$\sum_{t=1}^{T_A} K_t \le \mathsf{NMR}(\{\mathbf{z}_t\}_{t=1}^T; \alpha) - (T - T_A) + \mathcal{O}(\sqrt{T})$$

Proof of Claim EC.48. The proof follows the similar steps as that of Lemma 5 except for a minor detail arising from the time-varying learning rates. To avoid repetitions, we only highlight the difference. By applying the similar line of algebras in lines (EC.35)-(EC.37) to the new pseudo-reward (EC.111), we can deduce that

$$\begin{aligned}\sum_{t=1}^{T_A} K_t \le \underbrace{\sum_{t=1}^{T_A} \mathbf{w}_t \cdot \mathbf{z}_t - \alpha \sum_{i=1}^{m} \left(\sum_{t=1}^{T} z_{t,i} - c_i\right)_+}_{\mathsf{NMR}(\{\mathbf{z}_t\}_{t=1}^T;\alpha)} + \underbrace{\sum_{t=1}^{T} \boldsymbol{\theta}_t \cdot (\boldsymbol{\rho} - \mathbf{z}_t) - \sum_{t=1}^{T} \boldsymbol{\theta}^\star \cdot (\boldsymbol{\rho} - \mathbf{z}_t)}_{R_\theta} + \\ \underbrace{\sum_{t=1}^{T_A} \boldsymbol{\lambda}_t \cdot (\boldsymbol{\rho} - \mathbf{z}_t)}_{R_\lambda} + 2\alpha(T - T_A).\end{aligned} \tag{EC.112}$$

for some $\boldsymbol{\theta}^\star \in [0,\alpha]^m$. Similar to Claim EC.6, one can invoke the adversarial online learning guarantee of the online mirror descent (Claim EC.3) to show that

$$R_\theta \le 2\alpha \sum_{t=1}^{T} \eta_t + \frac{1}{\eta_1} V_h(\boldsymbol{\theta}^\star, \boldsymbol{\theta}_1) + \sum_{t=2}^{T} \left(\frac{1}{\eta_t} - \frac{1}{\eta_{t-1}} \right) V_h(\boldsymbol{\theta}^\star, \boldsymbol{\theta}_1) \tag{EC.113}$$

where we recall that (i) $\eta_t = \frac{k}{\sqrt{t}}$ with the input constant $k > 0$ (ii) $h(\cdot)$ is the negative entropy function, and (iii) $V_h(\cdot,\cdot)$ is the Bregman distance with respect to h. Because $\boldsymbol{\theta}_t \in [0,\alpha]^m$ for all $t \in [T]$, we can bound $\max_{t\in[T]} V_h(\boldsymbol{\theta}^\star, \boldsymbol{\theta}_t) \le \bar{V}$ for some constant $\bar{V}$. Utilizing the non-negativity of the Bregman divergence and the fact that η_t is non-increasing in t, we can further bound the right-hand side of the inequality (EC.113) as

$$R_\theta \le 2\alpha \sum_{t=1}^{T} \eta_t + \frac{\bar{V}}{\eta_T}.$$

The right hand side of the above inequality is $\mathcal{O}(\sqrt{T})$ because $\sum_{k=1}^{T} \frac{1}{\sqrt{t}} \le 2\sqrt{T}$ and $1/\eta_T = \Theta(\sqrt{T})$. Similarly, One can follow the same algebras and the steps taken in Claim EC.7 (Section EC.6.3) to show that $R_{\boldsymbol{\lambda}} \le O(\sqrt{T}) + (1+2\alpha)(T_A - T)$. Plugging these bound on R_θ and R_λ into line (EC.112) completes the proof. □

Finally, we note that Lemma EC.42 is a direct consequence of the above two claims, by setting $\hat{\mathbf{z}}_t = \mathbf{z}_t^*$ where $\mathbf{z}_t^*$ is the optimal offline solution (Definition 2) for arrival t for each sample path. This completes the proof of Lemma EC.42.

EC.8.4.2. Proof of Lemma EC.43

Our first step is to establish that the dual-based matching rate is L-Lipschitz continuous using Assumption 1, which we prove in the following claim.

Claim EC.49 **(Lipschitz Continuity of Dual-based Matching Rate)**. *Under Assumption 1, we have* $\|\boldsymbol{\mu}(\boldsymbol{\nu}) - \boldsymbol{\mu}(\boldsymbol{\nu}')\|_\infty \le L\|\boldsymbol{\nu} - \boldsymbol{\nu}'\|_1$ *for all* $\boldsymbol{\nu}, \boldsymbol{\nu}' \in \mathcal{V}$.

Proof of Claim EC.49. Let $\boldsymbol{\xi} := \boldsymbol{\nu}' - \boldsymbol{\nu}$. We will show that, for all $i \in [m]$,

$$|\mu_i(\boldsymbol{\nu}) - \mu_i(\boldsymbol{\nu}')| \le L\|\boldsymbol{\nu} - \boldsymbol{\nu}'\|_1.$$

First, observe that

$$\mu_i(\boldsymbol{\nu}) = \mathbb{P}[i^\dagger = 0] \times \mathbb{P}\Big[(w_i - \nu_i)_+ \ge (w_j - \nu_j)_+,\ \forall j \ne i \Big| i = 0\Big] + \mathbb{P}[i^\dagger = i]. \tag{EC.114}$$

Let $f(\cdot)$ denote the PDF of the reward distribution conditional that $i^\dagger = 0$. we then have:

$$\begin{aligned}
&\mathbb{P}\Big[(w_i-\nu_i)_+ \geq (w_j-\nu_j)_+,\ \forall j\neq i \Big| i=0\Big]\\
&=\int_{(w_i-\nu_i)_+\geq(w_j-\nu_j)_+,\ \forall j\neq i} f(\mathbf{w})d\mathbf{w}\\
&=\int_{(w_i-\nu_i'+\xi_i)_+\geq(w_j-\nu_j'+\xi_j)_+,\ \forall j\neq i} f(\mathbf{w})d\mathbf{w}\\
&=\int_{(w_i'-\nu_i')_+\geq(w_j'-\nu_j')_+,\ \forall j\neq i} f(\mathbf{w}'-\boldsymbol{\xi})d\mathbf{w}'\\
&\leq \int_{(w_i'-\nu_i')_+\geq(w_j'-\nu_j')_+,\ \forall j\neq i} f(\mathbf{w}')d\mathbf{w}' \quad + \quad L\|\boldsymbol{\xi}\|_1\int_{(w_i'-\nu_i')_+\geq(w_j'-\nu_j')_+,\ \forall j\neq i} d\mathbf{w}'\\
&\leq \int_{(w_i'-\nu_i')_+\geq(w_j'-\nu_j')_+,\ \forall j\neq i} f(\mathbf{w}')d\mathbf{w}' \quad + L\|\boldsymbol{\xi}\|_1\\
&=\mathbb{P}\Big[(w_i-\nu_i')_+ \geq (w_j-\nu_j')_+,\ \forall j\neq i \Big| i=0\Big] + L\|\boldsymbol{\xi}\|_1
\end{aligned}$$

The fourth line follows from the variable transformation $\mathbf{w}' = \mathbf{w}+\boldsymbol{\xi}$. In the fifth line, we used Assumption 1 that the PDF f is L-Lipschitz with respect to ℓ_1 norm. The sixth line follows because the support of the reward vector is $[0,1]^m$ by our assumption. The rest of the lines are mainly algebraic. Plugging the preceding bound into line (EC.114), we obtain

$$\begin{aligned}
\mu_i(\boldsymbol{\nu}) &\leq \mathbb{P}[i^\dagger=0]\times\left(\mathbb{P}\Big[(w_i-\nu_i')_+ \geq (w_j-\nu_j')_+,\ \forall j\neq i\Big| i=0\Big] + L\|\boldsymbol{\xi}\|_1\right) + \mathbb{P}[i^\dagger=i].\\
&= \mu_i(\boldsymbol{\nu}') + \mathbb{P}[i^\dagger=0]L\|\boldsymbol{\xi}\|_1\\
&\leq \mu_i(\boldsymbol{\nu}') + L\|\boldsymbol{\xi}\|_1
\end{aligned}$$

By switching the role of $\boldsymbol{\nu}$ and $\boldsymbol{\nu}'$, one can also show that $\mu_i(\boldsymbol{\nu}')-\mu_i(\boldsymbol{\nu})\leq L\|\boldsymbol{\xi}\|_1$. Hence, we have $|\mu_i(\boldsymbol{\nu})-\mu_i(\boldsymbol{\nu}')|\leq L\|\boldsymbol{\nu}-\boldsymbol{\nu}'\|_1$ for all $i\in[m]$. This completes the proof. □

Building on the preceding claim, we now translate the last iterate convergence in Proposition 3 to that of the matching rate. Precisely, we prove the following.

Claim EC.50. *For any $\boldsymbol{\nu}\in\mathcal{V}$, we have*

$$\max_{i\in[m]}\{\mu_i(\boldsymbol{\nu})-\rho_i\}\leq\sqrt{2L(D(\boldsymbol{\nu})-D(\boldsymbol{\nu}^*))}.$$

Proof of Claim EC.50. We first observe that Claim EC.49 implies L-smoothness of $D(\cdot)$. That is, for any $\boldsymbol{\nu}_1,\boldsymbol{\nu}_2\in\mathcal{V}$, we have

$$\|\nabla D(\boldsymbol{\nu}_1)-\nabla D(\boldsymbol{\nu}_2)\|_\infty\leq L\|\boldsymbol{\nu}_1-\boldsymbol{\nu}_2\|_1. \tag{EC.115}$$

To see this, we first recall that Assumption 1 implies that each $\tilde{\mathbf{z}}(\boldsymbol{\nu},\mathbf{A})$ is unique almost surely. Therefore, by Theorem 7.44 of (Shapiro et al. 2021), the function $D(\cdot)$ is differentiable and the partial derivative of $D(\cdot)$ is

$$\frac{\partial D(\boldsymbol{\nu})}{\partial\theta_i}=\frac{\partial D(\boldsymbol{\nu})}{\partial\lambda_i}=\rho_i-\mu_i(\boldsymbol{\nu}) \tag{EC.116}$$

which directly implies line (EC.115).

From Theorem 5.8-(iii) of Beck (2017), the L-smoothness in line (EC.115) is equivalent to

$$D(\boldsymbol{\nu}_1) - D(\boldsymbol{\nu}_2) \geq \nabla D(\boldsymbol{\nu}_2) \cdot (\boldsymbol{\nu}_1 - \boldsymbol{\nu}_2) + \frac{1}{2L} \|\nabla D(\boldsymbol{\nu}_1) - \nabla D(\boldsymbol{\nu}_2)\|_\infty^2$$

for all $\boldsymbol{\nu}_1, \boldsymbol{\nu}_2 \in \mathcal{V}$. In particular, letting $\boldsymbol{\nu}^* \in \arg\min_{\boldsymbol{\nu} \in \mathcal{V}} D(\boldsymbol{\nu})$,

$$D(\boldsymbol{\nu}) - D(\boldsymbol{\nu}^*) \geq \nabla D(\boldsymbol{\nu}^*) \cdot (\boldsymbol{\nu} - \boldsymbol{\nu}^*) + \frac{1}{2L} \|\nabla D(\boldsymbol{\nu}) - \nabla D(\boldsymbol{\nu}^*)\|_\infty^2 \geq \frac{1}{2L} \|\nabla D(\boldsymbol{\nu}) - \nabla D(\boldsymbol{\nu}^*)\|_\infty^2$$

for all $\boldsymbol{\nu} \in \mathcal{V}$. Here the last line follows from the first-order condition of $\boldsymbol{\nu}^*$ that minimizes $D(\cdot)$ over domain $\mathcal{V}$ (see, for example, Lemma 5.4 of Bubeck (2011)). Hence, from equation (EC.116), we deduce that, for all $i \in [m]$,

$$\mu_i(\boldsymbol{\nu}) \leq \mu_i(\boldsymbol{\nu}^*) + \sqrt{2L(D(\boldsymbol{\nu}) - D(\boldsymbol{\nu}^*))}.$$

The proof is complete by noting that $\mu_i(\boldsymbol{\nu}^*) \leq \rho_i$ for all $i \in [m]$ from Claim EC.31-(b). □

We now use the above claim to obtain the high-probability bound of good event $\mathcal{B}_T$. For ease of reference, we recall the (equivalent) definition of good event $\mathcal{B}_T$ from line (23):

$$\mathcal{B}_T := \Big\{ \max_{i \in [m]} \{\mu_i(\boldsymbol{\nu}_t) - \rho_i\} \leq \frac{\epsilon}{2} \text{ for all } \sqrt{T} \leq t \leq T_A \Big\}.$$

Claim EC.51. *There exists a constant $\underline{K}$, which only depends on (i) the input of the algorithm (other than T), (ii) $L > 0$ in Assumption 1, and (iii) $\epsilon = \Omega(1)$, such that for all $T \geq \underline{K}$, we have $\mathbb{P}[\mathcal{B}_T] \geq 1 - \frac{1}{T^2}$.*

Proof of Claim EC.51. By Proposition 3, there exists constant $\kappa > 0$ such that, for any given $t \leq T_A$ and $\delta_t \in (0,1)$,

$$\mathbb{P}\left[D(\boldsymbol{\nu}_t) - D(\boldsymbol{\nu}^*) \leq \kappa \frac{\log(t)\log(1/\delta_t)}{\sqrt{t}} \right] \geq 1 - \delta_t.$$

where the constant κ only depends on the input of Algorithm 2 (other than T). We now set $\delta_t = \frac{1}{T^3}$ for all $t \geq \sqrt{T}$ and use Claim EC.50 to obtain

$$\mathbb{P}\left[\max_{i \in [m]} \{\mu_i(\boldsymbol{\nu}_t) - \rho_i\} \leq \frac{\sqrt{6L\kappa \log(t)\log(T)}}{t^{1/4}} \right] \geq 1 - \frac{1}{T^3}$$

for all $t \geq \sqrt{T}$. Note that $f(t) = \frac{\log(t)}{t^{1/2}}$ is decreasing in $t \geq e^2$. Hence, for any $t \geq \sqrt{T} \geq e$, we have

$$\frac{\sqrt{6L\kappa \log(t)\log(T)}}{t^{1/4}} \leq \frac{\sqrt{3L\kappa}\log(T)}{T^{1/8}}$$

The right-hand side converges to zero as $T \to \infty$, meaning that there exists $f(\kappa, L, \epsilon)$ for which the right-hand side is at most $\epsilon/2$ for all $T \geq f(\kappa, L, \epsilon)$ (under the stable regime, we have $f(\kappa, L, \epsilon) = \Omega(1)$). Hence, defining $\underline{K} := \max\{e^4, f(\kappa, L, \epsilon)\}$, we conclude that for all $T \geq \underline{K}$ and $t \geq \sqrt{T}$,

$$\mathbb{P}\left[\max_{i \in [m]} \{\mu_i(\boldsymbol{\nu}_t) - \rho_i\} \leq \frac{\epsilon}{2} \right] \geq 1 - \frac{1}{T^3}. \tag{EC.117}$$

Finally, we use the above bound to lower-bound the good event $\mathcal{B}_T$. For any given T_A, we have:

$$\begin{aligned}\mathbb{P}[\mathcal{B}_T] &= 1 - \mathbb{P}\left[\max_{i\in[m]}\{\mu_i(\boldsymbol{\nu}_t) - \rho_i\} > \frac{\epsilon}{2} \quad \text{for some } t \in [\sqrt{T}, T_A]\right] \\ &\geq 1 - \sum_{t=\sqrt{T}}^{T_A} \mathbb{P}\left[\max_{i\in[m]}\{\mu_i(\boldsymbol{\nu}_t) - \rho_i\} > \frac{\epsilon}{2}\right] \\ &\geq 1 - \frac{1}{T^2}.\end{aligned}$$

In the second line, we used the union bound. The third line follows from inequality (EC.117) and the fact that $T_A \leq T$ for every sample path.[49] This completes the proof.

□

EC.8.4.3. Proof of Lemma EC.44

Recall from Lemma 3 that $\psi(\mathbf{b}_t) - \psi(\mathbf{b}_{t-1}) \leq \mathbf{b}_{t-1} \cdot (\mathbf{z}_t - \mathbf{s}_t) + \mathcal{O}(1)$. Hence, for any history $\mathcal{H}_{t-1}$,

$$\mathbb{E}[\psi(\mathbf{b}_t) - \psi(\mathbf{b}_{t-1})|\mathcal{H}_{t-1}] \leq \mathbf{b}_{t-1} \cdot (\mathbb{E}[\mathbf{z}_t|\mathcal{H}_{t-1}] - \boldsymbol{\rho} - \boldsymbol{\epsilon}) + \mathcal{O}(1) \tag{EC.118}$$

where we used the fact that (i) $\mathbf{b}_{t-1}$ is $\mathcal{H}_{t-1}$-measurable and (ii) the $\mathbf{s}_t$ is i.i.d random variables with mean $\boldsymbol{\rho} - \boldsymbol{\epsilon}$. Recall that, up to $t \leq T_A$, the primal decision is $\mathbf{z}_t = \tilde{\mathbf{z}}(\mathbf{A}_t, \boldsymbol{\nu}_t)$. Because the dual variables $\boldsymbol{\nu}_t$ is $\mathcal{H}_{t-1}$-measurable, we further have

$$\mathbb{E}[\mathbf{z}_t|\mathcal{H}_{t-1}] = \boldsymbol{\mu}(\boldsymbol{\nu}_t).$$

Hence, one can re-write line (EC.118) as

$$\mathbb{E}[\psi(\mathbf{b}_t) - \psi(\mathbf{b}_{t-1})|\mathcal{H}_{t-1}] \leq \mathbf{b}_{t-1} \cdot (\boldsymbol{\mu}(\boldsymbol{\nu}_t) - \boldsymbol{\rho} - \boldsymbol{\epsilon}) + \mathcal{O}(1)$$

for any history $\mathcal{H}_{t-1}$. We now consider two cases. Under the good event $\mathcal{B}_T$, we have

$$\mathbf{b}_{t-1} \cdot (\boldsymbol{\mu}(\boldsymbol{\nu}_t) - \boldsymbol{\rho} - \boldsymbol{\epsilon}) \leq -\frac{\epsilon}{2}\|\mathbf{b}_{t-1}\|_1$$

by the non-negativity of the backlog and by definition of the good event $\mathcal{B}_T$. Otherwise, we have $\mathbf{b}_{t-1} \cdot (\boldsymbol{\mu}(\boldsymbol{\nu}_t) - \boldsymbol{\rho} - \boldsymbol{\epsilon}) \leq \|\mathbf{b}_{t-1}\|_1$ by Cauchy-Schwartz inequality because $\mu_i(\boldsymbol{\nu}_t) \in [0,1]$ and $r_i \in (0,1]$ for all $t \in [T]$ and i. Combining these two cases, we have

$$\mathbb{E}[\psi(\mathbf{b}_t) - \psi(\mathbf{b}_{t-1})|\mathcal{H}_{t-1}] \leq \mathbb{1}[\mathcal{B}_T]\left(-\frac{\epsilon}{2}\|\mathbf{b}_{t-1}\|_1\right) + \mathbb{1}[\mathcal{B}_T^c]\|\mathbf{b}_{t-1}\|_1 + \mathcal{O}(1).$$

Summing the preceding inequality over $\sqrt{T} \leq t \leq T_A$ and taking the outer expectation, we obtain

$$\begin{aligned}\mathbb{E}\left[\sum_{t=\sqrt{T}}^{T_A} \mathbb{E}[\psi(\mathbf{b}_t) - \psi(\mathbf{b}_{t-1})|\mathcal{H}_{t-1}]\right] \leq -\frac{\epsilon}{2}\mathbb{E}\left[\sum_{t=\sqrt{T}-1}^{T_A-1} \|\mathbf{b}_t\|_1 \cdot \mathbb{1}[\mathcal{B}_T]\right] + \\ \mathbb{E}\left[\sum_{t=\sqrt{T}-1}^{T_A-1} \|\mathbf{b}_t\|_1 \cdot \mathbb{1}[\mathcal{B}_T^c]\right] + \mathcal{O}(\mathbb{E}[T_A - \sqrt{T}]).\end{aligned} \tag{EC.119}$$

[49] Note that our high-probability results hold for any value of T_A.

Now define $X_t := \psi(\mathbf{b}_t) - \psi(\mathbf{b}_{t-1})$ and $Y_t = X_t - \mathbb{E}[X_t|\mathcal{H}_{t-1}]$ with $Y_0 = 0$. It is straightforward to see that $\{Y_t\}_{t=1}^T$ is a martingale difference sequence with respect to $\{\mathcal{H}_t\}_{t=1}^T$. Since T_A is a bounded stopping time with respect to $\{\mathcal{H}_t\}$, the optional stopping theorem implies

$$\mathbb{E}\left[\sum_{t=\sqrt{T}}^{T_A} \mathbb{E}[\psi(\mathbf{b}_t) - \psi(\mathbf{b}_{t-1})|\mathcal{H}_{t-1}]\right] = \mathbb{E}\left[\sum_{t=\sqrt{T}}^{T_A} [\psi(\mathbf{b}_t) - \psi(\mathbf{b}_{t-1})]\right] = \mathbb{E}[\psi(\mathbf{b}_{T_A}) - \psi(\mathbf{b}_{\sqrt{T}-1})].$$

Combining this with (EC.119), we have

$$\frac{\epsilon}{2}\mathbb{E}\left[\sum_{t=\sqrt{T}-1}^{T_A-1} \|\mathbf{b}_t\|_1 \cdot \mathbb{1}[\mathcal{B}_T]\right] \le \mathbb{E}[\psi(\mathbf{b}_{\sqrt{T}-1}) - \psi(\mathbf{b}_{T_A})] + \mathbb{E}\left[\sum_{t=\sqrt{T}-1}^{T_A-1} \|\mathbf{b}_t\|_1 \cdot \mathbb{1}[\mathcal{B}_T^c]\right] + \mathcal{O}(\mathbb{E}[T_A - \sqrt{T}]). \tag{EC.120}$$

Each term on the right-hand side is $\mathcal{O}(T)$. To see this, by taking the worst case growth of the backlog, we have $\mathbf{b}_t \le \mathcal{O}(t)$ and therefore $\psi(\mathbf{b}_{\sqrt{T}-1}) \le \mathcal{O}(T)$. The second term is again $\mathcal{O}(T)$ because $\Pr[\mathcal{B}_T^c] \le \mathcal{O}(1/T)$ from Lemma EC.43 (along with the worst case growth of the build-up). Finally, the last term in line (EC.120) is $\mathcal{O}(T)$ since $T_A \le T$ for every sample path. The proof is complete by diving both sides of line (EC.120) by $(\epsilon/2)$.

EC.8.4.4. Proof of Claim EC.46

We will use the inequality (EC.120) in Appendix EC.8.4.3. For ease of reference, we restate it in the following:

$$\frac{\epsilon}{2}\mathbb{E}\left[\sum_{t=\sqrt{T}-1}^{T_A-1} \|\mathbf{b}_t\|_1 \cdot \mathbb{1}[\mathcal{B}_T]\right] \le \mathbb{E}[\psi(\mathbf{b}_{\sqrt{T}-1}) - \psi(\mathbf{b}_{T_A})] + \mathbb{E}\left[\sum_{t=\sqrt{T}-1}^{T_A-1} \|\mathbf{b}_t\|_1 \cdot \mathbb{1}[\mathcal{B}_T^c]\right] + \mathcal{O}(\mathbb{E}[T_A - \sqrt{T}]).$$

Note that the left-hand side is non-negative. Hence, moving $\mathbb{E}[\psi(\mathbf{b}_{T_A})]$ to the left-hand side, we have

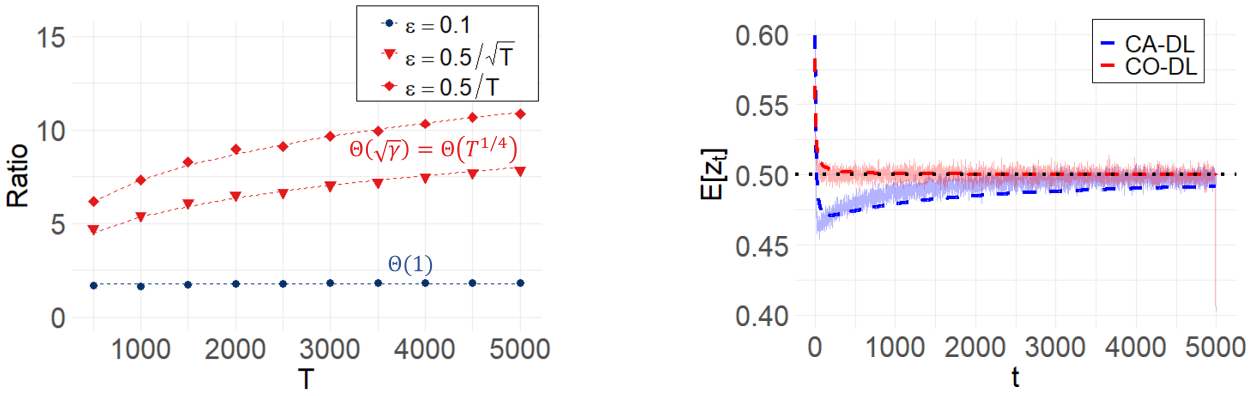
$$\mathbb{E}[\psi(\mathbf{b}_{T_A})] \le \mathbb{E}[\psi(\mathbf{b}_{\sqrt{T}-1})] + \mathbb{E}\left[\sum_{t=\sqrt{T}-1}^{T_A-1} \|\mathbf{b}_t\|_1 \cdot \mathbb{1}[\mathcal{B}_T^c]\right] + \mathcal{O}(\mathbb{E}[T_A - \sqrt{T}]).$$

In the proof of Lemma EC.44, we have already shown that each term in the right-hand side is $\mathcal{O}(T)$ (see the paragraph following inequality (EC.120)). The proof is complete by recalling the definition of $\psi(\mathbf{b}_{T_A}) = \frac{1}{2}\|\mathbf{b}_{T_A}\|_2^2$ and $\|\mathbf{b}_{T_A}\|_1^2 \le m\|\mathbf{b}_{T_A}\|_2^2$ by Cauchy-Schwartz inequality.

EC.9. Missing Details and Proof of Theorem 3

EC.9.1. Numerical Illustration of Theorem 3

In this section, we provide numerical illustration for the lower bound of regret in Theorem 3 by simulating our two algorithms. We consider the same one-affiliate instance as in Example 1. For each $T \in \{500, 1000, ..., 5000\}$, we use $\gamma = \sqrt{T}$ and $\epsilon \in \{0.1, \frac{0.5}{\sqrt{T}}, \frac{0.5}{T}\}$, where the first value of ϵ belongs to the stable regime and the other two to the near-critical regime. By using the notation of $\mathsf{Diff}^\pi := \mathbb{E}[\mathsf{OPT}(\gamma)] - \mathbb{E}[\mathsf{ALG}^\pi(\gamma)]$ for an algorithm π (note that α does not play any role in this instance because there are no tied cases), we display $\mathsf{Diff}^{\texttt{CO-DL}}/\mathsf{Diff}^{\texttt{CA-DL}}$ in Figure EC.1a (as a function of T). Consistent



(a) $\text{Diff}^{\texttt{CO-DL}}/\text{Diff}^{\texttt{CA-DL}}$ (b) $\mathbb{E}[z_t]$ for $T = 5000$ and $\epsilon = 0.5/\sqrt{T}$

Figure EC.1 Numerical Illustrations for Theorem 3 with 1000 sample paths and $\gamma = \sqrt{T}$. We use notation $\text{Diff}^{\pi} := \mathbb{E}[\text{OPT}(\gamma)] - \mathbb{E}[\text{ALG}^{\pi}(\gamma)]$. The learning rate of CA-DL is $\eta = \frac{1}{\sqrt{T}}$, with $\zeta = \frac{10}{\sqrt{T}}$ ($\zeta = \sqrt{\frac{\gamma}{T}}$, resp.) under the stable regime (the near-critical regime, resp.). The learning rate of CO-DL is $\eta_t = \frac{1}{\sqrt{t}}$ under both regimes.

with Theorem 3 and Corollary 1, we note that the ratios (two red curves) diverge in the order of $\sqrt{\gamma} = T^{1/4}$ under the near-critical regimes. In contrast, the ratio under the stable regime (blue curve) remains constant, which is also consistent with Theorems 1 and 2. However, we observe that CA-DL still achieves a lower regret than CO-DL by a constant factor (roughly 70%).

The main reason behind the performance difference lies in the endogenous arrival rates that each algorithm's decision induces. More specifically, Figure EC.1b illustrates the empirical average of z_t (solid line) for $T = 5000$ and $\epsilon = 0.5/\sqrt{T}$, which serves as a proxy for the (expected) endogenous arrival rates. The arrival rate of CO-DL quickly converges to $\rho = 0.5$, as evident from the cumulative time-average (dashed red line). In contrast, CA-DL induces an arrival rate strictly less than $\rho = 0.5$ (dashed blue line). Under the stable regime, maintaining an induced arrival rate of ρ ensures a constant average backlog. However, in the near-critical regime, we can show that the approach of CO-DL results in a time-average backlog of $\Omega(T^{1/2})$ for this specific instance (see Lemma EC.54). To gain some intuition, consider momentarily that the algorithm "exactly" induces the matching rate of ρ at every period. Because $\epsilon = \mathcal{O}(1/\sqrt{T})$, a simple anti-concentration result implies that, with constant probability, there remains $\Omega(\sqrt{t})$ backlog at each period t. By contrast, by penalizing the current backlog level, CA-DL achieves a better order of the backlog (and thus, a better order of regret).

EC.9.2. Proof of Theorem 3

Proof Sketch of Theorem 3: For the proof, we revisit the instance considered in Example 1 (we again omit the subscript of $i = 1$ for brevity). That is, we have $m = 1$ and $\rho = 0.5$. The reward w_t is an i.i.d sample from the uniform distribution on interval $(0, 1)$ and there is no tied case.

The proof consists of two steps. First, we establish that the value of Algorithm 2 (i.e. CO-DL) on this instance is at most $\frac{3}{8}T - \Theta(\gamma\sqrt{T})$. To prove this, we show that (i) the expected reward of CO-DL is at

most $\frac{3}{8}T$ and (ii) the average backlog of CO-DL is $\Omega(\sqrt{T})$ in expectation. In the second step, we show that the expected value of the optimal offline (Definition 2) is at least $\frac{3}{8}T - \Theta(\sqrt{\gamma T} + \gamma)$. Combining (and omitting the lower-order terms), we complete the proof of Theorem 3. In the following, we outline each step of the proof. (Throughout the proof, we use z_t and b_t to denote the matching decision and backlog of CO-DL at time t, respectively.)

Step 1: Upper Bound the Value of CO-DL.

The following lemma establishes an upper bound on the expected total reward of CO-DL.

LEMMA EC.52. $\mathbb{E}\left[\sum_{t=1}^{T} w_t z_t\right] \le \frac{3}{8}T$

Proof of Lemma EC.52. The proof is a direct consequence of Claim EC.31-(c) (Appendix EC.8.2). The optimal solution for the static dual problem (Definition 6) is such that $\theta^* + \lambda^* = 0.5$ at which a per-period expected reward is $\mathbb{E}[w_t \mathbb{1}[w_t \ge 0.5]] = \frac{3}{8}$. Hence, the result follows. □

Next, we lower-bound the average backlog of CO-DL by $\Omega(\sqrt{T})$ in expectation. The following lemma establishes a high probability bound on the total number of matched cases, which we prove in Appendix EC.9.2.1.

LEMMA EC.53. *There exists a constant $\kappa > 0$ such that, for any fixed $t \le 0.5T$ and $\delta \in (0,1)$,*

$$\mathbb{P}\left[\sum_{\tau=1}^{t} z_\tau \ge 0.5t - \kappa\sqrt{t\log(1/\delta)}\right] \ge 1 - \delta - \mathcal{O}(1/t).$$

We now use Lemma EC.53 to obtain a lower bound on the expected backlog.

LEMMA EC.54. *There exists constants $a > 0$ and $t(a)$, such that, whenever $T \ge 2t(a)$ and $\epsilon \le \frac{a}{4\sqrt{T}}$, we have:*

$$\mathbb{E}[b_t] \ge \frac{a\sqrt{t}\Phi(-3\sqrt{2}a)}{16}$$

for all $t(a) \le t \le \frac{T}{2}$. Here $\Phi(\cdot)$ is the cumulative distribution function of the standard normal random variable. The exact form of the constant $t(a)$, as a function of the constant $a > 0$, is given in the proof.

The proof of this lemma is presented in Appendix EC.9.2.2. For the proof, we use the following simple anti-concentration result. Let S_t be the total number of periods that the server is available up to time t. Under the near-critical regime, there is a constant probability that $S_t \le 0.5t - \Theta(\sqrt{t})$. Combining this with the lower bound on the total number of matched cases established in Lemma EC.53, we can show that at least $\Theta(\sqrt{t})$ cases remain in the backlog in expectation for each period $t(a) \le t \le \frac{T}{2}$. Finally, we note that summing the inequality Lemma EC.54 for all $t(a) \le t \le \frac{T}{2}$ implies the $\Omega(\sqrt{T})$ average backlog of Algorithm 2 in expectation.

Step 2: Lower-bounding the Value of the Optimal Offline. The following lemma lower-bound the value of the offline.

LEMMA EC.55. $\mathbb{E}[OPT(\gamma)] \ge \frac{3}{8}T - \Theta(\sqrt{\gamma T} + \gamma)$

The proof of this lemma can be found in Appendix EC.9.2.3. For the proof, we study the expected objective value of the following feasible solution for each sample path: let $\delta := \sqrt{\gamma/T}$ and define $z_t^* = \mathbb{1}[w_t \geq 0.5+\delta]$ up to $t \leq T_A$ and $z_t^* = 0$ for all $t > T_A$,[50] where the stopping time T_A is defined as $T_A = \min\{t \leq T : \sum_{\tau=1}^{t} z_t^* \geq 0.5T\}$. We then show that (i) the total expected reward of this feasible solution is at least $\frac{3}{8}T - \Theta(\sqrt{\gamma T}+\gamma)$ and (ii) the expected time-average backlog is $O(\sqrt{T/\gamma}+1)$.

EC.9.2.1. Proof of Lemma EC.53

For the proof, we first establish the following high probability bound.

PROPOSITION EC.56. *Suppose the arrival distribution is such that there are no tied cases. Then there exists a constant $M > 0$ such that, for any fixed $\delta \in (0,1)$ and $t \leq T_A$, the matching profile $\{\mathbf{z}_\tau\}_{\tau=1}^t$ of `CO-DL` satisfies that, with probability at least $1-\delta$,*

$$0 \leq \sum_{i=1}^{m} \nu_{t,i}^* \left(\sum_{\tau=1}^{t} z_{\tau,i} - t\rho_i \right) + M\sqrt{t\log(1/\delta)} \tag{EC.121}$$

where $\boldsymbol{\nu}_t^* = (\nu_{t,i})_{i=1}^m$ is a random variable defined by

$$\boldsymbol{\nu}_t^* \in \arg\min_{\boldsymbol{\nu}\in\mathbb{R}_+^m} \sum_{t=1}^{T} \{ \max_{\mathbf{z}\in\mathcal{Z}(i^\dagger)} (\mathbf{w}_t - \boldsymbol{\nu})\cdot\mathbf{z} + \boldsymbol{\rho}\cdot\boldsymbol{\nu} \}. \tag{EC.122}$$

The proof is fairly intricate, and we defer it to the end of this section. But first, we complete the proof of Lemma EC.53 using Proposition EC.56. Recall that the instance considered in Example 1 consists of one actual affiliate. Hence, we apply Proposition EC.56 to this instance and deduce that, for any $t \leq 0.5T$,

$$0 \leq \nu_t^* \left(\sum_{\tau=1}^{t} z_\tau - 0.5t \right) + M\sqrt{t\log(1/\delta)} \tag{EC.123}$$

with probability at least $1-\delta$ and some constant $M > 0$. Furthermore, because the reward follows the uniform distribution on $(0,1)$, the random variable ν_t^* is the sample median of t i.i.d. samples from the uniform distribution on $(0,1)$ (see Footnote 27). From Casella and Berger (2021) Example 5.4.5., the expectation and variance of ν_t^* is given by $\mathbb{E}[\nu_t^*] = 1/2$ and $\mathbb{V}[\nu_t^*] = \Theta(1/t)$, respectively. Hence, a straightforward application of ChebyShev's inequality implies that $\mathbb{P}[\nu_t^* \geq \frac{1}{4}] \geq 1 - \mathcal{O}(1/t)$. Combining this with line (EC.123) with the union bound, we conclude that, with probability at least $1-\delta-\mathcal{O}(1/t)$, we have

$$0 \leq \left(\sum_{\tau=1}^{t} z_\tau - 0.5t \right) + 4M\sqrt{t\log(1/\delta)}, \tag{EC.124}$$

which leads to the desired result of Lemma EC.53 by taking $\kappa = 4M$.

Finally, we conclude this section with the proof of Proposition EC.56.

[50] Note that this is $\delta = o(1)$ because we focus on $\gamma = o(T)$ in light of Proposition 2

Proof of Proposition EC.56. First, we provide some preliminaries that will be useful throughout the proof. With a slight abuse of notations, we define the following static problem.

$$D(\boldsymbol{\nu}) := \mathbb{E}\left[\max_{\mathbf{z}\in\mathcal{Z}(i^\dagger)} (\mathbf{w}-\boldsymbol{\nu})\cdot\mathbf{z} + \boldsymbol{\rho}\cdot\boldsymbol{\nu}\right]$$

$$D(\boldsymbol{\nu}^*) = \min_{\boldsymbol{\nu}\in\mathbb{R}^m_+} D(\boldsymbol{\nu}). \tag{EC.125}$$

Given $\boldsymbol{\nu}\in\mathbb{R}^m_+$, we similarly define the dual-based primal decision with respect to arrival type $\mathbf{A}$ as follows.

$$\tilde{\mathbf{z}}(\boldsymbol{\nu},\mathbf{A}) := \arg\max_{\mathbf{z}\in\mathcal{Z}(i^\dagger)} (\mathbf{w}-\boldsymbol{\nu})\cdot\mathbf{z}$$
$$\boldsymbol{\mu}(\boldsymbol{\nu}) := \mathbb{E}[\tilde{\mathbf{z}}(\boldsymbol{\nu},\mathbf{A})].$$

By following a similar step to the proof of claim EC.31 (Appendix EC.8.2), we can show the following properties of the above static problem. The proof mirrors that of Claim EC.31 and we omit it for brevity.

Claim EC.57. *For $D(\cdot)$ and $\boldsymbol{\nu}^*$ defined in line (EC.125), the following property holds:*

(a) $\|\boldsymbol{\nu}^*\|_\infty \le 1$.

(b) $\mu_i(\boldsymbol{\nu}^*) \le \rho_i$ *for all* $i\in[m]$ *with equality if* $\nu_i^* > 0$.

(c) $D(\boldsymbol{\nu}^*) = \mathbb{E}[\mathbf{w}\cdot\tilde{\mathbf{z}}(\boldsymbol{\nu}^*,\mathbf{A})]$

We will also use the following result on the sample average approximation of the stochastic optimization problem.

Lemma EC.58 **(Proposition 1 of Guigues et al. (2017))**. *Let $\mathcal{X}$ be a nonempty bounded convex set. Consider the following stochastic optimization problem:*

$$f^* = \min_{\mathbf{x}\in\mathcal{X}} \mathbb{E}_{\mathbf{A}\sim\mathcal{F}}[f(\mathbf{x},\mathbf{A})]$$

where $\mathbf{A}$ is a random vector from probability distribution $\mathcal{F}$ on support $\mathcal{A}$. Given i.i.d samples $\mathbf{A}_1,...,\mathbf{A}_t\sim\mathcal{F}$, consider the following sample-average approximation of f^*:

$$\hat{f}_t := \min_{\mathbf{x}\in\mathcal{X}} \frac{1}{t}\sum_{\tau=1}^{t} f(\mathbf{x},\mathbf{A}_\tau).$$

Assume that there exists positive constant M_1 such that $|f(\mathbf{x},\mathbf{A}) - \mathbb{E}_{\mathbf{A}\sim\mathcal{F}}[f(\mathbf{x},\mathbf{A})]| \le M_1$ for all $\mathbf{x}\in\mathcal{X}$ and $\mathbf{A}\in\mathcal{A}$. Then, for any given $\delta\in(0,1)$,

$$\mathbb{P}\left[\hat{f}_t \le f^* + 2M_1\sqrt{\frac{\log(1/\delta)}{t}}\right] \ge 1-\delta.$$

Main Proof. Fix an arbitrary time $t \le T_A$ and a sample path of arrivals $(\mathbf{A}_1, ..., \mathbf{A}_t)$.[51] Recall that we use $\mathbf{z}_\tau$ to denote the decision of CO-DL for arrival $\mathbf{A}_\tau$ for $\tau \in [T]$. Because CO-DL is unconstrained until $t \le T_A$, for any arbitrary time $\tau \le t$, we can write it as:

$$\mathbf{z}_\tau = \arg\max_{\mathbf{z} \in \mathcal{Z}(i_\tau^\dagger)} (\mathbf{w}_\tau - \boldsymbol{\theta}_\tau - \boldsymbol{\lambda}_\tau) \cdot \mathbf{z}. \tag{EC.126}$$

We will compare $\mathbf{z}_\tau$ against an alternative decision vector $\mathbf{z}_\tau^*$ defined as follows:

$$\mathbf{z}_\tau^* := \arg\max_{\mathbf{z} \in \mathcal{Z}(i_\tau^\dagger)} (\mathbf{w}_\tau - \boldsymbol{\nu}^*) \cdot \mathbf{z} = \mathbf{z}(\boldsymbol{\nu}^*, \mathbf{A}_\tau) \tag{EC.127}$$

where $\boldsymbol{\nu}^*$ is the minimizer of the static problem (EC.125). Note that $\mathbf{z}_\tau^*$ is i.i.d. random variables because $\boldsymbol{\nu}^*$ only depends on the arrival distribution. By the optimality criterion of the modified algorithm (EC.126), we have

$$\sum_{\tau=1}^{t} (\mathbf{w}_\tau - \boldsymbol{\theta}_\tau - \boldsymbol{\lambda}_\tau) \cdot \mathbf{z}_\tau + \boldsymbol{\rho} \cdot (\boldsymbol{\theta}_\tau + \boldsymbol{\lambda}_\tau) \ge \sum_{\tau=1}^{t} (\mathbf{w}_\tau - \boldsymbol{\theta}_\tau - \boldsymbol{\lambda}_\tau) \cdot \mathbf{z}_\tau^* + \boldsymbol{\rho} \cdot (\boldsymbol{\theta}_\tau + \boldsymbol{\lambda}_\tau). \tag{EC.128}$$

From Claim EC.4 (Appendix EC.6), there exists $\boldsymbol{\theta}^\star \in [0, \alpha]^m$ for which

$$\sum_{\tau=1}^{t} \boldsymbol{\theta}^\star \cdot (\mathbf{z}_\tau - \boldsymbol{\rho}) = \alpha \sum_{i=1}^{m} \left(\sum_{\tau=1}^{t} z_{\tau,i} - t\rho_i \right)_+ .$$

Hence, a straightforward manipulation of line (EC.128) leads to

$$\begin{aligned} \alpha \sum_{i=1}^{m} \left(\sum_{\tau=1}^{t} z_{\tau,i} - t\rho_i \right)_+ \le & \underbrace{\sum_{\tau=1}^{t} \mathbf{w}_\tau \cdot \mathbf{z}_\tau - \sum_{\tau=1}^{t} \mathbf{w}_\tau \cdot \mathbf{z}_\tau^*}_{(\mathsf{A})} + \underbrace{\sum_{\tau=1}^{t} \boldsymbol{\theta}_\tau \cdot (\mathbf{z}_\tau^* - \boldsymbol{\rho}) + \sum_{\tau=1}^{t} \boldsymbol{\lambda}_\tau \cdot (\mathbf{z}_\tau^* - \boldsymbol{\rho})}_{(\mathsf{B})} + \\ & \underbrace{\sum_{\tau=1}^{t} \boldsymbol{\theta}_\tau \cdot (\boldsymbol{\rho} - \mathbf{z}_\tau) - \sum_{\tau=1}^{t} \boldsymbol{\theta}^\star \cdot (\boldsymbol{\rho} - \mathbf{z}_\tau) + \sum_{\tau=1}^{t} \boldsymbol{\lambda}_\tau \cdot (\boldsymbol{\rho} - \mathbf{z}_\tau)}_{(\mathsf{C})} \end{aligned} \tag{EC.129}$$

We prove the main result through four claims. In the first three claims (Claim EC.59-EC.61), we obtain the high-probability bound of each term $(\mathsf{A}) - (\mathsf{C})$, respectively.

Claim EC.59 **(Bound (A))**. *There exists constant M_A such that, for any given $\delta \in (0,1)$,*

$$(A) \le \sum_{i=1}^{m} \nu_{t,i}^* \left(\sum_{\tau=1}^{t} z_{\tau,i} - t\rho_i \right) + M_A \sqrt{t \log(1/\delta)}$$

with probability at least $1 - \delta$ where $\boldsymbol{\nu}_t^*$ is defined in line (EC.122).

Proof of Claim EC.59. Consider the following linear program defined on the arrival sequence $(\mathbf{A}_1, ..., \mathbf{A}_t)$:

$$\overline{\mathsf{OPT}}_t(t\boldsymbol{\rho}) := \max_{\mathbf{z}_\tau \in \mathcal{Z}(i_\tau^\dagger)} \sum_{\tau=1}^{t} \mathbf{w}_\tau \cdot \mathbf{z}_\tau \quad \text{s.t.} \quad \sum_{\tau=1}^{t} \mathbf{z}_\tau \le t\boldsymbol{\rho} \qquad [\boldsymbol{\nu}_t^*] \tag{EC.130}$$

[51] Recall that we consider an arrival distribution with no tied cases. Further, the service sequence never affects the decision of CO-DL (as it is oblivious to the service sequence).

We use subscript t in the optimal objective value to emphasize that the optimization problem is defined on the arrival up to time t. In the similar spirit, the optimal value of the above offline depends on the right-hand side of the capacity constraint, which we also emphasized as the argument of $\overline{\mathsf{OPT}}_t(\cdot)$. The dual variable corresponding to the constraint (EC.130) (at the right hand side value $t\boldsymbol{\rho}$) is denoted by $\boldsymbol{\nu}_t^* \geq 0$. The above program has a feasible solution for every sample path because we consider the arrival distribution without tied cases.

Note that, by definition, $\{\mathbf{z}_\tau\}_{\tau=1}^t$ is a feasible solution for $\overline{\mathsf{OPT}}_t\left(\sum_{\tau=1}^t \mathbf{z}_\tau\right)$. Hence, we have

$$\begin{aligned}\sum_{\tau=1}^t \mathbf{w}_\tau \cdot \mathbf{z}_\tau &\leq \overline{\mathsf{OPT}}_t\left(\sum_{\tau=1}^t \mathbf{z}_\tau\right)\\ &= \overline{\mathsf{OPT}}_t\left(t\boldsymbol{\rho} + \sum_{\tau=1}^t \mathbf{z}_\tau - t\boldsymbol{\rho}\right)\\ &\leq \overline{\mathsf{OPT}}_t(t\boldsymbol{\rho}) + \sum_{i=1}^m \nu_{t,i}^*\left(\sum_{\tau=1}^t z_{\tau,i} - t\rho_i\right)\end{aligned} \tag{EC.131}$$

where in the last line we used the fact that (i) $\overline{\mathsf{OPT}}_t(\cdot)$ is concave and increasing in each coordinate of its argument (see, for example, Section 5.6.2 of Boyd et al. (2004)) and (ii) $\boldsymbol{\nu}_t^*$ is a gradient of $\overline{\mathsf{OPT}}_t(\cdot)$ evaluated at the right-hand side value of $t\boldsymbol{\rho}$ in line (EC.130).

We now turn our attention to lower-bounding $\sum_{\tau=1}^t \mathbf{w}_\tau \cdot \mathbf{z}_\tau^*$. Here, we will crucially rely on the properties of the static problem in Claim EC.57. By definition of $\mathbf{z}_\tau^*$ and i.i.d nature of the arrival sequence, each $\mathbf{w}_\tau \cdot \mathbf{z}_\tau^*$ is i.i.d random variables with $|\mathbf{w}_\tau \cdot \mathbf{z}_\tau^*| \leq 1$. Furthermore, from Claim EC.57-(c), the identical mean is $\mathbb{E}[\mathbf{w}_\tau \cdot \mathbf{z}_\tau^*] = D(\boldsymbol{\nu}^*)$ for all $\tau \in [t]$. Hence, the Azuma-Hoeffding inequality implies that, with probability at least $1-\delta$,

$$\sum_{\tau=1}^t \mathbf{w}_\tau \cdot \mathbf{z}_\tau^* \geq tD(\boldsymbol{\nu}^*) - \sqrt{\frac{t}{2}\log(1/\delta)}.$$

Combining the above bound with line (EC.131), we have

$$(\mathsf{A}) \leq \underbrace{\overline{\mathsf{OPT}}_t(t\boldsymbol{\rho}) - tD(\boldsymbol{\nu}^*)}_{\clubsuit} + \sum_{i=1}^m \nu_{t,i}^*\left(\sum_{\tau=1}^t z_{\tau,i} - t\rho_i\right) + \sqrt{\frac{t}{2}\log(1/\delta)}. \tag{EC.132}$$

To complete the proof of the claim, we obtain the high-probability bound on $\clubsuit$ by invoking Lemma EC.58. First, let us first define

$$D(\boldsymbol{\nu}; \mathbf{A}) := \max_{\mathbf{z} \in \mathcal{Z}(i^\dagger)} (\mathbf{w} - \boldsymbol{\nu}) \cdot \mathbf{z} + \boldsymbol{\rho} \cdot \boldsymbol{\nu}.$$

Given this notation, we first note that $D(\boldsymbol{\nu}^*)$ defined in line (EC.125) can be written as

$$D(\boldsymbol{\nu}^*) = \min_{\boldsymbol{\nu} \in [0,1]^m} \mathbb{E}[D(\boldsymbol{\nu}; \mathbf{A})]$$

where we used Claim EC.57-(a) to restrict the domain to $[0,1]^m$ without loss of optimality. On the other hand, we have:

$$\frac{\overline{\mathsf{OPT}}_t(t\boldsymbol{\rho})}{t} = \min_{\boldsymbol{\nu}_t \in \mathbb{R}_+^m} \frac{1}{t}\sum_{\tau=1}^{t} D(\boldsymbol{\nu}_t; \mathbf{A}_\tau) = \min_{\boldsymbol{\nu}_t \in [0,1]^m} \frac{1}{t}\sum_{\tau=1}^{t} D(\boldsymbol{\nu}_t; \mathbf{A}_\tau). \tag{EC.133}$$

The first equality follows from writing the dual of $\overline{\mathsf{OPT}}_t(t\boldsymbol{\rho})$. For the second equality, we again used the fact that the optimal value $\boldsymbol{\nu}_t^*$ is always in $[0,1]^m$ without loss of optimality. [52]

Hence, because the arrivals are i.i.d samples, we observe that $\frac{\overline{\mathsf{OPT}}_t(t\boldsymbol{\rho})}{t}$ is the sample average approximation for $D(\boldsymbol{\nu}^*)$ with t i.i.d samples $(\mathbf{A}_1, ..., \mathbf{A}_t)$. Finally, it is straightforward to show that $|D(\boldsymbol{\nu}; \mathbf{A})| \le 3$ for all $\boldsymbol{\nu} \in [0,1]^m$, and $\mathbf{A} \in \mathcal{A}$. Hence, $|D(\boldsymbol{\nu}; \mathbf{A}) - \mathbb{E}[D(\boldsymbol{\nu}; \mathbf{A})]| \le 6$ for all $\boldsymbol{\nu} \in [0,1]^m$ and $\mathbf{A} \in \mathcal{A}$. Invoking Lemma EC.58 with $M_1 = 6$, we obtain

$$\mathbb{P}\left[\frac{\overline{\mathsf{OPT}}_t(t\boldsymbol{\rho})}{t} \le D(\boldsymbol{\nu}^*) + 12\sqrt{\frac{\log(1/\delta)}{t}}\right] \ge 1-\delta.$$

This equivalently implies that, with probability at least $1-\delta$, the term ♣ in line (EC.132) is at most $12\sqrt{t\log(1/\delta)}$. Plugging this bound in line (EC.132) and taking the union bound, with probability at least $1-2\delta$, we have

$$(\mathsf{A}) \le \sum_{i=1}^{m} \nu_{t,i}^* \left(\sum_{\tau=1}^{t} z_{\tau,i} - t\rho_i\right) + M_{\mathsf{A}}\sqrt{t\log(1/\delta)}$$

where $M_{\mathsf{A}} = 12 + \sqrt{\frac{1}{2}}$. The proof is complete by replacing δ with $\delta/2$. □

The following lemma shows the high-probability bound of term (B) in line (EC.129).

Claim EC.60 **(Bound (B))**. *There exists constant M_B such that, for any given $\delta \in (0,1)$, we have*

$$\mathbb{P}[(B) \le M_B\sqrt{t\log(1/\delta)}] \ge 1-\delta.$$

Proof of Claim EC.60. Let $X_s := \boldsymbol{\theta}_s \cdot (\mathbf{z}_s^* - \boldsymbol{\rho}) + \boldsymbol{\lambda}_s \cdot (\mathbf{z}_s^* - \boldsymbol{\rho})$ and $Y_\tau := \sum_{s=1}^{\tau} X_s$, with $X_0 := 0$ and $Y_0 := 0$. Note that Y_τ is $\mathcal{H}_\tau$-measurable. We first note that Y_τ is supermartingale with respect to $\mathcal{H}_\tau$ because:

$$\begin{aligned}
\mathbb{E}[X_\tau|\mathcal{H}_{\tau-1}] &= \boldsymbol{\theta}_\tau \cdot (\mathbb{E}[\mathbf{z}_\tau^*|\mathcal{H}_{\tau-1}] - \boldsymbol{\rho}) + \boldsymbol{\lambda}_\tau \cdot (\mathbb{E}[\mathbf{z}_\tau^*|\mathcal{H}_{\tau-1}] - \boldsymbol{\rho}) \\
&= \boldsymbol{\theta}_\tau \cdot (\mathbb{E}[\mathbf{z}_\tau^*] - \boldsymbol{\rho}) + \boldsymbol{\lambda}_\tau \cdot (\mathbb{E}[\mathbf{z}_\tau^*] - \boldsymbol{\rho}) \\
&\le 0.
\end{aligned}$$

The first line is because $\boldsymbol{\theta}_\tau$ and $\boldsymbol{\lambda}_\tau$ is $\mathcal{H}_{\tau-1}$-measurable. The second line is because each $\mathbf{z}_\tau^*$ is i.i.d random variable. The final line is because the identical mean is $\mathbb{E}[\mathbf{z}_\tau^*] \le \boldsymbol{\rho}$ from Claim EC.57-(b). Furthermore, the Cauchy-Schwartz inequality gives $|X_\tau| \le 2(\alpha + \bar{\lambda})$ almost surely where $\bar{\lambda} := \frac{1+2\alpha}{\underline{\rho}}$

[52] Consider the minimization problem $\min_{\boldsymbol{\nu}_t \in \mathbb{R}_+^m} \sum_{\tau=1}^{t} D(\boldsymbol{\nu}_t; \mathbf{A}_\tau)$. Let $\boldsymbol{\nu}_t^*$ be the optimal solution of this problem. Suppose for a contradiction that there exists i such that $\nu_{t,i}^* > 1$. Such location i is not assigned any free case since the reward is at most one. Hence, one can always consider an alternative solution that reduces $\nu_{t,i}^*$ infinitesimally and strictly decreases the objective value.

because the dual variables $(\boldsymbol{\theta}, \boldsymbol{\lambda})$ are restricted to $\mathcal{V} = [0,\alpha]^m \times [0,\bar{\lambda}]^m$. Hence, we use Azuma-Hoeffding inequality for super martingale to obtain

$$\mathbb{P}\left[Y_t \le \sqrt{2t(\alpha+\bar{\lambda})\log(1/\delta)}\right] \ge 1-\delta$$

for any given $\delta \in (0,1)$. This completes the proof. □

The following lemma obtains a deterministic bound on term (C) of line (EC.129).

Claim EC.61 **(Bound (C))**. *There exists a constant M_{C} for which (C) $\le M_{\mathsf{C}}\sqrt{t}$ for every sample path.*

Proof of Claim EC.61. The proof directly follows from the adversarial regret guarantee of the online mirror descent (Claim EC.3). In particular, let

$$R_{\boldsymbol{\theta}}(t) := \sum_{\tau=1}^{t} \boldsymbol{\theta}_\tau \cdot (\boldsymbol{\rho} - \mathbf{z}_\tau) - \sum_{\tau=1}^{t} \boldsymbol{\theta}^\star \cdot (\boldsymbol{\rho} - \mathbf{z}_\tau)$$
$$R_{\lambda}(t) := \sum_{\tau=1}^{t} \boldsymbol{\lambda}_\tau \cdot (\boldsymbol{\rho} - \mathbf{z}_\tau)$$

One can follow the same line of proofs as Claim EC.48 (Appendix EC.8.4.1) to show that

$$R_{\boldsymbol{\theta}}(t) \le 2\alpha \sum_{\tau=1}^{t} \eta_\tau + \frac{\bar{V}_{\boldsymbol{\theta}}}{\eta_t}$$
$$R_{\boldsymbol{\lambda}}(t) \le 2\bar{\lambda} \sum_{\tau=1}^{t} \eta_\tau + \frac{\bar{V}_{\lambda}}{\eta_t}$$

where $\bar{\lambda} := \frac{1+2\alpha}{\rho}$, $h(\cdot)$ is the negative entroypy function, $\bar{V}_\theta = \max_{\tau\in[t]} V_h(\boldsymbol{\theta}^\star, \boldsymbol{\theta}_\tau)$, and $\bar{V}_\lambda = \max_{\tau\in[t]} V_h(\mathbf{0}, \boldsymbol{\lambda}_\tau)$. Because the dual variables are all restricted to the bounded domain $\mathcal{V} = [0,\alpha]^m \times [0,\bar{\lambda}]^m$, both $\bar{V}_\theta$ and $\bar{V}_\lambda$ are bounded by some constant $\bar{V}$. Furthermore, recall that the step size of Algorithm 2 is given by $\eta_\tau = \frac{k}{\sqrt{\tau}}$ for input constant $k > 0$. Hence, both terms above are upper bounded by $M_{\mathsf{C}}\sqrt{t}$ where a positive constant M_{C} only depends on the input of `CO-DL` (other than T). This completes the proof. □

To complete the proof of Proposition EC.56, we plug the high-probability bounds in the previous claims into line (EC.129) to obtain that, with probability at least $1-\delta$,

$$\alpha \sum_{i=1}^{m} \left(\sum_{\tau=1}^{t} z_{\tau,i} - t\rho_i\right)_+ \le \sum_{i=1}^{m} \nu^*_{t,i} \left(\sum_{\tau=1}^{t} z_{\tau,i} - t\rho_i\right) + M\sqrt{t\log(1/\delta)} \tag{EC.134}$$

where the constant $M > 0$ only depends on $(M_{\mathsf{A}}, M_{\mathsf{B}}, M_{\mathsf{C}})$ specified in Claims EC.59-EC.61. The proof is complete by non-negativity of the left-hand side. □

EC.9.2.2. Proof of Lemma EC.54

Recall that we are analyzing an instance with $m=1$. Hence, for brevity, we omit the subscript for $i=1$. Let $S_t=\sum_{\tau=1}^{t} s_\tau$ and $Z_t=\sum_{\tau=1}^{t} z_\tau$. We first obtain a simple bound on the backlog.

CLAIM EC.62. *$b_t \geq (Z_t - S_t)_+$ for all $t \geq 1$ for every sample path.*

Proof of Claim EC.62. The proof is by induction on t. The base case for $t=1$ is trivial because $b_0=0$. Now suppose that the claim is true for $t \geq 1$. Then we have

$$b_{t+1} = (b_t + z_{t+1} - s_{t+1})_+ \geq (Z_t - S_t + z_{t+1} - s_{t+1})_+ = (Z_{t+1} - S_{t+1})_+$$

where the first inequality follows from the induction hypothesis. This completes the proof. □

Fix a positive constant $a>0$, which we will specify later. For given constant $a>0$ and t, define an event

$$\begin{aligned}
\mathcal{Z}_{a,t} &= \{Z_t \geq 0.5t - a\sqrt{t}\} \\
\mathcal{S}_{a,t} &= \{0.5t - a\sqrt{t} - S_t \geq \frac{a}{4}\sqrt{t}\} \\
\mathcal{E}_{a,t} &:= \mathcal{Z}_{a,t} \cap \mathcal{S}_{a,t}
\end{aligned}$$

First, from Claim EC.62, we note that

$$\mathbb{E}[b_t] \geq \mathbb{E}[(Z_t - S_t)_+] \geq \mathbb{P}[\mathcal{E}_{a,t}]\mathbb{E}[(Z_t - S_t)_+ | \mathcal{E}_{a,t}] \geq \mathbb{P}[\mathcal{E}_{a,t}] \times \frac{a\sqrt{t}}{4}.$$

where the last inequality follows from the definition of $\mathcal{E}_{a,t}$. Hence, it suffices to lower-bound $\mathbb{P}[\mathcal{E}_{a,t}]$ by $\frac{\Phi(-3\sqrt{2}a)}{4}$ for some constant $a>0$. First, note that the decision of `CO-DL` *does not* depend on the service (and therefore backlog) process. Hence, the event $\mathcal{Z}_{a,t}$ and $\mathcal{S}_{a,t}$ must be independent for Algorithm 2, implying that $\mathbb{P}[\mathcal{E}_{a,t}] = \mathbb{P}[\mathcal{Z}_{a,t}]\mathbb{P}[\mathcal{S}_{a,t}]$.

Thus, it suffices to separately lower-bound the probability of $\mathcal{Z}_{a,t}$ and $\mathcal{S}_{a,t}$. We first obtain a lower bound on the former. From Lemma EC.53, there exists constant $\kappa>0$ such that, for any given $a>0$ and $t \leq 0.5T$, we have

$$\mathbb{P}[\mathcal{Z}_{a,t}] \geq 1 - \exp\left(-\frac{a^2}{\kappa^2}\right) - \mathcal{O}(1/t). \tag{EC.135}$$

For a sufficiently large constant $a>0$ and another constant $g(a)$, the right-hand side is at least $\frac{1}{2}$ for all $t \geq g(a)$. For such constant $a>0$, we now turn our attention to lower-bounding $\mathbb{P}[\mathcal{S}_{a,t}]$ as a function of the constant $a>0$. By Berry-Esseen Theorem (see, for example, Theorem 3.4.17 of Durrett (2019)), there exists a constant $d>0$ such that for all $t \geq 1$,

$$\mathbb{P}\left[S_t \leq \mathbb{E}[S_t] - 3\sqrt{2}a\sqrt{\mathrm{Var}[S_t]}\right] \geq \Phi(-3\sqrt{2}a) - \frac{d}{\sqrt{t}}.$$

where $\mathbb{E}[S_t] = (0.5+\epsilon)t$, $\mathrm{Var}[S_t] = (0.25-\epsilon^2)t$, and $\Phi(\cdot)$ is the cumulative distribution function of the standard normal random variable. Moving the terms, the preceding inequality is equivalent to

$$\mathbb{P}\left[0.5t - a\sqrt{t} - S_t \geq 3\sqrt{2}a \times \sqrt{\mathrm{Var}[S_t]} - a\sqrt{t} - \epsilon t\right] \geq \Phi(-3\sqrt{2}a) - \frac{d}{\sqrt{t}}.$$

Let us define a constant $t(a)$ as

$$t(a) := \max\left\{\frac{4d^2}{\Phi^2(-3\sqrt{2}a)}, \frac{a^2}{2}, g(a)\right\} \tag{EC.136}$$

We now use the assumption $\epsilon \le \frac{a}{4\sqrt{T}}$ and $T \ge 2t(a)$ to prove the following:

$$\begin{aligned}
3\sqrt{2}a \times \sqrt{\mathrm{Var}[S_t]} - a\sqrt{t} - \epsilon t &= 3\sqrt{2}a\sqrt{(0.25-\epsilon^2)t} - a\sqrt{t} - \epsilon t \\
&\ge 3\sqrt{2}a\sqrt{\left(0.25 - \frac{a^2}{16T}\right)t} - a\sqrt{t} - \frac{at}{4\sqrt{T}} \\
&\ge \frac{a}{2}\sqrt{t} - \frac{at}{4\sqrt{T}} \\
&\ge \frac{a}{4}\sqrt{t}
\end{aligned}$$

In the second line, we used the assumption $\epsilon \le \frac{a}{4\sqrt{T}}$. In the third line, we used the assumption $T \ge t(a) \ge \frac{a^2}{2}$. The last line follows from $T \ge t$. Furthermore, for all $t \ge t(a) \ge \frac{4d^2}{(\Phi(-3\sqrt{2}a))^2}$, it is straightforward to check that $\Phi(-3\sqrt{2}a) - \frac{d}{\sqrt{t}} \ge \frac{\Phi(-3\sqrt{2}a)}{2}$. Hence, for any $t(a) \le t \le \frac{1}{2}T$, we conclude that

$$\mathbb{P}[\mathcal{S}_{a,t}] = \mathbb{P}\left[0.5t - a\sqrt{t} - S_t \ge \frac{a}{4}\sqrt{t}\right] \ge \mathbb{P}\left[0.5t - a\sqrt{t} - S_t \ge 3\sqrt{2}a \times \sqrt{\mathrm{Var}[S_t]} - a\sqrt{t} - \epsilon t\right] \ge \frac{\Phi(-3\sqrt{2}a)}{2}$$

Combining, we conclude that, there exists a constant $a > 0$ and $t(a)$ such that, for all $t \in [t(a), T/2]$,

$$\mathbb{P}[\mathcal{E}_{a,t}] = \mathbb{P}[\mathcal{Z}_{a,t}]\mathbb{P}[\mathcal{S}_{a,t}] \ge \frac{\Phi(-3\sqrt{2}a)}{4}.$$

whenever $T \ge 2t(a)$ and $\epsilon \le \frac{a}{4\sqrt{T}}$. This completes the proof.

EC.9.2.3. Proof of Lemma EC.55

Let $\delta := \sqrt{\gamma/T}$. Consider the following feasible solution: $z_t^* = \mathbb{1}[w_t \ge 0.5+\delta]$ up to $t \le T_A$ and $z_t^* = 0$ for all $t > T_A$, where the stopping time T_A is defined as $T_A = \min\{t \le T : \sum_{\tau=1}^t z_t^* \ge 0.5T\}$. We analyze the the expected objective value of this feasible solution, which will be the lower bound on the expected value of the optimal offline solution.

We first analyze the reward of this policy. Because (i) T_A is a bounded stopping time with respect to $\{\mathcal{H}_t\}$ and (ii) each $w_t z_t$ is independent with identical mean for $t \le T_A$, Wald's equation implies that

$$\mathbb{E}\Big[\sum_{t=1}^{T_A} w_t z_t^*\Big] = \mathbb{E}[T_A]\mathbb{E}[w_1 z_1^*] = \mathbb{E}[T_A]\int_{0.5+\delta}^1 w dw = \mathbb{E}[T_A]\Big(\frac{3}{8} - \frac{1}{2}\delta - \frac{1}{2}\delta^2\Big). \tag{EC.137}$$

where in the second equality we used the fact that reward follows the uniform distribution on interval $(0,1)$. To further lower-bound $\mathbb{E}[T_A]$, we establish a concentration inequality of $T - T_A$. By Hoeffding's inequality, we have, for any $t \le T_A$,

$$\mathbb{P}\Big[\sum_{\tau=1}^t z_\tau^* \le (0.5-\delta)t + \sqrt{\frac{t}{2}\log(1/x)}\Big] \ge 1 - x$$

for all $x \in (0,1)$. By plugging $t(x) = T - \frac{1}{0.5}\sqrt{\frac{T}{2}\log(1/x)}$, we deduce that $\mathbb{P}[\sum_{t=1}^{t(x)} z^*_{t(x)} \le 0.5T] \ge 1-x$ for any $x \in (0,1)$. Equivalently, for any $x \in (0,1)$, we have

$$\mathbb{P}\Big[T_A \ge T - \frac{1}{0.5}\sqrt{\frac{T}{2}\log(1/x)}\Big] \ge 1-x. \tag{EC.138}$$

From this inequality, we now observe that

$$\mathbb{E}[(T-T_A)^2] = \int_0^{T^2} \mathbb{P}[(T-T_A)^2 \ge y]dy = \int_0^{T^2} e^{-\frac{8y}{T}}dy \le \Theta(T). \tag{EC.139}$$

Here, the third equality follows from inequality (EC.138). By Jensen's inequality, we conclude from inequality (EC.139) that $\mathbb{E}[(T-T_A)] \le \Theta(\sqrt{T})$, or equivalently, $\mathbb{E}[T_A] \ge T - \Theta(\sqrt{T})$.

Plugging this bound into line (EC.137) (with $\delta = \sqrt{\gamma/T}$), we obtain[53]

$$\mathbb{E}\Big[\sum_{t=1}^{T_A} w_t z^*_t\Big] \ge (T - \Theta(\sqrt{T}))\Big(\frac{3}{8} - \frac{1}{2}\delta - \frac{1}{2}\delta^2\Big) = \frac{3}{8}T - \Theta(\sqrt{\gamma T} + \gamma).$$

We now turn our attention to bounding the average backlog of the proposed feasible solution. Let $\{b^*_t\}_{t=1}^T$ be the induced backlog. From our drift lemma (Lemma 3), for all $t \le T_A$, we have:

$$\mathbb{E}\Big[\frac{(b^*_t)^2 - (b^*_{t-1})^2}{2}\Big|\mathcal{H}_{t-1}\Big] \le b^*_{t-1}(\mathbb{E}[z^*_t|\mathcal{H}_{t-1}] - 0.5) + O(1) \le -\delta b^*_{t-1} + O(1) \tag{EC.140}$$

where in the last inequality we used $\mathbb{E}[z^*_t|\mathcal{H}_{t-1}] = \mathbb{E}[z^*_t] = \mathbb{P}[w_t \ge 0.5+\delta] = 0.5 - \delta$ because the reward follows the uniform distribution on interval $(0,1)$. Summing up line (EC.140) for $t \le T_A$ and taking the outer expectation, we obtain that

$$\delta\mathbb{E}\Big[\sum_{t=1}^{T_A-1} b^*_t\Big] \le -\frac{1}{2}\mathbb{E}[(b^*_{T_A})^2] + O(T). \tag{EC.141}$$

Hence, we have

$$\begin{aligned}
\frac{\mathbb{E}[\sum_{t=1}^T b^*_t]}{T} &= \frac{\mathbb{E}[\sum_{t=1}^{T_A-1} b^*_t] + \mathbb{E}[\sum_{t=T_A}^T b^*_t]}{T} \\
&= \frac{\mathbb{E}[\sum_{t=1}^{T_A-1} b^*_t] + \mathbb{E}[(T-T_A+1)b^*_{T_A-1}]}{T} \\
&= \frac{\mathbb{E}[\sum_{t=1}^{T_A-1} b^*_t] + \sqrt{\mathbb{E}[(T-T_A+1)^2]}\sqrt{\mathbb{E}[(b^*_{T_A-1})^2]}}{T} \\
&\le O(\frac{1}{\delta}+1).
\end{aligned}$$

In the second line, we used the fact that $b_t \le b_{T_A}$ for all $t \ge T_A$ (the backlog cannot increase after $t \ge T_A$ by definition of the feasible solution z^*_t). In the third line, we used the Cauchy-Schwartz inequality. In the last line, we used (i) $\mathbb{E}[\sum_{t=1}^{T_A-1} b^*_t] \le O(T/\delta)$ from line (EC.141), (ii) $\mathbb{E}[(b^*_{T_A})^2] \le O(T)$ from line (EC.141), and (iii) $\mathbb{E}[(T-T_A)^2] \le O(T)$ from the high-probability bound (EC.138). Combining with $\delta = \sqrt{\gamma/T}$, we have $\frac{\gamma}{T}\mathbb{E}[\sum_{t=1}^T b^*_t] = O(\sqrt{\gamma T} + \gamma)$.

[53] Because $\gamma = o(T)$, for sufficiently large T, we have $\frac{3}{8} - \frac{1}{2}\delta - \frac{1}{2}\delta^2 \ge 0$.

Combining, we conclude that the objective value of the feasible solution $\{z_t\}_{t=1}^{T}$ is

$$\mathbb{E}\Big[\sum_{t=1}^{T_A} w_t z_t^*\Big] - \frac{\gamma}{T}\mathbb{E}[\sum_{t=1}^{T} b_t^*] \geq \frac{3}{8}T - \Theta(\sqrt{\gamma T} + \gamma). \tag{EC.142}$$

This completes the proof.

EC.10. Additional Details on Data

The U.S. refugee registry data were provided to us under a collaboration research agreement with Global Refuge. This agreement requires that we do not transfer or disclose the data. Researchers interested in the data can contact Global Refuge at 700 Light Street, Baltimore, Maryland 21230, info@globalrefuge.org.

Data Preprocessing and Employment Prediction.

We use data on working-age refugees (18 to 64 years old) resettled by our partner agency during 2014–2016. This includes $T = 4445$ cases in 2014, $T = 3819$ cases in 2015, and $T = 4980$ cases in 2016. These data contain details on the refugees' characteristics (such as age, gender, origin, etc.), their matched affiliates, and their employment outcome, which is whether a refugee was employed 90 days after arrival at their assigned affiliate. Refugees' employment 90 days after arrival is the primary outcome metric that is tracked by the U.S. government and that U.S. resettlement agencies are required to report. The data also include indicators for cases with U.S. ties, which must be matched to predetermined affiliates. Data from 2014 is reserved for only model training and parameter tuning (as described below), while data from 2015 and 2016 is used for testing our methods and producing our results. For both 2015 and 2016, roughly 70% of cases had U.S. ties. In 2015 refugees were resettled across $m = 45$ affiliates, while in 2016 refugees were resettled across $m = 49$ affiliates.

To generate reward vectors for each case $\mathbf{w}_t$, we follow the methodology of Bansak et al. (2018), using supervised machine learning to generate rewards for each case–affiliate pair based on demographic and other background characteristics. Specifically, we employ stochastic gradient boosted trees with binomial deviance loss, which we implement in R using the `gbm` package, to generate an individual model for each affiliate using the data on refugees matched to that affiliate. We employ 5-fold cross-validation to determine the optimal values for tuning parameters, specifically the number of boosting iterations (trees) and the interaction depth of the trees. Parameters were tuned independently for each affiliate-specific model, with cross-validation over tree depths of 3-5 and a sufficiently large number of iterations/trees to minimize loss. Preliminary assessments on the bag fraction, learning/shrinkage rate, and minimum number of observations per node demonstrated relatively little to no impact on model performance within conventional value ranges, and so these parameters were held fixed at 0.5, 0.1, and 5, respectively.

Using these trained models, we then generate employment predictions for every refugee–affiliate pair (i.e. for each individual family member of each refugee case t at affiliate i) in 2015 and 2016. Following

the approach of Bansak et al. (2018), we then calculate the case-level employment reward $w_{t,i}$ as the probability that at least one family member secures employment at affiliate i, assuming independence across family members' employment outcomes. These case–affiliate rewards $w_{t,i}$ in 2015 and 2016 serve as the final data we use in testing our matching algorithms and producing our case study results.

Summary Statistics.

Here, we present a variety of summary statistics that describe the distributions of the rewards $w_{t,i}$ both across cases and across affiliates. For clarity, let $\boldsymbol{W}_{year}$ denote the matrix containing the rewards $w_{t,i}$ for a particular year. Hence, the rows of $\boldsymbol{W}_{year}$ correspond to cases and the columns correspond to affiliates. The reward vector $\mathbf{w}_t$ denotes the cases/rows, and let $\mathbf{w}_{*i}$ denote the affiliates/columns. In this section, we present calculations for various summary statistics separately for 2015 and 2016. The summary statistics are defined below, with Table EC.3 subsequently presenting the results.

Case-wise Summary Statistics (i.e. functions of $\mathbf{w}_t$):

- **Case-wise Average of the Max**. This quantity first takes the maximum reward value in the reward vector of each case, and then takes the average over all cases: $\frac{1}{T}\sum_{t=1}^{T} max(\mathbf{w}_t)$. In other words, for each row of $\boldsymbol{W}_{year}$ the maximum is taken, and then the average is calculated over all of those maximum values.
- **Case-wise SD of the Max**. This quantity is similar to the previous quantity, though it calculates the standard deviation (rather than the average) over all of the case-wise maximum values.
- **Case-wise Average of the Min**. This quantity first takes the minimum reward value in the reward vector of each case, and then takes the average over all cases: $\frac{1}{T}\sum_{t=1}^{T} min(\mathbf{w}_t)$. In other words, for each row of $\boldsymbol{W}_{year}$ the minimum is taken, and then the average is calculated over all of those minimum values.
- **Case-wise SD of the Min**. This quantity is similar to the previous quantity, though it calculates the standard deviation (rather than the average) over all of the case-wise minimum values.
- **Case-wise Average of the SD**. This quantity first takes the standard deviation of the reward value in the reward vector of each case, and then takes the average over all cases: $\frac{1}{T}\sum_{t=1}^{T} sd(\mathbf{w}_t)$. In other words, for each row of $\boldsymbol{W}_{year}$ the standard deviation is calculated, and then the average is calculated over all of those standard deviation values.
- **Case-wise SD of the SD**. This quantity is similar to the previous quantity, though it calculates the standard deviation (rather than the average) over all of the case-wise standard deviation values.
- **Case-wise Average of the Range**. This quantity first takes the range of the reward values in the reward vector of each case, and then takes the average over all cases: $\frac{1}{T}\sum_{t=1}^{T} [max(\mathbf{w}_t) - min(\mathbf{w}_t)]$.

In other words, for each row of $\boldsymbol{W}_{year}$ the range is calculated, and then the average is calculated over all of those range values.

- **Case-wise SD of the Range**. This quantity is similar to the previous quantity, though it calculates the standard deviation (rather than the average) over all of the case-wise range values.
- **Case-wise Average of the Pairwise Correlations**. This quantity first calculates the correlation between the reward vectors for all $\binom{T}{2}$ pairs of cases, and then takes the average over all cases: $\frac{1}{\binom{T}{2}} \sum_{t=1}^{T-1} \left[\sum_{t'=t+1}^{T} \rho(\mathbf{w}_t, \mathbf{w}_{t'}) \right]$ where $\rho(\mathbf{w}_t, \mathbf{w}_{t'})$ is the Pearson correlation between $\mathbf{w}_t$ and $\mathbf{w}_{t'}$. In other words, for each pair of rows in $\boldsymbol{W}_{year}$ the correlation is calculated, and then the average is calculated over all of those correlation values.
- **Case-wise SD of the Pairwise Correlations**. This quantity is similar to the previous quantity, though it calculates the standard deviation (rather than the average) over all of the correlation values between pairs of cases.

Affiliate-wise Summary Statistics (i.e. functions of $\mathbf{w}_{*i}$):

- **Affiliate-wise Average of the Max**. This quantity first takes the maximum reward value in the reward vector of each affiliate, and then takes the average over all affiliates: $\frac{1}{m} \sum_{i=1}^{m} max(\mathbf{w}_{*i})$. In other words, for each column of $\boldsymbol{W}_{year}$ the maximum is taken, and then the average is calculated over all of those maximum values.
- **Affiliate-wise SD of the Max**. This quantity is similar to the previous quantity, though it calculates the standard deviation (rather than the average) over all of the affiliate-wise maximum values.
- **Affiliate-wise Average of the Min**. This quantity first takes the minimum reward value in the reward vector of each affiliate, and then takes the average over all affiliates: $\frac{1}{m} \sum_{i=1}^{m} min(\mathbf{w}_{*i})$. In other words, for each column of $\boldsymbol{W}_{year}$ the minimum is taken, and then the average is calculated over all of those minimum values.
- **Affiliate-wise SD of the Min**. This quantity is similar to the previous quantity, though it calculates the standard deviation (rather than the average) over all of the affiliate-wise minimum values.
- **Affiliate-wise Average of the SD**. This quantity first takes the standard deviation of the reward value in the reward vector of each affiliate, and then takes the average over all affiliates: $\frac{1}{m} \sum_{i=1}^{m} sd(\mathbf{w}_{*i})$. In other words, for each column of $\boldsymbol{W}_{year}$ the standard deviation is calculated, and then the average is calculated over all of those standard deviation values.
- **Affiliate-wise SD of the SD**. This quantity is similar to the previous quantity, though it calculates the standard deviation (rather than the average) over all of the affiliate-wise standard deviation values.

- **Affiliate-wise Average of the Range**. This quantity first takes the range of the reward values in the reward vector of each affiliate, and then takes the average over all affiliates: $\frac{1}{m}\sum_{i=1}^{m}[max(\mathbf{w}_{*i}) - min(\mathbf{w}_{*i})]$. In other words, for each column of $\boldsymbol{W}_{year}$ the range is calculated, and then the average is calculated over all of those range values.
- **Affiliate-wise SD of the Range**. This quantity is similar to the previous quantity, though it calculates the standard deviation (rather than the average) over all of the affiliate-wise range values.
- **Affiliate-wise Average of the Pairwise Correlations**. This quantity first calculates the correlation between the reward vectors for all $\binom{m}{2}$ pairs of affiliates, and then takes the average over all affiliates: $\frac{1}{\binom{m}{2}}\sum_{i=1}^{m-1}\left[\sum_{i'=i+1}^{m}\rho(\mathbf{w}_{*i}, \mathbf{w}_{*i'})\right]$ where $\rho(\mathbf{w}_{*i}, \mathbf{w}_{*i'})$ is the Pearson correlation between $\mathbf{w}_{*i}$ and $\mathbf{w}_{*i'}$. In other words, for each pair of columns in $\boldsymbol{W}_{year}$ the correlation is calculated, and then the average is calculated over all of those correlation values.
- **Affiliate-wise SD of the Pairwise Correlations**. This quantity is similar to the previous quantity, though it calculates the standard deviation (rather than the average) over all of the correlation values between pairs of affiliates.

EC.11. Supplementary Results for Case Study (Section 5)

EC.11.1. Further Discussion on Learning Within-year Arrival Patterns

In Section 1, we motivated the design of distribution-free algorithms through Figure 1, which shows the significant year-to-year variations of the refugee pool composition. In light of the annual quota, we reiterate that the decision-making horizon of refugee matching is one year. We take 2015 as a running example. At the beginning of 2015, a new process begins for dynamic matching decisions. One approach (taken in prior work see Sections 1.1 and 1.3) is to assume that 2015 resembles earlier years, such as 2014, and use them as distributional knowledge. However, Figure 1 highlights the significant differences across the years, motivating design of algorithms without distributional knowledge, i.e., past years data. To this end, our learning-based algorithms are designed to learn the arrival pattern *within the year* through the dual variables $(\boldsymbol{\theta}^*, \boldsymbol{\lambda}^*)$ in Proposition 3.1. Note that $(\boldsymbol{\theta}^*, \boldsymbol{\lambda}^*)$ are defined for the one-year problem. For example, for year 2015, we have $(\boldsymbol{\theta}^*_{2015}, \boldsymbol{\lambda}^*_{2015})$. Since we do not rely on the data of 2014 to learn $(\boldsymbol{\theta}^*_{2015}, \boldsymbol{\lambda}^*_{2015})$, the difference between 2014's and 2015's pool composition is irrelevant to learning $(\boldsymbol{\theta}^*_{2015}, \boldsymbol{\lambda}^*_{2015})$.

That said, learning the dual variables for the one-year problem requires stationarity in arrivals within that year. For example, if the arrival patterns in the first half of 2015 significantly differ from those in the second half, we cannot learn $(\theta^*_{2015}, \lambda^*_{2015})$ effectively. In this section, we provide numerical evidence against such drastic changes. In particular, we show that the variation in arrival patterns *within the year* is smaller compared to the variation *across years*.

In the following, similar to Figure 1, we focus on the normalized number of tied cases. Specifically, we fix a year $y \in \{2014, 2015, 2016\}$ and use $N^y_{t,i}$ to denote the number of tied cases up to period t (i.e.,

Table EC.3 Reward Data Summary Statistics

Summary Statistic	Year	Case-wise	Affiliate-wise
Average of the Max	2015	0.881	0.948
SD of the Max	2015	0.141	0.076
Average of the Min	2015	0.030	0.017
SD of the Min	2015	0.023	0.020
Average of the SD	2015	0.225	0.213
SD of the SD	2015	0.049	0.051
Average of the Range	2015	0.852	0.931
SD of the Range	2015	0.134	0.079
Average of the Pairwise Correlations	2015	0.508	0.435
SD of the Pairwise Correlations	2015	0.153	0.164
Average of the Max	2016	0.909	0.957
SD of the Max	2016	0.118	0.077
Average of the Min	2016	0.031	0.020
SD of the Min	2016	0.024	0.019
Average of the SD	2016	0.238	0.214
SD of the SD	2016	0.045	0.051
Average of the Range	2016	0.878	0.938
SD of the Range	2016	0.111	0.079
Average of the Pairwise Correlations	2016	0.558	0.450
SD of the Pairwise Correlations	2016	0.141	0.160

the t-th arrival in the actual arrival sequence) at affiliate i. Letting T^y denote the total number of arrivals in year y, the normalized number of tied cases in year y at affiliate i is then given by $N^y_{T^y,i}/T^y$. These normalized values were presented in Figure 1. For each of comparison, we present them again in Figure EC.2a. On the same figure, we also present the coefficient of variation of $N^y_{T,i}/T^y$ across years $y \in \{2014, 2015, 2016\}$ (referred to as the *across-years CV* hereafter), as a measure of across-year variation of the number of tied cases.

To further investigate the within-year variations, we repeat the same process but this time within a year. Specifically, we divide a year into 4 quarters and compute the "quarterly" cumulative normalized

number of tied cases. Formally, we define $m^y_{q,i}$ ($q \in [4]$) as

$$m^y_{q,i} := \frac{N^y_{\lceil qT^y/4 \rceil, i}}{\lceil qT^y/4 \rceil}, \quad q \in [4] \tag{EC.143}$$

and report $\{m^y_{q,i}\}_{q\in[4]}$ for each affiliate i. If the arrival within year y is stationary, the variation of $\{m^y_{q,i}\}_{q\in[4]}$ (across q) should be small. Motivated by this observation, we calculate and present the coefficient of variation for $\{m^y_{q,i}\}_{q\in[4]}$ for each year y, referring to this as the *within-year CV* (of year y). Figure EC.2b presents $\{m^y_{q,i}\}_{q\in[4]}$ for each affiliate i and year y considered in Figure EC.2a along with the corresponding within-year CV.

By comparing Figures EC.2a and EC.2b, we observe that within-year variation is generally smaller than across-years variation. For example, when visually inspecting affiliate 44, we notice that the normalized number of tied cases varies significantly across the years 2014-2016, yet the tied cases arrived in a relatively stationary manner within each year. This observation is further supported by the significantly larger across-years CV compared to the within-year CV. We also note that some affiliates do not exhibit ideal stationarity in certain years (e.g., affiliate 34 in 2016). However, even for these cases, the within-year CV is generally smaller than the across-years CV. This suggests that our partner agency could benefit from algorithms that learn the arrival pattern *within the year*, rather than relying on past years' data. As explained in detail in Section 5, indeed our distribution-free algorithm (that only relies on the current year's data to make decisions) outperforms the algorithms that rely on past years' data.

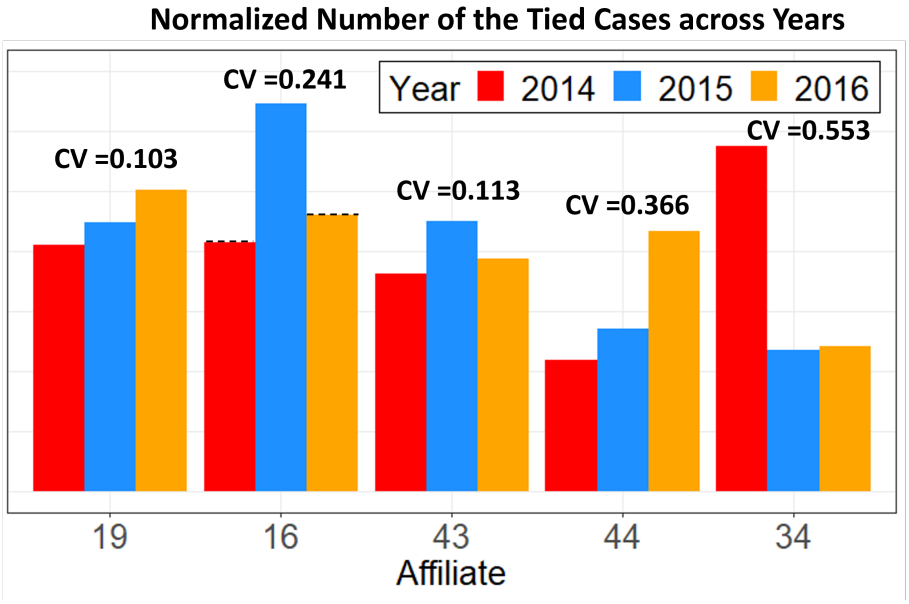
EC.11.2. Implementation Details of `RO-Learning` and `CO-DL`

In the following, we present more details on the implementation of `RO-Learning` for our case study. Specifically, Recall that Algorithm 1 (which `RO-Learning` is based on) requires the parameters η and ζ (i.e., the step-sizes for $(\boldsymbol{\theta}, \boldsymbol{\lambda})$ and $\boldsymbol{\beta}$, respectively). We set these step sizes as a function of the penalty parameters as follows:

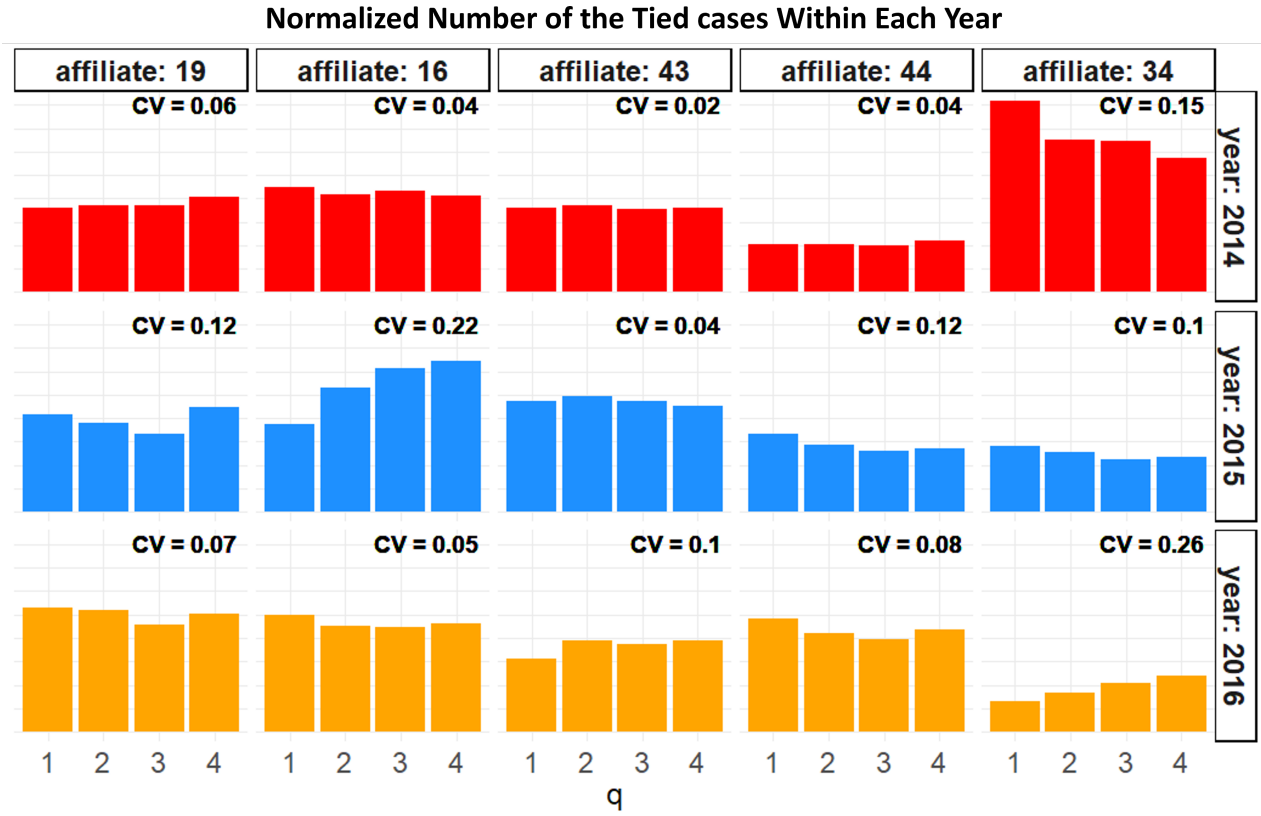
$$\eta = \frac{s_\eta \log(\alpha+1)}{\sqrt{T}}$$
$$\zeta = \frac{s_\zeta \gamma}{\sqrt{T}}$$

Intuitively, this makes the algorithm more conservative in terms of the cost metrics as the penalty parameters α and γ increase.

We adopt a data-driven approach of tuning the parameters (s_η, s_ζ), to select values that lead to acceptable outcomes for both over-allocation and average backlog. Specifically, we run the algorithm on the dataset of year 2014, obtaining the over-allocation and average backlog for each $1 \le s_\eta \le 5$ and $0.1 \le s_\zeta \le 1$. The resulting *acceptable region* for (s_η, s_ζ) is defined by values that satisfy two criteria: (i) achieving an over-allocation at most $1.2\times$ (the minimum value across all parameter settings) and (ii) maintaining an average backlog at most $1.1\times$ (the value achieved by the actual placement in



(a) Across-years variation of the normalized number of tied cases



(b) Within-year variation of the normalized number of tied cases

Figure EC.2 Across-years versus within-year variation of the normalized number of tied cases.

the year 2014). Figure EC.3 illustrates the acceptable region for some penalty parameters, with the ratio of employment outcome compared to the optimal level of surrogate primal (Definition 5). Upon visual examination, we identify that values within the ranges $4 \leq s_\eta \leq 5$ and $0.1 \leq s_\zeta \leq 1$ generally fall

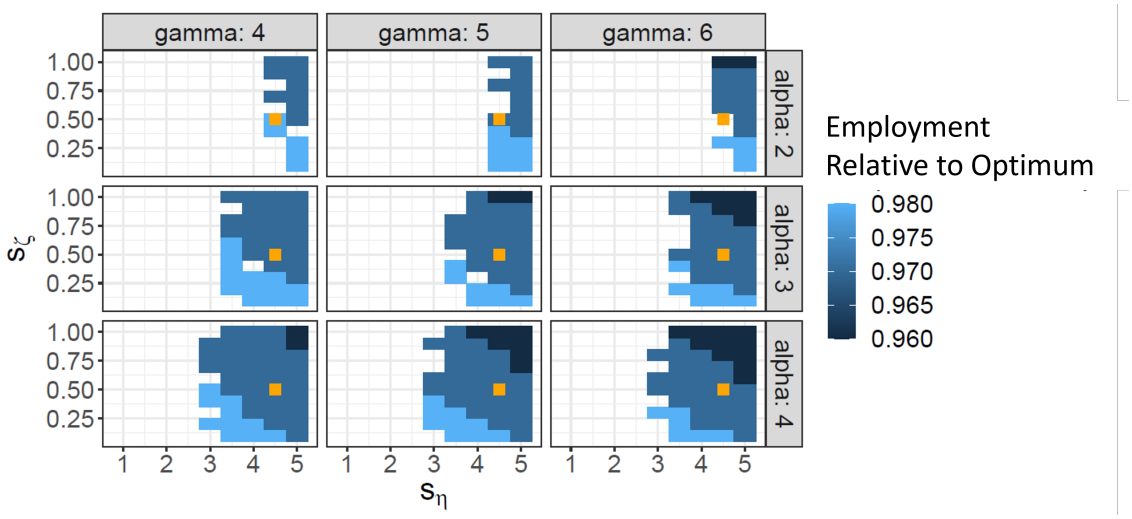


Figure EC.3 Acceptable region of (s_η, s_ζ) based on data of year 2014

within the acceptable region across a broad range of the penalty parameters. Consequently, we opt for $s_\eta = 4.5$ and $s_\zeta = 0.5$, roughly representing a central point within the established acceptable region. This selection aims to manage cost metrics without compromising employment outcomes too severely. The choice is further motivated to ensure the robustness of our case study results with respect to variations in step sizes.

We applied the same data-driven procedure to tune the step size η_t for implementing `CO-DL`. Specifically, we set the time-varying step size as $\eta_t = \frac{k \log(\alpha+1)}{\sqrt{t}}$, and selected $k \in [1, 10]$ based on the 2014 data, using the same acceptability criteria described above. The resulting choice was $k = 4$.

EC.11.3. Implementation Details of `Sampling`

In the following section, we present the details of `Sampling` for the sake of completeness. The original version of `Sampling` (Bansak and Paulson 2024) was designed for a setting without tied cases. Here, we formally describe the extension of the original version to accommodate scenarios with tied cases.

As we discussed in Section 5, `Sampling` *requires* a sample trajectory of future arrivals. The future arrivals are drawn from a *sampling pool*, denoted by $\mathcal{P}$, with replacement. Following Bansak and Paulson (2024), we use the previous year's arrival data as the sampling pool. At each time t, let $c_{t,i}$ denote the remaining capacity for each affiliate i. Upon arrival of case t and observing $(\mathbf{w}_t, i_t^\dagger)$, `Sampling` first samples a future arrival $(\mathbf{w}_\tau, i_\tau^\dagger)_{\tau=t+1}^T$ from its sampling pool $\mathcal{P}$. In our simulation, this means that we sample future arrivals of time $t+1$ and onward from the previous year's arrival data. Given this sampled trajectory of future arrivals, `Sampling` solves the following optimization problem

to obtain the matching decision: [54]

$$\max_{\{\mathbf{z}_\tau\}_{\tau=t}^{T}} \quad \tilde{\mathbf{w}}_t \cdot \mathbf{z}_t + \sum_{\tau=t+1}^{T} \mathbf{w}_t \cdot \mathbf{z}_t - \alpha \sum_{i=1}^{m} \left(\sum_{\tau=t}^{T} z_{t,i} - c_{t,i} \right)_+$$
$$\text{s.t} \quad \sum_{\tau=t}^{T} \mathbb{1}[i_t^\dagger = 0] z_{t,i} \leq \left(c_{t,i} - \sum_{\tau=t}^{T} \mathbb{1}[i_\tau^\dagger = i] \right)_+$$

where

$$\tilde{w}_{t,i} = w_{t,i} - \frac{\gamma}{T} \left\lceil \frac{b_{t-1,i} - \rho_i}{\rho_i} \right\rceil \mathbb{1}[b_{t-1,i} > 0]. \tag{EC.144}$$

That is, `Sampling` solves an offline problem that maximizes the net matching reward given the current remaining capacity with a tweak: to further control the congestion, as in equation (EC.144), it penalizes the (observed) employment probability at affiliate i for the *current* case t by the waiting time the case will experience upon placement computed based on the first-come-first-served manner (see Lemma 1 of Bansak and Paulson (2024) for discussion on such penalty function), with the penalty increasing with γ (the penalty parameter for congestion). The `Sampling` algorithm repeats this process for K simulated trajectories of future arrivals and assigns case t to the location where it was matched most often. In our case study, we set $K = 5$ for these simulations.

EC.11.4. Comparison of `CO-DL` and `RO-Learning` Across Penalty Parameters

To further investigate the impact of penalty parameters on the performance of `CO-DL`, we report the relative difference in objective value (as defined in Equation (Objective)) between `RO-Learning` and `CO-DL` across a broad range of penalty parameters. The heatmaps in Figure EC.4 report the relative improvement of `RO-Learning` in terms of the objective compared to `CO-DL`. We observe that `RO-Learning` consistently achieves a higher objective value than `CO-DL` for all parameter values considered. This improvement naturally stems from `RO-Learning` using backlog information and thus being able to better mitigate congestion. However, the performance gap narrows when the congestion penalty parameter γ is small; for instance, in year 2016, the relative improvement is less than 5% across all values of α when $\gamma = 1$.

EC.11.5. Tradeoff Curves Across a Range of Penalty Parameters

Figure EC.5 presents additional numerical results for years 2015 and 2016, covering all combinations of penalty parameters $\alpha \in \{1, 2, \ldots, 5\}$ and $\gamma \in \{0, \ldots, 10\}$. Each figure includes two panels per value of α: employment rate versus over-allocation (left) and employment rate versus average backlog (right). Results are shown for `RO-Learning`, `Sampling`, and `Actual`, with `Opt Emp` included as an additional benchmark. Overall, we observe similar qualitative patterns to those shown in Figure 3, including the dominance of `RO-Learning` over `Sampling` across a wide range of penalty parameters.

[54] We also note that, when $\alpha \geq 1$ (i.e., when the over-allocation penalty parameter is greater than the maximum employment outcome of each case), `Sampling` does not over-allocate free cases with respect to the current capacity due to the nature of the offline program it solves. Hence, as observed in Figure EC.5 in Section 5, the outcome of `Sampling` does not change with respect to the values of α.

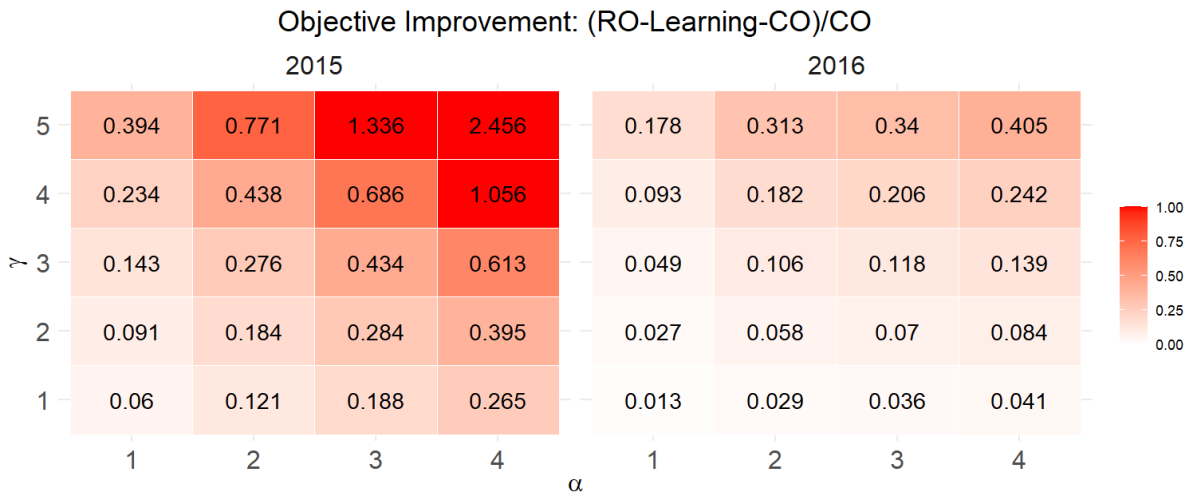


Figure EC.4 Relative objective improvement of `RO-Learning` over `CO-DL` across a grid of penalty parameters (α,γ). Each cell reports the relative improvement in the objective (Equation (Objective)) for the corresponding pair of penalty parameters. Results are shown separately for years 2015 and 2016.

EC.12. Generalized Model with Multiple Knapsack Constraints

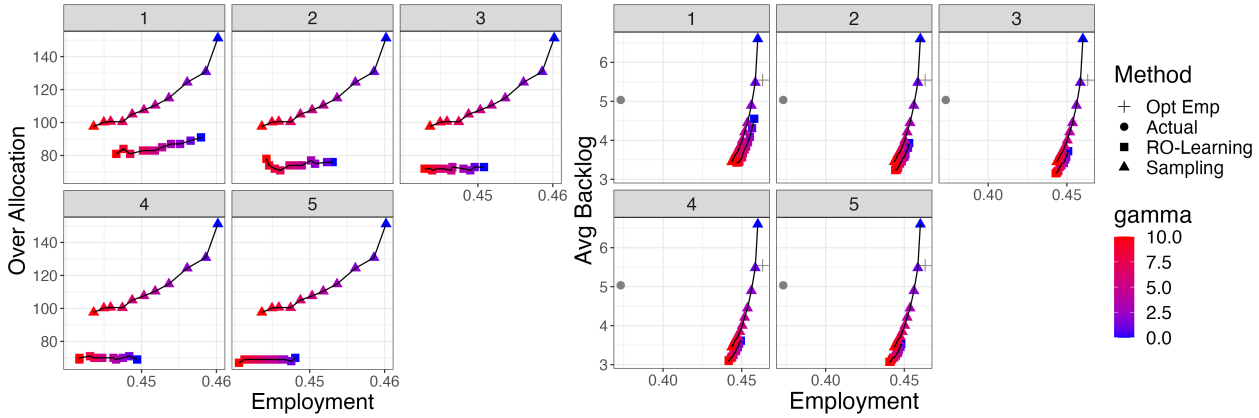
In our main model, for simplicity of exposition, we assumed that (i) each affiliate has one type of static resource (i.e., annual quota) and (ii) each case consumes one unit of the static resource at the matched affiliate. While these assumptions are largely aligned with the practice of our partner agency, in other contexts, a case may consume more than one type of resource. For example, for a family with children, there may be a constraint in terms of the number of children that can be enrolled at the local school (a static resource). Additionally, a case may have more than one child, thus consuming multiple units. Furthermore, enrolling children may require different forms of assistance, implying that it also requires a separate post-allocation service. In this section, we describe how our model, algorithms, and their performance guarantee seamlessly generalize in these settings.

EC.12.1. Generalized Model

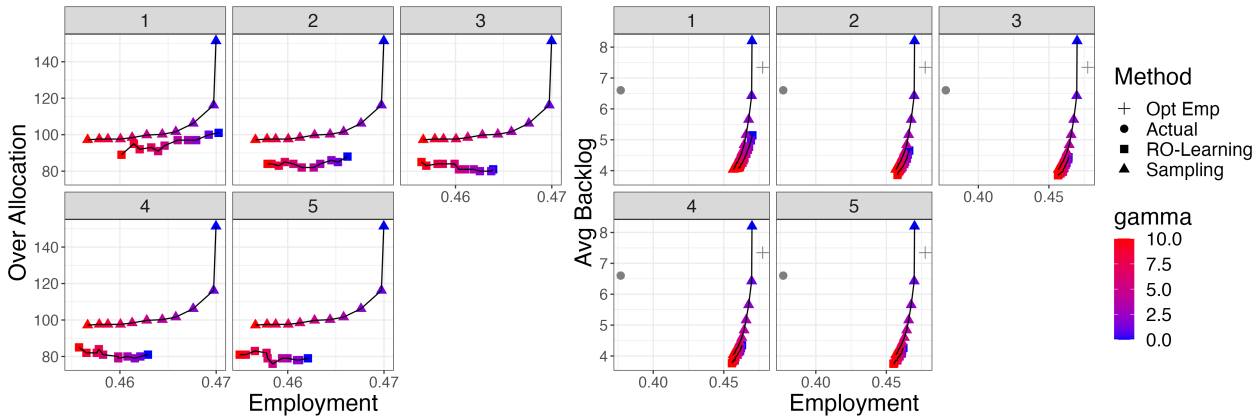
(i) Static Resource (Capacity): In this generalized model,[55] each affiliate $i \in [m]$ can have l types of *static* resources, referred to as the (static) resource-type (i,j) for $j \in [l]$. Each case t is characterized by its type $\mathbf{A}_t = (\mathbf{w}_t, i_t^\dagger, \boldsymbol{n}_t)$, where $\boldsymbol{n}_t = (n_{t,i,j})_{i\in[m],j\in[l]} \in \mathbb{R}_+^{ml}$ is the number of units consumed by case t for resource-type (i,j) when matched to affiliate i. We assume that $(n_{t,i,j})_{i\in[m],j\in[l]}$ is bounded by a constant denoted $\bar{n}$. Upon arrival of case t, the agency chooses a matching decision $\mathbf{z}_t \in \mathcal{Z}(i_t^\dagger)$ where $\mathcal{Z}(i^\dagger) = \Delta_m$ if $i^\dagger = 0$ and $\mathcal{Z}(i^\dagger) = \{\mathbf{e}_{i^\dagger}\}$ otherwise.[56] Similar to our base model, we assume that $\mathbf{A}_t$ is generated from an unknown i.i.d. distribution $\mathcal{F}$.

[55] We note that a similar model with multiple knapsack constraints was studied in Ahani et al. (2021) and Delacrétaz et al. (2023). However, these papers studies the static (rather than dynamic) matching problem and does not model the dynamic resource.

[56] While the type-feasibility set allows for fractional allocation, our algorithms will always make an integral decision. For example, when a refugee case is a family of multiple members, the case will be matched to the same affiliate.



(a) Year 2015



(b) Year 2016

Figure EC.5 **Numerical results with $\alpha \in \{1, 2, .., 5\}$ and $\gamma \in \{0, 1, ..., 10\}$. In each panel, we display the trajectory of the outcome metrics of fixed α (label of the penal) and varying γ. The left (the right, resp.) set of panels shows the over-allocation (the average backlog, resp.) versus the employment rate. `Opt Emp` on the right panels is the outcome of the surrogate primal problem (Definition 5).**

Each affiliate i is endowed with a fixed capacity $c_{i,j}$ for the static resource-type (i,j). We use $\rho_{i,j} = c_{i,j}/T$ to denote the capacity ratio of the static resource-type (i,j). Similar to our main model in Section 2, we use $\underline{\rho} := \min_{i\in[m],j\in[l]} \rho_{i,j}$ to denote the minimum capacity ratio and impose mild assumptions on these ratios: (i) $\sum_{i=1}^{m} \rho_{i,j} \le 1$ for all $j \in [l]$ and (ii) $\rho_{i,j} - \mathbb{E}[n_{t,i,j}\mathbb{1}[i_t^\dagger = i]] = \Theta(1)$ for all $i \in [m]$ and $j \in [l]$, which means that, in expectation, we have $\Theta(T)$ capacity for free cases for each static resource-type (i,j). Similar to our main base model, we impose the hard constraint that, for each static resource-type (i,j), no over-allocation can be made from free cases. Formally,

$$\sum_{\tau=1}^{t} \mathbb{1}[i_\tau^\dagger = 0] n_{\tau,i,j} z_{\tau,i} \le \left(c_{i,j} - \sum_{\tau=1}^{t} \mathbb{1}[i_\tau^\dagger = i] n_{\tau,i,j} \right)_+, \quad \forall i \in [m], j \in [l], t \in [T].$$

(ii) Dynamic Resource (Server): To incorporate the post-allocation service, we endow each static resource-type (i,j) with a dedicated server, referred to as the dynamic resource-type (i,j). Each server's availability $s_{t,i,j}$ for period t follows i.i.d Bernoulli process with service rate $r_{i,j}$. Similar to

our base model, we impose a stability condition that $r_{i,j} = \rho_{i,j} + \epsilon$ where $\epsilon \geq 0$ is a service slack. After case t is matched to affiliate i, it will add $n_{t,i,j}$ unit of workload to the corresponding server. Hence, the backlog for the dynamic resource-type (i,j) at time t will be given by

$$b_{t,i,j} = (b_{t-1,i,j} + n_{t,i,j} z_{t,i} - s_{t,i,j})_+. \tag{EC.145}$$

(iii) Benchmark and Regret: Analogous to (Objective), for given penalty parameters α and γ, the objective of online algorithm π is given by

$$\mathsf{ALG}^\pi(\alpha,\gamma) := \sum_{t=1}^{T}\sum_{i=1}^{m} w_{t,i} z^\pi_{t,i} - \alpha \sum_{i=1}^{m}\sum_{j=1}^{l}\left(\sum_{t=1}^{T} n_{t,i,j} z^\pi_{t,i} - c_{i,j}\right)_+ - \frac{\gamma}{T}\sum_{t=1}^{T}\sum_{i=1}^{m}\sum_{j=1}^{l} b^\pi_{t,i,j} \tag{EC.146}$$

where $\{\mathbf{z}^\pi_t\}_{t=1}^T$ is the matching profile of online algorithm π and $\{b^\pi_{t,i,j}\}_{t=1}^T$ is the corresponding backlog induced by the backlog dynamics (EC.145).

The optimal offline benchmark (generalization of Definition 2) is given by

$$\begin{aligned}
\mathsf{OPT}(\alpha,\gamma) := \max_{\substack{\mathbf{z}_t \in \mathcal{Z}(i^\dagger_t) \\ \mathbf{b}_t \geq \mathbf{0}}} & \sum_{t=1}^{T}\sum_{i=1}^{m} w_{t,i} z_{t,i} - \alpha \sum_{i=1}^{m}\sum_{j=1}^{l}\left(\sum_{t=1}^{T} n_{t,i,j} z_{t,i} - c_{i,j}\right)_+ - \frac{\gamma}{T}\sum_{t=1}^{T}\sum_{i=1}^{m}\sum_{j=1}^{l} b_{t,i,j} \\
\text{s.t. } & \sum_{t=1}^{T} \mathbb{1}[i^\dagger_t = 0] n_{t,i,j} z_{t,i} \leq \left(c_{i,j} - \sum_{t=1}^{T} \mathbb{1}[i^\dagger_t = i] n_{t,i,j}\right)_+, && \forall i \in [m], j \in [l] \\
& b_{t,i,j} \geq b_{t-1,i,j} + n_{t,i,j} z_{t,i} - s_{t,i,j}, && \forall t \in [T], i \in [m], j \in [l]
\end{aligned}$$

where we define $b_{0,i,j} = 0$ for all $i \in [m]$ and $j \in [l]$ by convention.

Given this benchmark, we define the same notion of the regret as Definition 3. That is, we evaluate an algorithm by its worst-case performance over all instances, where each instance $\mathcal{I}$ consists of (i) the set of recourse types $(i,j) \in [m] \times [l]$, (ii) capacity ratios $(\rho_{i,j})_{(i,j)\in[m]\times[l]}$, and (iii) the (unknown) arrival type distribution $\mathcal{F}$. The regret of an online algorithm π is then given by

$$\mathsf{Regret}^\pi_T := \sup_{\mathcal{I}} \mathbb{E}\left[\mathsf{OPT}(\alpha,\gamma) - \mathsf{ALG}^\pi(\alpha,\gamma)\right]. \tag{EC.147}$$

where the expectation is over the arrival distribution $\mathcal{F}$, the Bernoulli service process with a service rate vector $\mathbf{r}$, and (potential) randomness of the algorithm itself.

EC.12.2. Algorithms and Analysis

Having described the set up of the generalized model, in this section, we explain how we modify our algorithms and their analysis for this generalized model. To avoid repetition, we mainly focus on our first algorithm (Algorithm 1) and provide the details.

Congestion-Aware Dual Learning Algorithm (`CA-DL`$^{\mathtt{M}}$): To understand how to modify `CA-DL` (Algorithm 1) for the generalized model, we write the dual of the optimal offline benchmark. Define G_T be the event where sample path $(\mathbf{A}_t, \mathbf{s}_t)_{t=1}^T$ satisfies $\sum_{t=1}^T \mathbb{1}[i^\dagger_t = i] n_{t,i,j} \leq c_{i,j}$ for all $(i,j) \in [m] \times [l]$. Because we assumed that $\bar{n} = \Theta(1)$ and $\min_{i\in[m],j\in[l]} \rho_{i,j} - \mathbb{P}[n_{t,i,j}\mathbb{1}[i^\dagger_t = i]] = \Theta(1)$, Azuma-Hoeffding

inequality (and the union bound) implies that G_T occurs with probability $1-\mathcal{O}(e^{-T})$ (we again suppress the dependence on ml because they are constant by our assumption). Hence, in the same vein of Proposition 1, we can characterize the dual program of the optimal offline benchmark as follows:

PROPOSITION EC.63. *Define a per-period dual function D_t for case t as:*

$$D_t(\boldsymbol{\theta},\boldsymbol{\lambda},\boldsymbol{\beta}) := \max_{\mathbf{z}_t \in \mathcal{Z}(i_t^\dagger)} \sum_{i=1}^{m} \left\{ \left(w_{t,i} - \sum_{j=1}^{l} n_{t,i,j}(\theta_{i,j}+\lambda_{i,j}+\beta_{t,i,j}) \right) z_{t,i} + \sum_{j=1}^{l} (\rho_{i,j}\theta_{i,j} + \rho_{i,j}\lambda_{i,j} + s_{t,i,j}\beta_{t,i,j}) \right\}$$

Further, for any given sample path $(\mathbf{A}_t,\mathbf{s}_t)_{t=1}^T$, Consider the following dual program:

$$\mathsf{Dual}(\alpha,\gamma) := \min_{\boldsymbol{\theta},\boldsymbol{\lambda},\boldsymbol{\beta}_t \geq \mathbf{0}} \sum_{t=1}^{T} D_t(\boldsymbol{\theta},\boldsymbol{\lambda},\boldsymbol{\beta}_t) \tag{EC.148}$$

$$s.t.\ \theta_{i,j} \leq \alpha, \qquad \forall i \in [m], j \in [l] \tag{EC.149}$$

$$\beta_{t,i,j} - \beta_{t+1,i,j} \leq \frac{\gamma}{T} \qquad \forall t \in [T-1], i \in [m], j \in [l] \tag{EC.150}$$

$$\beta_{T,i,j} \leq \frac{\gamma}{T} \qquad \forall i \in [m], j \in [l] \tag{EC.151}$$

Then, we have the following strong duality: $\mathsf{OPT}(\alpha,\gamma)\mathbb{1}[G_T] = \mathsf{Dual}(\alpha,\gamma)\mathbb{1}[G_T]$ where G_T is the event that sample path $(\mathbf{A}_t,\mathbf{s}_t)_{t=1}^T$ satisfies $\sum_{t=1}^T \mathbb{1}[i_t^\dagger = i] n_{t,i,j} \leq c_{i,j}, \forall i \in [m], j \in [l]$.

The proof of Proposition EC.63 follows the same steps as Proposition 1 and is therefore omitted. As in Proposition 1, the matrix $\boldsymbol{\theta} \in \mathbb{R}^{m \times l}$ in Proposition EC.63 represents the dual variables for over-allocation costs, where $\theta_{i,j}$ corresponds to the resource-type (i,j). The interpretation of other dual variables can be similarly understood in parallel with Proposition 1. From Proposition EC.63, we again observe that the optimal offline benchmark matches case t to the affiliate that maximizes the adjusted score. However, the main difference from Proposition 1 is that adjustment for each affiliate i is $\sum_{j=1}^{l} n_{t,i,j}(\theta_{i,j}+\lambda_{i,j}+\beta_{t,i,j})$. In words, when matched with affiliate i, the case consumes multiple units of multiple resource types, and therefore the opportunity cost of matching case t to that affiliate should reflect that.

Given Proposition EC.63, we can now follow the same steps outlined in Section 3 to adapt Algorithm 1 for the generalized model. Specifically, we consider the surrogate dual problem (analogous to Definition 4) and then apply the same online learning update rules to each dual variable in the surrogate dual problem. We call this modified algorithm `CA-DL`$^{\mathtt{M}}$. For brevity, we focus on describing the key differences between `CA-DL`$^{\mathtt{M}}$ and Algorithm 1 below:

Primal Phase: `CA-DL`$^{\mathtt{M}}$ matches case t to affiliate i with the maximum dual-adjusted score $w_{t,i} - \sum_{j=1}^{l} n_{t,i,j}(\theta_{t,i,j}+\lambda_{t,i,j}+\eta b_{t-1,i,j})$ (breaking ties arbitrarily), as long as (i) the adjusted score is non-negative and (ii) every affiliate i has remaining capacity for the static resource-type (i,j). Otherwise, the case is matched to the dummy affiliate. Here the scaled build-up $\zeta b_{t-1,i,j}$ again implicitly plays a role of the dual variable for the dynamic resource-type (i,j).

Dual Phase: The dual update rule (equation (7) in Algorithm 1) is replaced with

$$\begin{aligned}\theta_{t+1,i,j} &= \min\{\theta_{t,i,j}\exp(\eta(n_{t,i,j}z_{t,i}-\rho_{i,j})),\alpha\}\\ \lambda_{t+1,i,j} &= \min\left\{\lambda_{t,i,j}\exp(\eta(n_{t,i,j}z_{t,i}-\rho_{i,})),\frac{1+2\alpha}{\underline{\rho}}\right\}.\end{aligned} \tag{EC.152}$$

Analogous to equation (8), the dual variables for the dynamic resource are implicitly updated as $\beta_{t+1,i,j} = (\beta_{t,i,j} + \zeta(n_{t,i,j}z_{t,i} - s_{t,i,j}))_+$ with $\beta_{0,i,j} = 0$. Similar to our base model, we can show that $\beta_{t+1,i,j} = \zeta b_{t,i,j}$ using an induction argument.

With these modifications, the regret guarantees of `CA-DL` (Theorem 1 and Corollary 1) extend to `CA-DL`$^{\mathtt{M}}$. We summarize these analogous results in the following corollary.

COROLLARY EC.64 **(Regret of `CA-DL`$^{\mathtt{M}}$).** *Let $\eta = \Theta(1/\sqrt{T})$ and $\zeta = \Theta(1/\sqrt{T})$. Under the stable regime (Definition 1), the regret of `CA-DL`$^{\mathtt{M}}$ is $\mathcal{O}(\sqrt{T} + \frac{\gamma}{\epsilon})$. Furthermore, under the near-critical regime (Definition 1), the regret of `CA-DL`$^{\mathtt{M}}$ is $\mathcal{O}(\sqrt{\gamma T})$ by setting $\eta = \Theta(1/\sqrt{T})$ and $\zeta = \Theta(\sqrt{\gamma/T})$.*

We remark that even though the regret bounds given in Corollary EC.64 have the same order as the ones given in Theorem 1 and Corollary 1, they are not the same because the new parameters $\bar{n}$ (the maximum number of units consumed for each static resource type by a case) and l (the number of resource types) impact the bounds. However, under the natural assumption that these parameters are constant (i.e., do not increase with T), they do not change the order of the regret.

The proof of Corollary EC.64 follows nearly identical steps to those outlined in Section 3.3. The only change in the proof is in the definition of the pseudo-rewards due to the modification in the dual update rules. Specifically, for the backlog vector $\mathbf{b}_t = (b_{t,i,j})_{i\in[m],j\in[l]}$, we consider the quadratic potential function and its drift:

$$\psi(\mathbf{b}_t) := \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{l} b_{t,i,j}^2, \qquad D_t := \psi(\mathbf{b}_t) - \psi(\mathbf{b}_{t-1}). \tag{EC.153}$$

For the proof of Corollary EC.64, we replace the pseudo-rewards K_t in equation (14) with the following:

$$K_t := \mathbf{w}_t \cdot \mathbf{z}_t + \sum_{i=1}^{m}\sum_{j=1}^{l}\theta_{t,i,j}(\rho_{i,j} - n_{t,i,j}z_{t,i}) + \sum_{i=1}^{m}\sum_{j=1}^{l}\lambda_{t,i,j}(\rho_{i,j} - n_{t,i,j}z_{t,i}) - \zeta D_t. \tag{EC.154}$$

With the above definition of the pseudo-rewards, one can repeat the proofs of Lemma 4 and Lemma 5 to establish the lower and upper bound of the expected sum of the pseudo-rewards, leading to an analogous inequality to (20) (see Section 3.3). The details are omitted for brevity.

Congestion-Oblivious Dual Learning Algorithm: One can apply the similar ideas in Section 4.2 to modify `CO-DL` (Algorithm 2) for the generalized model. The modified algorithm, which we call `CO-DL`$^{\mathtt{M}}$, is again identical to the `CA-DL`$^{\mathtt{M}}$ except that (i) `CO-DL`$^{\mathtt{M}}$ only maintains the dual variables $\theta_{t,i,j}$ and $\lambda_{t,i,j}$ for each static resource-type (i,j) and (ii) the fixed step size η for the dual update rules

in equation (EC.152) is replaced with the time-varying step size $\eta_t = \Theta(1/\sqrt{t})$ for each period t. The regret guarantee of Theorem 2 extends to CO-DL$^{\text{M}}$. That is, the regret of CO-DL$^{\text{M}}$ is $\mathcal{O}(\sqrt{T} + \frac{\gamma}{\epsilon})$ under the stable regime. The proof again follows the identical steps as the proof of Theorem 2 and hence is omitted for brevity.

EC.13. Numerical Results with Mid-year Disruption and Capacity Revisions

Our main case study in Section 5 was based on data from 2014-2016. However, in the middle of fiscal year 2017, a change in federal administration led to significant shifts in refugee resettlement policy, including a sharp reduction and changes to the composition of arriving cases. These changes introduced substantial disruptions in the refugee arrival process. In this section, we examine the applicability of our proposed algorithms in the presence of such policy disruptions. To do so, we first provide brief background on the major changes to the U.S. refugee policy that took place in 2017. We then describe how we construct synthetic data that reflects those changes and simulate refugee arrivals under this disruption. Finally, we evaluate the performance of our algorithms on this synthetic data and show that they continue to yield meaningful improvements relative to the benchmarks.

Background. In fiscal year 2017, a change in federal administration brought about significant disruptions to the U.S. refugee admissions system. The presidential ceiling on refugee admissions was significantly reduced, and this reduction was accompanied by a sharp decline in the number of free cases. For example, Executive Order 13769 restricted the admission of refugees from certain countries, with exceptions primarily granted to those with existing U.S. ties (Howe 2017, Asian Americans Advancing Justice 2019). As a result, both the total number and composition of refugee arrivals changed substantially during 2017.

When such policy changes occur, resettlement agencies typically revise capacity for each affiliate based on the newly announced ceiling. While this revision process involves informal negotiation with the U.S. State Department, capacity adjustments are generally made in proportion to the original annual quota across affiliates.

Synthetic Data of Disruption. Motivated by the above background, we construct a synthetic dataset that mimics the policy disruption that occurred in year 2017. To do so, we apply a synthetic shock to the 2016 arrival sequence. Let T denote the total number of cases in the original dataset. We preserve the overall setup of our base case study (with unit-size case; Sections 5.1 and 5.2). We use the actual number of cases resettled at each affiliate as its initial capacity c_i, and set the normalized capacity $\rho_i = c_i/T$ for each affiliate i. To simulate the disruption, we select a disruption time t_d and randomly remove 50% of the free cases (those that were to arrive after t_d), while preserving the original arrival order among the remaining cases. This models both the drop in overall arrivals and the shift in composition similar to the 2017 policy changes. Let $\hat{T}$ denote the total number of arrivals after the

disruption (including those before t_d). We assume that the algorithm is informed of $\hat{T}$ at the time of disruption. This reflects institutional practice, where a revised ceiling is publicly announced and used to revise capacities. Affiliate capacities are then proportionally updated to $\hat{c}_i = \rho_i \hat{T}$, in line with our partner agencys typical approach to revising capacity proportionally to the initial ones.

Algorithm Evaluation and Numerical Results. We focus on comparing the performance of `Sampling` and `RO-Learning`.[57] Both algorithms are run exactly as in the base case study (Section 5.1), with one exception: upon the disruption at time t_d, we replace the original capacity c_i with the revised capacity $\hat{c}_i$. (For `RO-Learning`, the learning rates are left unchanged.) We set $t_d = T/3$ (resp. $t_d = 2T/3$) to model early (resp. late) disruption.

Table EC.4 presents the numerical results. Comparing `RO-Learning` with `Sampling`, we note that `RO-Learning` achieves employment outcomes comparable to `Sampling`, while further reducing both over-allocation and average backlog. The main source of this performance improvement lies in the robust learning process of `RO-Learning` (as highlighted in Section 5.2): after the disruption at t_d, although the dual variables initially reflect pre-disruption arrivals, the underlying adversarial learning algorithm exhibits a powerful self-correcting property that quickly adapts to changes in the arrival pattern.[58] This fast adaptation is what enables `RO-Learning` to maintain strong performance even under sudden structural shifts.

Table EC.4 Numerical performance under synthetic disruption (early vs. late reduction in free cases, 2016). We use the same penalty cost parameter $\alpha = 3$ and $\gamma = 5$ as in Table 1. Early (late, resp.) disruption randomly reduces the remaining number of free cases by half at $t_d = T/3$ (resp, $t_d = 2T/3$). Each cell shows the result under early disruption with the late disruption result in parentheses. The results are averaged over 10 simulation runs (each involving both the random disruption and the internal randomness of `Sampling`).

	Employment Rate (%)	Total Over- allocation	Average Backlog
`Sampling`	43.7 (45.5)	191.1 (149.4)	149.9 (179.3)
`RO-Learning`	43.7 (45.3)	190.2 (129.8)	140.4 (160.2)

EC.14. Incorporating Batched Arrivals

In our main model, we have assumed that refugee cases arrive one by one (which is true for Switzerland or the Netherlands). However, in some countries (such as the U.S.), the resettlement agency can encounter a periodic batch of refugee arrivals (e.g., weekly), for which matching decisions can be made

[57] Because our simulation is based on synthetic data set, we do not know an actual placement that would have been made by our partner.

[58] In fact, we implemented an alternate version of the algorithm that explicitly "restarts" at the time of disruption by resetting the dual variables to their default initial values (i.e., e^{-1}), and found that its performance rarely differs from the original version. From a theoretical standpoint, the initial dual variables affects regret guarantees only through a constant multiplicative factor and thus not impact the asymptotic order of regret (Hazan et al. 2016).

simultaneously. In this section, we discuss how our main algorithm can be modified to incorporate batched arrivals and explore the impact of such modifications.

Before going forward, we highlight that our main algorithms (in particular `RO-Learning` in Section 5) can still be implemented in practice for batched arrivals by simply ignoring batching and matching the cases within the batch one-by-one. We do not expect the loss from making decisions one by one to be substantial, partly because, in practice, batch sizes are relatively small. Specifically, even though our partner manages a weekly batch, the placement officer typically splits it into 20-30 cases. This is because finalizing a matching decision often requires communication with the corresponding affiliate, making it overwhelming for the officer to handle a larger number of cases at once.

That said, batched matching with even moderate batch sizes may offer an opportunity for performance improvement. To explore this opportunity, we propose a modification of `RO-Learning`—an adaptation of `CA-DL` for our partner agency (see Section 5)—to incorporate batching. Before proceeding, we remark that developing a primal-dual algorithm with adversarial online learning for batched arrivals is not straightforward and has not been studied in the literature (to the best of our knowledge).

EC.14.1. Model

We first begin by describing our model under batching. In describing the model and algorithm, we specifically restrict our attention to the same setup as in the numerical case study (Section 5). In particular, we consider a deterministic service flow with service rate $\boldsymbol{\rho}$ (which is currently used by our partner to measure congestion).

We consider a discrete-time model with T periods. There is a sequence of T cases, which arrive in K batches. For each batch $k \in [K]$, there are B_k cases. We index each arrival in batch k by $\mathcal{T}_k = \{t_k, t_k+1, \ldots, t_k+B_k-1\}$, where $t_1 = 1$ and $t_k = \sum_{j=1}^{k-1} B_j$ for $k \geq 2$. At the beginning of period t_k, batch k arrives, and we observe their types $\{\mathbf{A}_t\}_{t\in\mathcal{T}_k}$. The agency can make matching decisions $\{\mathbf{z}_t\}_{t\in\mathcal{T}_k}$ for cases within batch k simultaneously. However, for simplicity, we assume that cases arrive at their matched affiliate sequentially and in the order of their indices; that is, case $t \in \mathcal{T}_k$ arrives at period t. This assumption is aligned with practice as it is unlikely that several cases can simultaneously arrive at the affiliates. Hence, the backlog dynamic t remains the same and is given by

$$b_{t,i} = (b_{t-1,i} + z_{t,i} - \rho_i)_+ \tag{EC.155}$$

for each $t \in [T]$ and $i \in [m]$. The objective is again given by equation (Objective).

EC.14.2. Warm-up: `RO-Learning-B`

We begin with a natural yet naive modification of `RO-Learning`. Recall that `RO-Learning` finds the matching decision for case t by maximizing its adjusted score, subject to the capacity constraint. The adjusted score was defined by the reward subtracted by (i) the current estimate of dual variables for the static resource and (ii) the scaled level of the current backlog. Hence, a natural modification, which we

call `RO-Learning-B` (with B standing for batching), is as follows: find the matching decisions for batch k by maximizing the sum of the adjusted scores within the batch (subject to the capacity constraint), given the current estimate of dual variables. Formally, let $\mathbf{c}_k = (c_{k,i})_{i\in[m]}$ denote the current capacity at time t_k (i.e., upon the arrival of batch k). We further use $\boldsymbol{\theta}$ and $\boldsymbol{\lambda}$ to denote the current estimate of dual variables for the static resource (for brevity, we omit the time index for the dual variables). Then `RO-Learning-B` implements the following primal and dual phases:

Primal Phase: Find the matching decisions $\{\mathbf{z}_t\}_{t\in\mathcal{T}_k}$ for batch k by solving:[59]

$$\{\mathbf{z}_t\}_{t\in\mathcal{T}_k} = \underset{\substack{\mathbf{z}_t\in\mathcal{X}(i_t^\dagger),\ t\in\mathcal{T}_k\\ \forall t\in\mathcal{T}_k}}{\arg\max} \sum_{t\in\mathcal{T}_k} (\mathbf{w}_t - \boldsymbol{\theta} - \boldsymbol{\lambda} - \zeta\mathbb{1}[t=t_k]\mathbf{b}_{t-1})\cdot\mathbf{z}_{t,i} \tag{EC.156}$$

$$\text{s.t.} \sum_{t\in\mathcal{T}_k} \mathbb{1}[i_t^\dagger = 0] z_{t,i} \le \Big(c_{k,i} - \sum_{t\in\mathcal{T}_k} \mathbb{1}[i_t^\dagger = i]\Big)_+, \quad \forall i\in[m]. \tag{EC.157}$$

where where $\mathcal{X}(i^\dagger) = \Delta_m$ if $i^\dagger = 0$ and $\mathcal{X}(i^\dagger) = \{\mathbf{e}_{i^\dagger}\}$ otherwise (recall that `RO-Learning` matches every case to an actual affiliate—see Section 5).

Dual Phase: Based on the primal matching decision, we obtain the "batched" gradient, which is simply the summation of the gradient information $\mathbf{z}_t - \boldsymbol{\rho}$ over all arrival $t\in\mathcal{T}_k$. We then perform the multiplicative update (similar to Algorithm 1) using this batched gradient:

$$\begin{aligned}\theta_i &\leftarrow \min\Big\{\theta_i \exp\Big(\eta\sum_{t\in\mathcal{T}_k}(z_{t,i}-\rho_i)\Big), \alpha\Big\}\\ \lambda_i &\leftarrow \min\Big\{\lambda_i \exp\Big(\eta\sum_{t\in\mathcal{T}_k}(z_{t,i}-\rho_i)\Big), \frac{1+2\alpha}{\underline{\rho}}\Big\}\end{aligned} \tag{EC.158}$$

We evaluate `RO-Learning-B` in Table EC.5. For the numerical evaluation, we preserve the overall setup of our base case study (with unit-size case; Section 5), but we use $B_k = 30$ (similar to the current practice) and the same penalty parameters. The step sizes are tuned based on the 2014 data, following the procedure described in Section EC.11.2. Notably, the heuristic performs worse than our original algorithm (which ignores batching) in terms of the objective (equation (Objective)), primarily due to an increased average backlog.

In fact, despite its intuitive appeal, the above heuristic has two potential issues. First, the primal and dual variables are not updated sufficiently. To understand how this can lead to underperformance, consider the extreme case of having only one batch. In this case, the current backlog $\mathbf{b}_0$ is simply zero, and the initial dual variables $(\boldsymbol{\theta}, \boldsymbol{\lambda})$ are arbitrarily set. As a result, not only can the primal decisions from the primal phase be highly inefficient, but the updated dual variables from the dual phase are also never used to update these inefficient primal decisions. This suggests the need for multiple iterations of the primal-dual phase within each batch, as typically done in the batch gradient descent (Ruder 2016). Second, the objective function of (EC.156) does not account for the backlog dynamics within each batch. In summary, these two issues suggest that iterating the primal-dual phases multiple times per batch, as well as incorporating the backlog dynamics within each batch, may improve performance.

[59] It is straightforward to see that the original capacity constraint (i.e., (Capacity Feasibility-t) for all $t\in\mathcal{T}_k$) is satisfied if and only if the constraint (EC.157) is satisfied.

EC.14.2.1. Main Heuristic: `RO-Learning-B-Iterate`

Motivated by the above observations, we propose a more sophisticated modification of `RO-Learning`, called `RO-Learning-B-Iterate`. The formal description of the algorithm is presented in Algorithm 3. For brevity, we only describe the main changes from `RO-learning-B`. The most significant change from `RO-Learning-B` lies in how we obtain the primal decisions through the "inner" primal–dual iterations. Specifically, we account for the backlog induced within the current batch and iterate the inner primal–dual phases $L > 1$ times. These inner iterations within the batch aim to obtain higher-quality primal matching decisions. Each inner iteration consists of the following primal and dual phases.

Inner Primal Phase (line 6-8): The objective function for finding the "best" matching $\{\mathbf{z}_t^{(l)}\}_{t\in\mathcal{T}_k}$ (for the l-th iteration of batch k) is now the sum of the adjusted scores, penalized by the cost of over-allocation and average backlog within the batch (see line 7). In other words, we leverage the knowledge of the arrivals within the current batch (along with the deterministic service flow) to incorporate the average backlog induced by their matching decisions. Here, due to the backlog terms, the solution from the offline program in line 7 can be fractional. Therefore, in line 8, we take the maximum coordinate of the solutions across the affiliates to make them integral.

Inner Dual Phase (line 9): We perform the dual updates using the batched gradient. Here, we divide the original step size η by the number of iterations L to account for the multiple updates within each batch.

Finally, we set the actual matching decisions for batch k to be $\{z_t^{(L)}\}_{t\in\mathcal{T}_k}$, the final iterate of the primal phase after L iterations (line 11). The remaining steps are identical to `RO-Learning-B`—we use these final primal decisions for matching and then update the dual variables as described in equation (EC.161).

The numerical performance of `RO-Learning-B-Iterate` is presented in Table EC.5. We use $L = 10$ and the step sizes are tuned based on the 2014 data (following the approach in Section EC.11.2). We note that `RO-Learning-B-Iterate` performs slightly better than `RO-Learning` in terms of the objective value (by 4% for both years 2015 and 2016). This mild improvement is driven by a reduction in the average backlog (by 4-6%) and the over-allocation, with a small decrease in the employment rate (less than 2%). This numerical result suggests that leveraging batched arrivals in the setting of our case study may yield a small performance gain. However, we emphasize that the algorithm required to achieve such a gain is significantly more complex and less interpretable. This level of complexity appears necessary for performance improvement: as discussed in Section EC.14.2, our more straightforward yet naive modification of `RO-Learning` (`RO-Learning-B`) indeed hurts performance. Given this, it is unclear whether the additional complexities are justified by such a small gain.

Algorithm 3 `RO-Learning-B-Iterate`

Input: T, $\boldsymbol{\rho}$, η, ζ, L. // L is the number of iterations for inner primal-dual phase

Initialize $\theta_i = \exp(-1)$, $\lambda_i = \exp(-1)$, $c_{0,i} = \rho_i T$, $b_{0,i} = 0$ for all $i \in [m]$, and $t_1 = 1$.

// For brevity, we omit time index for the dual variables for static resources

for Each batch $k \in [K]$ **do**

Set the arrival index for batch k as $\mathcal{T}_k = \{t_k, t_k + 1, ..., t_k + B_k - 1\}$

for Each iteration $l \in [L]$ **do**

Inner Primal Phase: Set the dual-adjusted score as

$$\overline{\mathbf{w}}_t := \mathbf{w}_t - \boldsymbol{\theta} - \boldsymbol{\lambda} - \zeta \mathbf{b}_{t-1} \mathbb{1}[t = t_k], \quad \forall t \in \mathcal{T}_k. \tag{EC.159}$$

Solve the following program:

$$\begin{aligned}
\{\tilde{\mathbf{z}}_t^{(l)}, \tilde{\mathbf{b}}_t^{(l)}\}_{t\in\mathcal{T}_k} = \underset{\substack{\mathbf{z}_t^{(l)} \in \mathcal{X}(i_t^\dagger), \forall t \in \mathcal{T}_k \\ \mathbf{b}_t^{(l)} \geq \mathbf{0}, \forall t \in \mathcal{T}_k}}{\arg\max} & \sum_{t\in\mathcal{T}_k} \overline{\mathbf{w}}_t \cdot \mathbf{z}_{t,i}^{(l)} - \alpha \sum_{i\in[m]} \Big(\sum_{t\in\mathcal{T}_k} z_{t,i}^{(l)} - c_{k,i} \Big)_+ - \frac{\gamma}{B_k} \sum_{t\in\mathcal{T}_k} \sum_{i\in[m]} b_{t,i}^{(l)} \\
\text{s.t.} \quad & \sum_{t\in\mathcal{T}_k} \mathbb{1}[i_t^\dagger = 0] z_{t,i}^{(l)} \leq \Big(c_{k,i} - \sum_{t\in\mathcal{T}_k} \mathbb{1}[i_t^\dagger = i] \Big)_+, && \forall i \in [m] \\
& b_{t,i}^{(l)} \geq b_{t-1,i}^{(l)} + z_{t,i}^{(l)} - \rho_i, && \forall t \in \mathcal{T}_k, i \in [m]
\end{aligned}$$

where $\mathcal{X}(i^\dagger) = \Delta_m$ if $i^\dagger = 0$ and $\mathcal{X}(i^\dagger) = \{\mathbf{e}_{i^\dagger}\}$ otherwise. // Every case is matched to an actual affiliate consistent with the case study (Section 5)

Set the inner primal decisions $z_{t,i}^{(l)} = \mathbb{1}[i = \arg\max_{j\in[m]} \tilde{z}_{t,j}^{(l)}] \quad \forall t \in \mathcal{T}_k, i \in [m]$

Inner Dual Phase: update the dual variables as

$$\begin{aligned}
\theta_i &\leftarrow \min\Big\{ \theta_i \exp\Big(\frac{\eta}{L} \sum_{t\in\mathcal{T}_k} (z_{t,i}^{(l)} - \rho_i) \Big), \alpha \Big\} \\
\lambda_i &\leftarrow \min\Big\{ \lambda_i \exp\Big(\frac{\eta}{L} \sum_{t\in\mathcal{T}_k} (z_{t,i}^{(l)} - \rho_i) \Big), \frac{1+2\alpha}{\underline{\rho}} \Big\}
\end{aligned} \tag{EC.160}$$

end for

Set the primal decisions: $z_{t,i} = z_{t,i}^{(L)}, \quad \forall t \in \mathcal{T}_k, i \in [m]$ // Use the primal decision of the last iterate to make the actual matching decision.

Update the dual variables

$$\begin{aligned}
\theta_i &\leftarrow \min\Big\{ \theta_i \exp\Big(\eta \sum_{t\in\mathcal{T}_k} (z_{t,i} - \rho_i) \Big), \alpha \Big\} \\
\lambda_i &\leftarrow \min\Big\{ \lambda_i \exp\Big(\eta \sum_{t\in\mathcal{T}_k} (z_{t,i} - \rho_i) \Big), \frac{1+2\alpha}{\underline{\rho}} \Big\}
\end{aligned} \tag{EC.161}$$

Update the backlog: $b_{t,i} = (b_{t-1,i} + z_{t,i} - \rho_i)_+ \quad \forall t \in \mathcal{T}_k, i \in [m]$

Update the remaining capacity: $c_{k,i} \leftarrow c_{k-1,i} - \sum_{t\in\mathcal{T}_k} z_{t,i} \quad \forall i \in [m]$

end for

Table EC.5 Numerical Performance for year 2015 (2016, resp.) of `RO-Learning`, with and without batched matching. We use penalty parameters $\alpha = 3$ and $\gamma = 5$ (as used for Table 1 in Section 5) to evaluate the objective defined in equation (Objective). We use the batch size of 30 cases per batch. The total number of cases is $T = 3819$ (4980, resp.) for year 2015 (2016, resp.).

	Employment Rate (%)	Total Over- allocation	Average Backlog	Objective (in eq. (Objective))
`RO-Learning`	44.6 (46)	71 (81)	151 (199)	737 (1054.3)
`RO-Learning-B`	44.2 (45.8)	71 (79)	155.6 (219.6)	698.4 (944.0)
`RO-Learning-B-Iterate`	44.0 (45.3)	67 (71)	142.1 (189.5)	768.5 (1097.3)

EC.15. An Alternative Service Model with Server Idleness

In our main model, we have assumed that the service availability sequence $\{\mathbf{s}_t\}_{t=1}^T$ consists of i.i.d. Bernoulli random variables with mean $r_i = \rho + \epsilon$. In particular, this assumption implies that the server does not necessarily remain idle if it becomes available when no case is waiting. This modeling choice is relevant in settings where servers may initiate other tasks when the system is empty. For example, case workers in our partner may assist other teams when there is no case to serve.

In other contexts, however, a server may remain idle when the queue is empty, allowing them to serve the next arriving case without delay. In this section, we describe an alternative service model that explicitly allows for such idleness, and we extend the impossibility result of Proposition 2. Section EC.15.1 formally defines the idle-server model. At first glance, one might hope that this idleness would help an online algorithm better manage its backlogsperhaps weakening the impossibility result stated in Proposition 2. However, in Section EC.15.2, we show that even under this more favorable regime, no online algorithm making integral decisions can achieve sublinear regret when the congestion penalty parameter γ is $\Omega(T)$. Finally, in Section EC.15.3, we show that the regret upper bounds for `CA-DL` (Theorem 1 and Section EC.6.6) and `CO-DL` extend to the alternative model with server idleness. Thus, `CA-DL` continues to achieve sublinear regret in all regimes where it is possible (i.e., when $\gamma = o(T)$) under the alterative service model.

EC.15.1. Model

Model. We now formally describe an extension of our base model that incorporates server idleness. As in the base model, a server at each affiliate i becomes newly available with probability $r_i = \rho_i + \epsilon$, modeled by an i.i.d. Bernoulli random variable $s_{t,i} \sim \texttt{Ber}(r_i)$. We refer to this as a *fresh service availability*. If a server becomes newly available when there is no case to serve, it enters the idle state and remains idle until a new case arrives. To capture extra availability from idleness, we introduce a binary idleness state variable $\ell_{t,i} \in \{0, 1\}$, where $\ell_{t,i} = 1$ if the server has remained idle and thus is immediately available for service for case t. We emphasize that $s_{t,i}$ and $\ell_{t,i}$ refer to different modes of availability for the same servernot to separate servers.

We now formalize the service dynamics. Each period t begins with the arrival of a case t with type $\mathbf{A}_t = (\mathbf{w}_t, i_t^\dagger)$. The algorithm immediately selects a matching decision $\mathbf{z}_t \in \mathcal{Z}^{\text{int}}(i_t^\dagger)$ for case t. We define a *pre-processed backlog* $q_{t,i}$, which is updated as

$$q_{t,i} := b_{t-1,i} + z_{t,i}.$$

Next, for each affiliate i, we draw a fresh service availability $s_{t,i} \sim \text{Bernoulli}(\rho_i + \epsilon)$, indicating whether the server at affiliate i becomes newly available during period t. A server is able to serve a case at period t if it has become newly available (i.e., $s_{t,i} = 1$) or has already been idle (i.e., $\ell_{t,i} = 1$). When both conditions $s_{t,i} = 1$ and $\ell_{t,i} = 1$ hold, the case is served using the servers idle status, and the fresh service opportunity $s_{t,i}$ does not carry over. This naturally reflects that an idle server immediately becomes busy once it serves an arriving case.[60] Formally, we define the *effective availability* as:

$$u_{t,i} := \ell_{t,i} + (1 - \ell_{t,i}) \cdot s_{t,i}, \tag{EC.162}$$

and update the backlog (after processing case t) as:

$$b_{t,i} = (q_{t,i} - u_{t,i})_+ = (b_{t-1,i} + z_{t,i} - u_{t,i})_+. \tag{EC.163}$$

We now describe how the servers idleness state is updated. A server enters or remains idle at period $t+1$ if and only if (i) it is not used to serve a case at time t and (ii) is available in some formeither because it was already idle, or because it has just become newly available and found no case awaiting service. On the other hand, a server's idle status will reset to zero whenever it serves a case. Concretely, we have:

- If the server was idle ($\ell_{t,i} = 1$) and there is no case to serve ($q_{t,i} = 0$), it remains idle ($\ell_{t+1,i} = 1$).
- If no case was waiting ($q_{t,i} = 0$) and the server became newly available ($s_{t,i} = 1$), it enters the idle state ($\ell_{t+1,i} = 1$).
- If an idle server serves an arriving case ($\ell_{t,i} = 1$, $q_{t,i} \geq 1$), it becomes unavailable and the idle status resets to zero ($l_{t+1,i} = 0$).

Combining the above, we can succinctly write the idleness dynamics as:

$$\ell_{t+1,i} = \mathbb{1}[q_{t,i} = 0 \text{ and } (\ell_{t,i} = 1 \text{ or } s_{t,i} = 1)]. \tag{EC.164}$$

Decision, Benchmark, and Regret. An online algorithm π makes an immediate and irrevocable matching decision $\mathbf{z}_t^\pi \in \mathcal{Z}(i_t^\dagger)$ for each arriving case t. This decision may depend on the history $\mathcal{H}_{t-1} = \{\mathbf{A}_\tau, \mathbf{s}_\tau, \mathbf{z}_\tau^\pi\}_{\tau=1}^{t-1}$ and the current arrival type $\mathbf{A}_t$, but not on the current realization of fresh service

[60] One could equivalently define the model by drawing $s_{t,i}$ only when $\ell_{t,i} = 0$. However, for simplicity of exposition-particularly to define the offline benchmark in Definition EC.65 in a consistent waywe draw $s_{t,i}$ at every time step, regardless of the servers idle status. Alternatively, one can interpret $s_{t,i}$ as an indicator that the server *would have become newly available* at time t had it not already been idle.

availability $\mathbf{s}_t$. We further allow the algorithm to observe the server's idle status $\boldsymbol{\ell}_t$ when making its decision, which is well defined since $\boldsymbol{\ell}_t$ is determined by prior state variables $\ell_{t-1,i}$ and $\mathcal{H}_{t-1}$ through (EC.164). The objective value of algorithm π on a given sample path of arrivals and services, with penalty parameters (α, γ), is denoted by $\mathsf{ALG}^{\pi}(\alpha, \gamma)$ as in (Objective).

Similar to Definition 2, we define the offline benchmark as an algorithm that observes the full arrival sequence $\{\mathbf{A}_t\}_{t=1}^T$ and the fresh service availability sequence $\{\mathbf{s}_t\}_{t=1}^T$. A key difference from the base model in Section 2, however, is that the backlog dynamics now depend on the idle status of servers, which is endogenously determined by the offline's own matching decisions. To capture this dependency, we augment the offline optimization program to include idle-state dynamics, as formally defined below.

Definition EC.65 **(Offline Benchmark with Server Idleness)**. *Given the full sample path $\{\mathbf{A}_t, \mathbf{s}_t\}_{t=1}^T$ of arrivals and fresh service availabilities, the offline benchmark solves the following optimization problem:*

$$\mathsf{OPT}(\alpha,\gamma) := \max_{\substack{\mathbf{z}_t \in \mathcal{Z}(i_t^\dagger) \\ \mathbf{b}_t \geq \mathbf{0}}} \sum_{t=1}^{T} \mathbf{w}_t \cdot \mathbf{z}_t - \alpha \sum_{i=1}^{m} \left(\sum_{t=1}^{T} z_{t,i} - c_i \right)_+ - \frac{\gamma}{T} \sum_{t=1}^{T} \sum_{i=1}^{m} b_{t,i}$$

$$s.t. \quad \sum_{t=1}^{T} \mathbb{1}[i_t^\dagger = 0] z_{t,i} \leq \left(c_i - \sum_{t=1}^{T} \mathbb{1}[i_t^\dagger = i] z_{t,i} \right)_+ \quad \forall i \in [m] \qquad \text{(Capacity Feasibility)}$$

$$b_{t,i} \geq b_{t-1,i} + z_{t,i} - u_{t,i} \quad \forall t \in [T],\ i \in [m] \qquad \text{(Backlog Dynamics)}$$

$$u_{t,i} = \ell_{t,i} + (1 - \ell_{t,i}) \cdot s_{t,i} \qquad \text{(Effective Availability)}$$

$$\ell_{t+1,i} = \mathbb{1}[q_{t,i} = 0 \text{ and } (\ell_{t,i} = 1 \text{ or } s_{t,i} = 1)] \quad \forall t < T,\ i \in [m] \qquad \text{(Idle-State Update)}$$

with initial conditions $b_{0,i} = 0$ and $\ell_{1,i} = 0$ for all $i \in [m]$.

With the above definition of the offline benchmark, we define a regret of an online algorithm π the same way as in Definition 3.

EC.15.2. Impossibility Result under Alternative Service Model

We now extend the impossibility result in Proposition 2 to the model with server idleness.

Proposition EC.66 **(Lower Bound on Achievable Regret with Server Idleness)**. *For $\gamma = \Omega(T)$ and any service slack parameter $\epsilon \geq 0$ such that the resulting service rates satisfy $r_i \in [\rho_i + \epsilon,\ 1)$ for all $i \in [m]$, there exists an instance for which the regret of any online algorithm making integral decisions is $\Omega(T)$ under the model with server idleness.*[61]

[61] While our original impossibility result (Proposition 1) does not rely on the integrality assumption, we assume integral decisions ($z_t \in \{0, 1\}$) for technical clarity in Proposition 2. This ensures that the backlog evolves in discrete units, which simplifies our analysisparticularly in deriving certain inequalities such as (EC.172) and (EC.173). With more refined analysis, we conjecture that a similar result would hold even without the integrality assumption.

To prove Proposition EC.66, we consider the same instance used in the proof of Proposition 2 (which we re-iterate below for completeness). Under the alternative service model, the server remains idle when it becomes available and finds no case to serve. This appears to offer an online algorithm additional flexibility, especially when it can also observe the idle status. However, the potential benefit of idleness is fundamentally constrained by a key tradeoff: idle periods can only arise when the algorithm chooses *not* to match a case (to an actual affiliate). Yet, to avoid linear regret, the algorithm must match almost as many cases as the benchmark. These two objectivespreserving idle periods and collecting enough rewardare directly in conflict. Building on this intuition, we show that if the algorithm matches enough cases to avoid linear regret, then idle periods become too scarce to meaningfully reduce congestion, and the algorithm must once again incur a constant average backlog in the similar vein of Proposition 2.

Proof of Proposition EC.66. We consider an instance with $m = 1$. For brevity, we omit the subscript for i. There are T arrivals with deterministic reward $w_t = 1$ for all $t \in [T]$. There are no tied cases and the capacity is $c = 0.5T$. The service rate is $r = 0.5 + \epsilon$ where $\epsilon \in [0, 0.5)$. Note that we require $\epsilon < 0.5$ to exclude a trivial regime where service rate is 1. Thus, we assume that $0.5 - \epsilon$ is a constant (independent of T) bounded away from zero. We further recall that α does not play any role here. Hence, we denote the objective value of offline benchmark (Definition EC.65) and algorithm by $\mathsf{OPT}(\gamma)$ and $\mathsf{ALG}(\gamma)$, respectively.

Similar to Claim-EC.1, we first lower-bound the value of the offline benchmark.

Claim EC.67. *For any $\epsilon \in [0, 0.5)$, the value of offline benchmark under the idle-server model (Definition EC.65) satisfies*

$$\mathbb{E}[\mathsf{OPT}(\gamma)] \geq 0.5T - \Theta(\sqrt{T}).$$

Proof of Claim EC.67 The proof mirrors Claim EC.1, with a minor adjustment. Consider the following solution: match a case at time t to the actual affiliate ($z_t = 1$) if and only if $s_t = 1$, until reaching the capacity limit $\sum_t z_t \leq 0.5T$. This is feasible by construction. Note that from equation (EC.162), a server is always available when $s_t = 1$: either via fresh availability or by having remained idle from an earlier period. Thus, this feasible solution does not incur any backlog. The rest of the proof follows the same line of argument in Claim-EC.1 and is thus omitted. □

Fix any online algorithm with integral decisions, and let $z_t \in \{0, 1\}$ denote its matching decision at time t. Since z_t is binary, the backlog evolves in integer increments and remains a non-negative integer. By Claim-EC.67, the offline benchmark collects approximately $0.5T$ in reward with zero backlog. Hence, to achieve sublinear regret, the algorithm must accept at least $0.5T - o(T)$ cases in expectation. Thus, without loss of generality, we assume:

$$0.5T - o(T) \leq \mathbb{E}\left[\sum_{t=1}^{T} \mathbb{1}[z_t = 1]\right] \leq 0.5T, \tag{EC.165}$$

where the last inequality is simply just due to the capacity constraint.

We now show that under (EC.165), any such online algorithm must incur a constant average backlog:

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}[b_t] \geq \Omega(1). \tag{EC.166}$$

Combining (EC.165), (EC.166), and Claim-EC.67, we obtain:

$$\mathsf{OPT}(\gamma) - \mathsf{ALG}(\gamma) \geq \Omega(\gamma - \sqrt{T}). \tag{EC.167}$$

Thus, whenever $\gamma = \Omega(T)$, the regret of any online algorithm is $\Omega(T)$.

We now prove (EC.166) by contradiction. Assume instead that the expected total backlog is sub-linear:

$$\sum_{t=1}^{T}\mathbb{E}[b_t] \leq o(T). \tag{EC.168}$$

Assuming both (EC.165) and (EC.168), we derive a contradiction in four steps. First, we show that most accepted cases must occur when the backlog is empty (Claim EC.68). Second, building on Step 1, we establish a lower bound on the number of matches that must be served via idle servers (Claim EC.69). Third, we derive an upper bound on the total number of such matches (Claim EC.70). Finally, we show that these lower and upper bounds are incompatible unless $\epsilon \geq 0.5$, which contradicts our assumption that $\epsilon < 0.5$ (ensuring the service rate remains strictly below one). This contradiction implies that any online algorithm satisfying (EC.165) must incur a total backlog of $\Omega(T)$. We elaborate on each step below. The proofs of all auxiliary claims are deferred to the end of this section.

Step 1. The following claim shows that the majority of matches must occur when the backlog is empty in order to satisfy both assumptions (EC.165) and (EC.168).

Claim EC.68. *Under assumptions* (EC.165) *and* (EC.168)*, we have:*

$$\sum_{t=1}^{T}\mathbb{E}\left[\mathbb{1}[b_{t-1}=0,\, z_t=1]\right] \geq 0.5T - o(T). \tag{EC.169}$$

The proof builds on a simple algebraic property of the backlog and the integrality of decisions.

Step 2. We build on Step 1 to lower bound the number of idle periods that an online algorithm must utilize. Specifically, the following claim shows that most matched cases must be served by an idle server.

Claim EC.69. *Under assumptions* (EC.165) *and* (EC.168)*, we have:*

$$\sum_{t=1}^{T}\mathbb{E}\left[\mathbb{1}[b_{t-1}=0,\, z_t=1,\, \ell_t=1]\right] \geq 0.5T - o(T). \tag{EC.170}$$

The proof uses the fact that matches made without idle servers create a backlog unless fresh availability occurs (i.e., $s_t = 1$). Because an online algorithm does not observe the realization of s_t, such matches lead to backlog with constant probability. We use this to argue that if a linear number of matches occur without idle servers, the total backlog would grow linearly in Tcontradicting assumption (EC.168).

Step 3. We now establish an upper bound on the number of idle periods that an online algorithm can utilize. The following claim shows that idle periods are 'too scarce' to serve many matches.

CLAIM EC.70. *Under assumptions* (EC.165) *and* (EC.168)*, for any* $\epsilon \in [0, 0.5)$*, we have:*

$$\sum_{t=1}^{T} \mathbb{E}\left[\mathbb{1}[b_{t-1} = 0,\, z_t = 1,\, \ell_t = 1]\right] \le (0.5 + \epsilon) \cdot 0.5T + o(T). \tag{EC.171}$$

The proof uses a simple charging argument: each idle period must be preceded by a time when the algorithm chose not to match a case, despite having no backlog. By Step 1, any algorithm satisfying assumptions (EC.165) and (EC.168) must match most cases during such zero-backlog periodsleaving few opportunities for servers to remain idle. We use this intuition to upper bound the total number of idle periods that an online algorithm can utilize.

Step 4. Putting everything together. We now combine the previous claims to prove (EC.166). Combining Claims EC.69 and EC.70, we obtain:

$$0.5T - o(T) \le 0.5(0.5 + \epsilon)T + o(T).$$

By dividing by T and letting $T \to \infty$, we have:

$$0.5 \le 0.5(0.5 + \epsilon),$$

which implies $\epsilon \ge 0.5$. However, this contradicts our requirement that $\epsilon < 0.5$. It follows that our initial assumption (EC.168) must be false. Hence, every algorithm that accepts at least $0.5T - o(T)$ cases must incur $\Omega(T)$ total backlog, which proves (EC.166). This completes the proof. □

Proof of Claim EC.68. Observe that for any sample path, we have:

$$b_t = (b_{t-1} + z_t - u_t)_+ \ge (2 - u_t) \cdot \mathbb{1}[b_{t-1} \ge 1,\, z_t = 1] \ge \mathbb{1}[b_{t-1} \ge 1,\, z_t = 1], \tag{EC.172}$$

where final inequality uses that $u_t = l_t + (1 - l_t) \cdot s_t \le 1$.

Taking expectations and summing over time $t \in [T]$:

$$\begin{aligned}
\mathbb{E}\left[\sum_{t=1}^{T} b_t\right] &\ge \mathbb{E}\left[\sum_{t=1}^{T} \mathbb{1}[b_{t-1} \ge 1,\, z_t = 1]\right] \\
&= \mathbb{E}\left[\sum_{t=1}^{T} \mathbb{1}[z_t = 1] - \sum_{t=1}^{T} \mathbb{1}[b_{t-1} = 0,\, z_t = 1]\right] \\
&\ge 0.5T - o(T) - \sum_{t=1}^{T} \mathbb{E}\left[\mathbb{1}[b_{t-1} = 0,\, z_t = 1]\right],
\end{aligned}$$

where the last inequality uses assumption (EC.165). Now combining this with the assumption (EC.168), we conclude:

$$\sum_{t=1}^{T} \mathbb{E}\left[\mathbb{1}[b_{t-1} = 0,\, z_t = 1]\right] \ge 0.5T - o(T).$$

□

Proof of Claim EC.69. We begin by decomposing the indicator:

$$\sum_{t=1}^{T}\mathbb{E}\left[\mathbb{1}[b_{t-1}=0,\, z_t=1,\, \ell_t=1]\right]=\sum_{t=1}^{T}\mathbb{E}\left[\mathbb{1}[b_{t-1}=0,\, z_t=1]\right]-\sum_{t=1}^{T}\mathbb{E}\left[\mathbb{1}[b_{t-1}=0,\, z_t=1,\, \ell_t=0]\right].$$

By Claim EC.68, the first term is at least $0.5T-o(T)$. Therefore, it suffices to show that:

$$\sum_{t=1}^{T}\mathbb{E}\left[\mathbb{1}[b_{t-1}=0,\, z_t=1,\, \ell_t=0]\right]\le o(T).$$

To prove this, note that if $b_{t-1}=0$, $z_t=1$, and $\ell_t=0$, then the case is served only if $s_t=1$. Otherwise, it contributes to the backlog. Thus, for every sample path, we have:

$$b_t\ge \mathbb{1}[b_{t-1}=0,\, z_t=1,\, \ell_t=0]\cdot(1-s_t). \tag{EC.173}$$

Taking expectations:

$$\begin{aligned}\mathbb{E}[b_t]&\ge\mathbb{E}\left[\mathbb{1}[b_{t-1}=0,\, z_t=1,\, \ell_t=0]\cdot(1-s_t)\right]\\&=\mathbb{E}\left[\mathbb{1}[b_{t-1}=0,\, z_t=1,\, \ell_t=0]\cdot\mathbb{E}[1-s_t\mid\mathcal{H}_{t-1}]\right]\\&=(0.5-\epsilon)\cdot\mathbb{E}\left[\mathbb{1}[b_{t-1}=0,\, z_t=1,\, \ell_t=0]\right].\end{aligned}$$

In the first inequality, we applied the tower property of expectation: conditioned on the algorithm's history $\mathcal{H}_{t-1}$, the indicator $\mathbb{1}[b_{t-1}=0,\, z_t=1,\, \ell_t=0]$ is deterministic and the random variable s_t is independent of $\mathcal{H}_{t-1}$ and follows the Bernoulli distribution with success probability $0.5+\epsilon$.

Summing over all $t\in[T]$ gives:

$$\sum_{t=1}^{T}\mathbb{E}[b_t]\ge(0.5-\epsilon)\cdot\sum_{t=1}^{T}\mathbb{E}\left[\mathbb{1}[b_{t-1}=0,\, z_t=1,\, \ell_t=0]\right].$$

Using (EC.168) and because $0.5-\epsilon$ is a strictly positive constant, it follows that:

$$\sum_{t=1}^{T}\mathbb{E}\left[\mathbb{1}[b_{t-1}=0,\, z_t=1,\, \ell_t=0]\right]\le o(T).$$

This completes the proof. □

Proof of Claim EC.70. Define the indicator:

$$\texttt{newly\_idle}_t:=\mathbb{1}[q_t=0,\, \ell_t=0,\, s_t=1],$$

which captures the event that the server transitions from non-idle to idle at period t.

By construction, every match served by an idle server (i.e., when $b_{t-1}=0$, $z_t=1$, and $\ell_t=1$) must be preceded by a time when the server entered the idle state. Thus, we can charge each such match to a previous $\texttt{newly\_idle}_t$ event, yielding:

$$\sum_{t=1}^{T}\mathbb{E}\left[\mathbb{1}[b_{t-1}=0,\, z_t=1,\, \ell_t=1]\right]\le\sum_{t=1}^{T}\mathbb{E}[\texttt{newly\_idle}_t].$$

Next, observe that a server can become newly idle at time t only if $q_t = 0$ — or equivalently $b_{t-1} = 0$ and $z_t = 0$ — and $s_t = 1$. Therefore:

$$\texttt{newly\_idle}_t \le \mathbb{1}[b_{t-1} = 0,\, z_t = 0,\, s_t = 1],$$

and so:

$$\mathbb{E}[\texttt{newly\_idle}_t] \le \mathbb{E}[\mathbb{1}[b_{t-1} = 0,\, z_t = 0]] \cdot (0.5 + \epsilon),$$

where we used the tower rule and independence of s_t from the algorithms history $\mathcal{H}_{t-1}$.

Now summing over t, we obtain:

$$\begin{aligned}
\sum_{t=1}^{T} \mathbb{E}[\texttt{newly_idle}_t] &\le (0.5+\epsilon) \cdot \sum_{t=1}^{T} \mathbb{E}[\mathbb{1}[b_{t-1} = 0,\, z_t = 0]] \\
&= (0.5+\epsilon) \cdot \left(\sum_{t=1}^{T} \mathbb{E}[\mathbb{1}[b_{t-1} = 0]] - \sum_{t=1}^{T} \mathbb{E}[\mathbb{1}[b_{t-1} = 0,\, z_t = 1]] \right) \\
&\le (0.5+\epsilon) \cdot (T - 0.5T + o(T)) \\
&= 0.5(0.5+\epsilon)T + o(T),
\end{aligned}$$

where the last inequality follows from Claim EC.68 (and by a trivial bound $\mathbb{E}[\mathbb{1}[b_{t-1} = 0]] \le 1$). This completes the proof. □

EC.15.3. Regret Upper bounds under Alternative Service Model

We now explain how our regret upper bounds for `CA-DL` (Theorem 1, Corollary 1) and `CO-DL` (Theorem 2) seamlessly extend to the alternative service model with server idleness. Both of our proposed algorithms naturally adapt to the alternative model. Specifically, `CA-DL` is identical to Algorithm 1 except that it now uses the backlog updated via equation (EC.163)that is, based on the effective availability $u_{t,i}$ defined in equation (EC.162), which accounts for server idleness. The following corollary shows that `CA-DL` achieves the same regret upper bounds under this alternative model.

Corollary EC.71 **(Regret of `CA-DL` under Server Idleness).** *Let $\eta = \Theta(1/\sqrt{T})$ and $\zeta = \Theta(1/\sqrt{T})$. Under the stable regime (Definition 1), the regret of `CA-DL` under the service model with server idleness is $\mathcal{O}(\sqrt{T} + \frac{\gamma}{\epsilon})$. Furthermore, under the near-critical regime (Definition 1), the regret of `CA-DL` is $\mathcal{O}(\sqrt{\gamma T})$ by setting $\eta = \Theta(1/\sqrt{T})$ and $\zeta = \Theta(\sqrt{\gamma/T})$.*

Proof of Corollary EC.71. The proof mirrors that of Theorem 1 (stable regime) and Corollary 1 (near-critical regime); we only highlight a minor change. The only difference lies in inequality (EC.14) for the proof of Lemma 4 (Section EC.6.2). Under the new backlog update rule (EC.163), we can establish an analogous drift inequality to Lemma 3, that is, $D_t \le \mathbf{b}_{t-1} \cdot (\mathbf{z}_t - \mathbf{u}_t) + \mathcal{O}(1)$, where D_t is the drift of the quadratic Lyapunov function ψ (see equation (13)). This change requires replacing $\mathbf{s}_t$ in line (EC.14) with $\mathbf{u}_t$, and verifying $\mathbb{E}[u_{t,i} \mid \mathcal{H}_{t-1}] \ge \rho_i + \epsilon$ for all $i \in [m]$. Since $u_{t,i} = s_{t,i} + \ell_{t,i}(1 - s_{t,i}) \ge s_{t,i}$ for all sample paths (see equation (EC.162)), we have $\mathbb{E}[u_{t,i} \mid \mathcal{H}_{t-1}] \ge \mathbb{E}[s_{t,i} \mid \mathcal{H}_{t-1}] \ge$

$\rho_i + \epsilon$. That is, allowing server idleness can only increase the service rate. The rest of the argument is unchanged. Specifically, inequality (20) still holds under the alternative service model, thereby extending Theorem 1 and Corollary 1. □

For `CO-DL`, since it does not use backlog information, the algorithm remains unchanged. The regret guarantee of Theorem 2 extends to the alternative service model. The proof again follows analogous arguments to those used in the proof of Corollary EC.71 and is omitted for brevity.